

Plano de ensino

Curso: CCI-BAC - Bacharelado em Ciência da Computação

Turma: CCI031-07U - CIENCIA DA COMPUTAÇÃO FASE 07

Disciplina: ETG-I - ESTAGIO CURRICULAR I

Período letivo: 2017/1

Carga horária: 180

Professor: 3374807 - LUCIANA RITA GUEDES

Ementa

1. Desenvolvimento e apresentação de um trabalho prático final de curso na área de computação ou informática que pode ser realizado na própria instituição ou em empresa pública ou privada.

Objetivo geral

1. O estágio curricular visa oportunizar ao acadêmico a aplicação prática dos conhecimentos adquiridos no curso e uma vivência efetiva da prática profissional na sua área de atuação.

Objetivo específico

1. - Possibilitar a aplicação dos conhecimentos adquiridos no curso;
- Possibilitar a vivência da prática profissional
- Possibilitar o conhecimento do mercado de trabalho

Conteúdo programático

1. 1. Apresentação
Orientação aos alunos sobre as regras e procedimentos envolvidos no estágio curricular, apresentação do cronograma, explanação sobre o Termo de Compromisso e o Plano de Estágio.
2. Aplicação dos conhecimentos adquiridos durante o curso de graduação a casos reais ou teóricos envolvendo Ciência da Computação e afim, em empresas privadas e/ou públicas.

Metodologia

1. O estágio é realizado sob a orientação e acompanhamento de um professor orientador (professor efetivo do CCT) e de um supervisor na concedente. É precedido do termo de compromisso - contrato de celebrado entre acadêmico, empresa e instituição de ensino - e do plano de estágio - planejamento das atividades a serem desenvolvidas e no qual deve ficar claro que estas atividades condizem com a área do curso. Ao final do semestre, o acadêmico deverá submeter um relatório do estágio para avaliação por dois professores e que deverá conter a avaliação de desempenho emitida pela unidade concedente do estágio.

Sistema de avaliação

1. O desempenho do aluno será avaliado com base nos seguintes itens e critérios:
a) Relatório de estágio (Peso: 60%): será avaliado por dois professores em relação ao cumprimento das atividades propostas, adequação do texto quanto às normas vigentes para trabalhos técnicos, clareza e encadeamento na definição do escopo e dos resultados. O aluno deve sanar as falhas apontadas no relatório pelos avaliadores.
b) Avaliação da empresa (Peso 40%): será feita pelo supervisor da unidade concedente, avaliando uma série de quesitos como: assiduidade, pontualidade, iniciativa, organização, cumprimento de prazos, etc.

Bibliografia básica

1. Lei no. 11.788 de 25 de setembro de 2008 (Lei do Estágio), disponível em http://www.joinville.udesc.br/portal/estagios/arquivos/nova_lei_estagio_11788.pdf

Modelos dos documentos necessários (termo de compromisso, plano de estágio)
Disponível em <http://www.cct.udesc.br/?id=481>

Resoluções da UDESC sobre estágio curricular,
Disponíveis em <http://www.cct.udesc.br/?id=485>

Bibliografia complementar

1. Manual para Elaboração de Trabalhos Acadêmicos da UDESC, disponível em <http://secon.udesc.br/consepe/resol/2012/001-2012-cpe-anexo.pdf>

Plano de ensino

Curso: CCI-BAC - Bacharelado em Ciência da Computação

Turma: CCI031-08U - CIENCIA DA COMPUTAÇÃO FASE 08

Disciplina: ETG-II - ESTAGIO CURRICULAR II

Período letivo: 2017/1

Carga horária: 180

Professor: 3374807 - LUCIANA RITA GUEDES

Ementa

1. Desenvolvimento e apresentação de um trabalho prático final de curso na área de computação ou informática que pode ser realizado na própria instituição ou em empresa pública ou privada.

Objetivo geral

1. O estágio curricular visa oportunizar ao acadêmico a aplicação prática dos conhecimentos adquiridos no curso e uma vivência efetiva da prática profissional na sua área de atuação.

Objetivo específico

1. - Possibilitar a aplicação dos conhecimentos adquiridos no curso;
- Possibilitar a vivência da prática profissional
- Possibilitar o conhecimento do mercado de trabalho

Conteúdo programático

1. 1. Apresentação
Orientação aos alunos sobre as regras e procedimentos envolvidos no estágio curricular, apresentação do cronograma, explanação sobre o Termo de Compromisso e o Plano de Estágio.
2. Aplicação dos conhecimentos adquiridos durante o curso de graduação a casos reais ou teóricos envolvendo Ciência da Computação e afim, em empresas privadas e/ou públicas.

Metodologia

1. O estágio é realizado sob a orientação e acompanhamento de um professor orientador (professor efetivo do CCT) e de um supervisor na concedente. É precedido do termo de compromisso - contrato de celebrado entre acadêmico, empresa e instituição de ensino - e do plano de estágio - planejamento das atividades a serem desenvolvidas e no qual deve ficar claro que estas atividades condizem com a área do curso. Ao final do semestre, o acadêmico deverá submeter um relatório do estágio para avaliação por dois professores e que deverá conter a avaliação de desempenho emitida pela unidade concedente do estágio.

Sistema de avaliação

1. O desempenho do aluno será avaliado com base nos seguintes itens e critérios:
a) Relatório de estágio (Peso: 60%): será avaliado por dois professores em relação ao cumprimento das atividades propostas, adequação do texto quanto às normas vigentes para trabalhos técnicos, clareza e encadeamento na definição do escopo e dos resultados. O aluno deve sanar as falhas apontadas no relatório pelos avaliadores.
b) Avaliação da empresa (Peso 40%): será feita pelo supervisor da unidade concedente, avaliando uma série de quesitos como: assiduidade, pontualidade, iniciativa, organização, cumprimento de prazos, etc.

Bibliografia básica

1. Lei no. 11.788 de 25 de setembro de 2008 (Lei do Estágio), disponível em http://www.joinville.udesc.br/portal/estagios/arquivos/nova_lei_estagio_11788.pdf

Modelos dos documentos necessários (termo de compromisso, plano de estágio)
Disponível em <http://www.cct.udesc.br/?id=481>

Resoluções da UDESC sobre estágio curricular,
Disponíveis em <http://www.cct.udesc.br/?id=485>

Bibliografia complementar

1. Manual para Elaboração de Trabalhos Acadêmicos da UDESC, disponível em <http://secon.udesc.br/consepe/resol/2012/001-2012-cpe-anexo.pdf>

Plano de ensino

Curso: CCI-BAC - Bacharelado em Ciência da Computação

Turma: CCI122-01A - CIENCIA DA COMPUTAÇÃO FASE 01A

Disciplina: AGT0001 - ALGORITMOS

Período letivo: 2017/1

Carga horária: 72

Professor: 3639428 - EVERLIN FIGHERA COSTA MARQUES

Ementa

1. Noções de arquitetura e programação de computadores. Algoritmo, fluxograma e pseudo-codificação. Entrada e saída de dados. Constantes e variáveis. Operadores e expressões. Desvios e laços. Vetores e matrizes. Programação estruturada. Experimentação em linguagem de alto nível.

Objetivo geral

1. Capacitar os estudantes a definir soluções de problemas através do desenvolvimento de algoritmos a serem executados por computador.

Objetivo específico

1. Conhecer noções de arquitetura e programação de computadores.
Construir soluções para problemas computacionais tanto na forma de algoritmo, fluxograma ou pseudo-código, envolvendo comandos básicos, vetores e matrizes.
Implementar um problema em pseudocódigo em uma ferramenta de linguagem de programação de alto nível.

Conteúdo programático

1. 4 h
 1. Noções de arquitetura e programação de computadores.
 - 1.1. Conceitos de Hardware e componentes do computador
 - 1.2. Unidades de Entrada e Saída
 - 1.3. Organização e tipos de Memórias
 - 1.4. Unidade Central de Processamento
 - 1.5. Conceito de Software e tipos de Software
 - 1.6. Conceito de Algoritmo e formas de representação
2. 16 h
 2. Sintaxe Básica de Pseudocódigo
 - 2.1. Estrutura de um algoritmo
 - 2.2. Tipos de dados
 - 2.3. Variáveis e constantes
 - 2.4. Operadores e expressões
 - 2.5. Operador de atribuição
 - 2.6. Comandos de entrada e saída
 3. Desvios e Laços
 - 3.1. Estruturas de seleção
 - 3.1.1. Seleção simples: (SE...ENTÃO)
 - 3.1.2. Seleção composta: (SE...ENTÃO...SENÃO)
 - 3.1.3. Múltiplas escolhas: (CASO)
3. 16 h
 4. Estruturas de repetição
 - 4.1. Teste no início (ENQUANTO...FAÇA)
 - 4.2. Teste no fim (REPITA...ATÉ)
 - 4.3. Teste no início e variável de controle (PARA...FAÇA)
4. 16 h
 5. Vetores e Matrizes
 - 5.1. Vetores unidimensionais
 - 5.2. Manipulação de strings (vetores de caracteres)
 - 5.3. Vetores bidimensionais (matrizes)

5. 16 h

Plano de ensino

6. Procedimentos e Funções
6.1. Passagem de parâmetro (por valor e por referência)
6.2. Recursividade

6. 4 h

7. Experimentação em linguagem de alto nível
7.1. Introdução à linguagem C

Metodologia

1. Aulas expositivas e dialogadas em sala
Aulas práticas em laboratório de informática
Uso de software para escrever, compilar e testar os algoritmos criados nas aulas

Obs: até 20% (vinte por cento) da carga horária poderá ser desenvolvida à distância com apoio do Ambiente Virtual de Aprendizagem Moodle, disponível na Universidade.

Sistema de avaliação

1. Do desempenho do discente:
A qualidade do desempenho do estudante será avaliada com base no desenvolvimento das seguintes atividades e com os seguintes critérios:
Prova 1 (escrita) = 20%
Prova 2 (escrita) = 25%
Prova 3 (escrita) = 25%
Exercícios avaliativos (escritos e práticos) = 10%
Trabalho final (prático) = 20%

Do desempenho da disciplina e do professor:
Os discentes terão, igualmente, a oportunidade de fazer uma avaliação mais completa do desempenho do professor e da disciplina através do sistema de avaliação (SIGA).

Bibliografia básica

1. DE OLIVEIRA, J.F.; MANZANO, José Augusto N. G. Algoritmos: Lógica para Desenvolvimento de Programação de Computadores. Editora Érica, 16a ed., 2004. ISBN 857194718X.
DE SOUZA, M.A.F., GOMES, M.M., SOARES, M.V., CONCILIO, R. Algoritmos e Lógica de Programação. Thomson Learning, 2004. ISBN 8522104646.
MEDINA, M., FERTIG, C. Algoritmos e Programação - Teoria e Prática. Novatec, 2005. ISBN 857522073X.

Bibliografia complementar

1. CORMEN, Thomas H. Algoritmos: teoria e prática. Rio de Janeiro: Campus, 2002. 916 p. ISBN 8535209263
DEITEL, Harvey M.; DEITEL, Paul J; KURBAN, Amir. Como programar em C. 2.ed. Rio de Janeiro: Livros Técnicos e Científicos, c1999. 486 p. ISBN 8521611919
GUIMARÃES, Angelo de Moura. Introdução à ciência da computação. Rio de Janeiro: Livros Técnicos e Científicos, 1984.
Instituto Brasileiro de Pesquisa em Informática. Dominando a linguagem C. Rio de Janeiro: IBPI, c1993. 236 p. ; ISBN 8585588012
RAMALHO, José Antônio. Introdução à informática: teoria e prática. São Paulo: Berkeley, 2000.
Artigos/materiais fornecidos pelo professor.

Plano de ensino

Curso: CCI-BAC - Bacharelado em Ciência da Computação

Turma: CCI122-01A - CIENCIA DA COMPUTAÇÃO FASE 01A

Disciplina: LMA0001 - LÓGICA MATEMÁTICA

Período letivo: 2017/1

Carga horária: 72

Professor: 3569829 - ROGERIO EDUARDO DA SILVA

Ementa

1. História da lógica. Cálculo proposicional. Fórmulas tautológicas, contra-válidas e consistentes. Transformação entre conectivos lógicos. Equivalências. Argumentos válidos. Teorema lógico. Axiomatização. Metodos de prova. Lógica de 1a. Ordem (LPO). Quantificadores. Fórmulas. Argumentos. Axiomatização. Noções sobre teorias lógicas (completude e corretude). Notação clausal. Introdução à Provas de LPO. Uso de uma linguagem baseada em lógica, como instância da LPO. Axiomatização. Noções sobre teorias lógicas (completude e corretude). Notação clausal. Introdução à Provas de LPO. Uso de uma linguagem baseada em lógica, como instância da LPO.

Objetivo geral

1. Apresentar os conceitos das lógicas proposicionais e de primeira ordem, provas de teoremas lógicos, e uma ferramenta derivada da lógica. Esta como um paradigma de programação. Desenvolver o raciocínio lógico-matemático e uma mentalidade alicerçada no rigor e na observação. Adquirir uma formação dedutiva e indutiva para efetuar pesquisas mais profundas principalmente na Matemática.

Objetivo específico

1. Especificamente, após o curso o estudante deverá ser capaz de:
 - . CARACTERIZAR o conceito de proposição e fórmulas proposicionais
 - . INTRODUIR os conceitos de provas de teoremas
 - . INTRODUIR à lógica como uma ferramenta de propósito geral à resolução de problemas
 - . INTRODUIR o uso de instância computacional da lógica de 1ª Ordem: Prolog
 - . INTRODUIR o conceito de provas de teoremas e do Prolog na resolução de Problemas
 - . Familiarizar-se com a inferência lógica e reconhecer como esta pode ser usada em Matemática e em outras ciências empíricas.
 - . INTRODUIR conceitos de lógica nebulosa

Conteúdo programático

1. Apresentação da Disciplina
Objetivos e ementa da disciplina
Conteúdo Programático
Critérios de Avaliação
Metodologia de Ensino
2. Introdução à Lógica Proposicional
O que é Lógica?
Conceitos introdutórios: proposição, princípios das proposições, valores lógicos, paradoxo
Conceitos introdutórios: alfabeto, fórmulas bem formadas, tabelas-verdade
Tabelas-verdade fundamentais: negação, conjunção, disjunção, disjunção exclusiva, implicação, bi-implicação
Exercícios
3. Construção de Tabelas-Verdade
Método por conector lógico
Método por símbolo do alfabeto
Precedência de conectores lógicos - uso de parêntesis
Verificação do valor lógico de proposições compostas
4. Tautologias, Contradições e Contingências
Definições e tabelas-verdade
Exercícios
5. Implicação Lógica
Definições e tabelas-verdade
Regras de Inferência: Adição, simplificação, silogismo disjuntivo, Modus Ponens
Regras de Inferência: Modus Tollens, silogismo hipotético
Exercícios
6. Equivalência Lógica
Definições e tabelas-verdade

Plano de ensino

Exercícios
7. Álgebra das Proposições Propriedades da Conjunção Propriedades da Disjunção Propriedades da Conjunção e Disjunção Negação da Condicional Negação da Bi-condicional Exercícios
8. Método Dedutivo Definição Demonstrações: Adição, Modus Ponens, Modus Tollens, Silogismo Disjuntivo Redução do Número de Conectivos Forma Normal Conjuntiva Forma Normal Disjuntiva Exercícios
9. Argumentos - Regras de Inferência Validade de Argumento Argumentos Válidos Fundamentais Regras de Inferência
10. Validade mediante Regras de Inferência Exemplos e Exercícios
11. Validade por Regras de Inferência e Equivalência Regra da Substituição Equivalências Notáveis Exemplos e Exercícios Inconsistências
12. Demonstração Condicional e Indireta Demonstração Condicional Demonstração Indireta Exemplos e Exercícios
13. Sentenças Abertas Sentenças abertas de uma variável Conjunto-verdade para sentenças abertas Sentenças abertas de duas variáveis Sentenças abertas de N variáveis Exercícios
14. Operações Lógicas sobre Sentenças Abertas Conjunção, disjunção e negação Condicional e bi-condicional Exercícios
15. Quantificadores Quantificador Universal e Existencial Variável aparente e Variável livre Princípio da Substituição de Variável Aparente Quantificador de Existência e Unicidade Negação de quantificadores
16. Quantificadores com Sentenças de N Variáveis Quantificação Parcial e Múltipla Negação de Quantificadores Múltiplos Exemplos e Exercícios
17. Lógica de Primeira Ordem Introdução Representação de Predicados
18. Lógica de Primeira Ordem II Definições e conceitos: termo, átomo, fórmula
19. Lógica de Primeira Ordem III Introdução à Lógica de Predicados Predicados Computáveis Unificação
20. Representação de Conhecimento Conhecimento declarativo vs Conhecimento Procedimental
21. Prolog Cláusulas de Horn Unificação e Backtracking

Plano de ensino

22. Prolog II Recursividade
23. Prolog III Listas
24. Resolução de Problemas Introdução aos Sistemas Especialistas Busca em Espaço de Soluções por Força Bruta
25. Lógica Nebulosa Introdução à Lógica Nebulosa
26. Revisão para Avaliação Exercícios de Revisão sobre o Conteúdo Ministrado
27. Lógica de Primeira Ordem IV Forma Normal: Conjuntiva e Disjuntiva Axiomatização
28. Lógica de Primeira Ordem V Regras de derivação Dedução natural com regras Prova direta
29. Lógica de Primeira Ordem VI Prova Indireta Prova por Implicação Indireta
30. Avaliação Prova Escrita
31. Projeto Final Desenvolvimento do Projeto Final de Disciplina
32. Semana Acadêmica Palestras sobre assuntos diversos pertinentes à disciplina
33. Palestra Palestra com professor convidado em tema relacionado à lógica e à programação
34. Atividades Extra-Classe Lista de exercícios Leitura de material didático de apoio Plantão de dúvidas online Evento acadêmico em tema relacionado a aplicações da lógica
35. Aula Prática Exercícios de Revisão e Preparação para Prova

Metodologia

1. Conteúdos expostos pelo professor com participação colaborativa dos estudantes, acompanhados pela proposição de exercícios práticos associados. Implementação de problemas clássicos da área. O conteúdo da disciplina poderá ser ministrado na modalidade de ensino a distância em até 20% do total de sua Carga Horária (MEC PORTARIA Nº 4.059, DE 10 DE DEZEMBRO DE 2004 publicado no DOU de 13/12/2004, Seção 1, p. 34).

Sistema de avaliação

1. Os estudantes serão avaliados com base no desempenho em avaliações escritas, exercícios de implementação e participação nas aulas e em eventos, conceituados de 0,0 a 10,0. Conforme regulamento da UDESC, o comparecimento às aulas deve ser de no mínimo 75%. Ao longo do semestre, serão realizados: - 3 avaliações escritas, P1, ao final do primeiro terço do semestre; P2, ao final da segunda parte e P3, ao final do semestre. A média destas corresponderá a 90% da nota final; - O projeto final, Pf, mediante o seu manuscrito (em formato de artigo) e defesa: 10% A média geral, M, portanto, será obtida por meio da seguinte fórmula: $M = 0,9[(P1+P2+P3)/3] + 0,1Pf$ Exame Caso o discente não obtenha média M igual ou superior a 7,0, um exame escrito será aplicado objetivando aferir o conhecimento teórico do estudante. Não há recuperação das provas P1, P2 e P3 por não comparecimento, exceto nos casos previstos no regulamento da UDESC. Do desempenho da disciplina e do professor: Os estudantes terão, igualmente, a oportunidade de efetuar, ao andamento da disciplina uma avaliação mais completa do desempenho do professor e da disciplina. O formulário para esta atividade será preparado para a ocasião ou fornecido pelo coordenador do curso.
--

Plano de ensino

Bibliografia básica

1. Guilherme Bittencourt. Inteligência Artificial - Ferramentas e Teorias, Editora da Unicamp, 10a. Escola de Computação, Instituto de Computação - UNICAMP, 1996.
 - ABE, Jair M. SCALZITTI, Alexandre. SILVA FILHO, João Inácio Introdução à lógica matemática para a Ciência da computação, São Paulo: Arte Ciência, 2001
 - Alencar Filho, E. - Iniciação à Lógica Matemática - Nobel, 1984.

Bibliografia complementar

1. CASANOVA, M. A., GIORNO, F.A.C & FURTADO, A.L., "Programação em Lógica e a Linguagem Prolog", Ed. Edgard Blucher, 1987
 - Castrucci, B. - Introdução à Lógica Matemática - Nobel, 1952.
 - Gersting, J. L. - Fundamentos Matemáticos para Ciência da Computação - LTC, 1995. (4ª edição)
 - SWI-Prolog: <http://www.swi-prolog.org/>
 - Souza, João Nunes de, "Lógica para Ciência da Computação: Fundamentos de linguagem, semântica e sistemas de dedução", Ed. Campus, 2002.
 - Kandel, Abraham, "Fuzzy Mathematical Techniques with Applications", Addison-Wesley, 1986.

Plano de ensino

Curso: CCI-BAC - Bacharelado em Ciência da Computação

Turma: CCI122-01B - CIENCIA DA COMPUTAÇÃO FASE 01B

Disciplina: AGT0001 - ALGORITMOS

Período letivo: 2017/1

Carga horária: 72

Professor: 3639428 - EVERLIN FIGHERA COSTA MARQUES

Ementa

1. Noções de arquitetura e programação de computadores. Algoritmo, fluxograma e pseudo-codificação. Entrada e saída de dados. Constantes e variáveis. Operadores e expressões. Desvios e laços. Vetores e matrizes. Programação estruturada. Experimentação em linguagem de alto nível.

Objetivo geral

1. Capacitar os estudantes a definir soluções de problemas através do desenvolvimento de algoritmos a serem executados por computador.

Objetivo específico

1. Conhecer noções de arquitetura e programação de computadores.
Construir soluções para problemas computacionais tanto na forma de algoritmo, fluxograma ou pseudo-código, envolvendo comandos básicos, vetores e matrizes.
Implementar um problema em pseudocódigo em uma ferramenta de linguagem de programação de alto nível.

Conteúdo programático

1. 4 h
 1. Noções de arquitetura e programação de computadores.
 - 1.1. Conceitos de Hardware e componentes do computador
 - 1.2. Unidades de Entrada e Saída
 - 1.3. Organização e tipos de Memórias
 - 1.4. Unidade Central de Processamento
 - 1.5. Conceito de Software e tipos de Software
 - 1.6. Conceito de Algoritmo e formas de representação
2. 16 h
 2. Sintaxe Básica de Pseudocódigo
 - 2.1. Estrutura de um algoritmo
 - 2.2. Tipos de dados
 - 2.3. Variáveis e constantes
 - 2.4. Operadores e expressões
 - 2.5. Operador de atribuição
 - 2.6. Comandos de entrada e saída
 3. Desvios e Laços
 - 3.1. Estruturas de seleção
 - 3.1.1. Seleção simples: (SE...ENTÃO)
 - 3.1.2. Seleção composta: (SE...ENTÃO...SENÃO)
 - 3.1.3. Múltiplas escolhas: (CASO)
3. 16 h
 4. Estruturas de repetição
 - 4.1. Teste no início (ENQUANTO...FAÇA)
 - 4.2. Teste no fim (REPITA...ATÉ)
 - 4.3. Teste no início e variável de controle (PARA...FAÇA)
4. 16 h
 5. Vetores e Matrizes
 - 5.1. Vetores unidimensionais
 - 5.2. Manipulação de strings (vetores de caracteres)
 - 5.3. Vetores bidimensionais (matrizes)

5. 16 h

Plano de ensino

6. Procedimentos e Funções
6.1. Passagem de parâmetro (por valor e por referência)
6.2. Recursividade

6. 4 h

7. Experimentação em linguagem de alto nível
7.1. Introdução à linguagem C

Metodologia

1. Aulas expositivas e dialogadas em sala
Aulas práticas em laboratório de informática
Uso de software para escrever, compilar e testar os algoritmos criados nas aulas

Obs: até 20% (vinte por cento) da carga horária poderá ser desenvolvida à distância com apoio do Ambiente Virtual de Aprendizagem Moodle, disponível na Universidade.

Sistema de avaliação

1. Do desempenho do discente:
A qualidade do desempenho do estudante será avaliada com base no desenvolvimento das seguintes atividades e com os seguintes critérios:
Prova 1 (escrita) = 20%
Prova 2 (escrita) = 25%
Prova 3 (escrita) = 25%
Exercícios avaliativos (escritos e práticos) = 10%
Trabalho final (prático) = 20%

Do desempenho da disciplina e do professor:
Os discentes terão, igualmente, a oportunidade de fazer uma avaliação mais completa do desempenho do professor e da disciplina através do sistema de avaliação (SIGA).

Bibliografia básica

1. DE OLIVEIRA, J.F.; MANZANO, José Augusto N. G. Algoritmos: Lógica para Desenvolvimento de Programação de Computadores. Editora Érica, 16a ed., 2004. ISBN 857194718X.
DE SOUZA, M.A.F., GOMES, M.M., SOARES, M.V., CONCILIO, R. Algoritmos e Lógica de Programação. Thomson Learning, 2004. ISBN 8522104646.
MEDINA, M., FERTIG, C. Algoritmos e Programação - Teoria e Prática. Novatec, 2005. ISBN 857522073X.

Bibliografia complementar

1. CORMEN, Thomas H. Algoritmos: teoria e prática. Rio de Janeiro: Campus, 2002. 916 p. ISBN 8535209263
DEITEL, Harvey M.; DEITEL, Paul J; KURBAN, Amir. Como programar em C. 2.ed. Rio de Janeiro: Livros Técnicos e Científicos, c1999. 486 p. ISBN 8521611919
GUIMARÃES, Angelo de Moura. Introdução à ciência da computação. Rio de Janeiro: Livros Técnicos e Científicos, 1984.
Instituto Brasileiro de Pesquisa em Informática. Dominando a linguagem C. Rio de Janeiro: IBPI, c1993. 236 p. ; ISBN 8585588012
RAMALHO, José Antônio. Introdução à informática: teoria e prática. São Paulo: Berkeley, 2000.
Artigos/materiais fornecidos pelo professor.

Plano de ensino

Curso: CCI-BAC - Bacharelado em Ciência da Computação

Turma: CCI122-01B - CIENCIA DA COMPUTAÇÃO FASE 01B

Disciplina: LMA0001 - LÓGICA MATEMÁTICA

Período letivo: 2017/1

Carga horária: 72

Professor: 3092798 - CLAUDIO CESAR DE SA

Ementa

1. História da lógica. Cálculo proposicional. Fórmulas tautológicas, contra-válidas e consistentes. Transformação entre conectivos lógicos. Equivalências. Argumentos válidos. Teorema lógico. Axiomatização. Metodos de prova. Lógica de 1a. Ordem (LPO). Quantificadores. Fórmulas. Argumentos. Axiomatização. Noções sobre teorias lógicas (completude e corretude). Notação clausal. Introdução à Provas de LPO. Uso de uma linguagem baseada em lógica, como instância da LPO. Axiomatização. Noções sobre teorias lógicas (completude e corretude). Notação clausal. Introdução à Provas de LPO. Uso de uma linguagem baseada em lógica, como instância da LPO.

Objetivo geral

1. Apresentar os conceitos das lógicas proposicionais e de primeira ordem, provas de teoremas lógicos, e uma ferramenta derivada da lógica. Esta como um paradigma de programação. Desenvolver o raciocínio lógico-matemático e uma mentalidade alicerçada no rigor e na observação. Adquirir uma formação dedutiva e indutiva para efetuar pesquisas mais profundas principalmente na Matemática.

Objetivo específico

1. Programa da Disciplina:
 1. Histórico, Motivação
 - 1.1. Introdução contextual: Ciência e lógica; Lógica e razão
 - 1.2. Definição de Validade Lógica
 - 1.3. Argumento.
 - 1.4. Definição dos conectivos, silogismo (disjuntivo), exemplos
 - 1.5. Conectivos da negação, conjunção, disjunção, implicação e bi-implicação, tabelas-verdade
 - 1.6. Verdades e Falácias
 - 1.7. Paradoxos semânticos e lógicos.
 2. Introdução as Provas Lógicas
 - 2.1. Tautologias e contradições
 - 2.2. Forma normal disjuntiva e conjuntiva
 - 2.3. Uma axiomatização ao cálculo proposicional
 - 2.4. Teoria da Dedução Natural: Conceitos sintáticos:
 - 2.5. Lógica como um Sistema Formal
 - 2.6. Regras Derivadas, Teoremas
 - 2.7. Equivalências e Implicações Lógicas
 - 2.8. Teoremas, Dedução Natural com Regras de Inferências.
 3. Introdução à Lógica de Predicados de Primeira Ordem
 - 3.1. Conceitos de símbolos, constantes, funções, variáveis, interpretações, provas, exemplos.
 4. Introdução a Programação em Lógica.
 - 4.1. A linguagem Prolog e suas variações.
 - 4.2. Princípios de funcionamento da máquina Prolog.
 - 4.3. Construção de predicados para problemas clássicos.
 - 4.4. Abordagem de eficiência da execução.
 - 4.5. Recursividade.
 5. Aplicações em linguagem de programação lógica Prolog e a Lógica como uma ferramenta na Resolução de Problemas.
 6. Introdução a Lógica Nebulosa (Fuzzy)
 - 6.1. Conjuntos nebulosos. Funções nebulosas. Relações nebulosas. Lógica nebulosa
 - 6.2. Introdução a Programação com Lógica Nebulosa: FFL

Conteúdo programático

1. Apresentação da disciplina
Datas das provas

Plano de ensino

e Exame Final Ementa Processo de Avaliação Contexto da disciplina no curso Conteúdo do Curso
2. Conceitos de proposição Valores lógicos das proposições Definição de validade lógica Argumento
3. Introdução contextual: Ciência e lógica Historico da logica Aristotélica
4. Definição dos conectivos - 1a parte Definição dos conectivos, silogismo (disjuntivo), exemplos Conectivos da negação, conjunção, disjunção, Verdades e falácias Argumentos
5. Outros Conectivos Lógicos Conectivos da negação, conjunção, disjunção, implicação e bi-implicação, tabelas-verdade Paradoxos semânticos e lógicos Exemplos
6. Tabela-verdade de uma proposição composta Número de linhas de uma tabela-verdade Construção de tabela-verdade de uma proposição composta Exemplos
7. Tipos de Formulas Logicas Tautologias Contingência Contradição Exemplos
8. Definição de implicação lógica Propriedade da implicação lógica Exemplos Propriedade da equivalência lógica Exemplos
9. Tautologias e equivalência lógica Proposições associadas a uma condicional Negação conjunta de duas proposições Negação disjunta de duas proposições Negação da condicional Negação da bicondicional Dúvidas e exercícios
10. 1a Avaliação escrita
11. Forma normal; disjuntiva e conjuntiva Uma axiomatização ao cálculo proposicional Exemplos
12. Lógica como um sistema formal Regras derivadas, teoremas Exemplos
13. Regras de Derivação Regras de derivação e teoremas Da implicação ao teorema
14. Dedução natural com regras Teoremas, dedução natural com regras de inferências. Esquemas de provas
15. Prova Direta Prova direta ou via dedução natural da LPO
16. Prova Indireta Exemplos
17. Prova por Implicação Indireta Prova por Implicação Indireta $x \dots y \rightarrow a \rightarrow b$ logo $a \dots y \dots a \rightarrow b$ Exemplos
18. Método da Resolução para LPO Resolvente Literal Exemplos

Plano de ensino

19. Lógica Primeira Ordem (LPO) Definições da LPO Exemplos
20. Quantificadores da LPO Quantificador existencial Quantificador universal Sentenças abertas com uma variável Conjunto-verdade de uma sentença aberta com uma variável Sentenças com duas variáveis Sentenças abertas com n variáveis Sentenças com duas variáveis Conjunto-verdade de uma sentença aberta
21. Equivalência da NEGACAO de quantificadores Quantificador de existência e unicidade Variável aparente/ligada e variável livre Negação de proposições com quantificadores Exemplos
22. Transformação Clausal (TC) 10 passos da TC Exemplos
23. Exercícios de LPO TC Comutatividade dos quantificadores Quantificação múltipla Quantificação parcial
24. Conexão LPO x PICAT A linguagem PICAT Exemplos Conexão com LPO
25. Exemplos de PICAT Laboratório
26. 2a. Avaliação
27. Revisão e dúvidas Exercícios
28. Exame Final
29. Resolução de Exercícios do conteúdo corrente
30. Revisão de conteúdo Revisão de conteúdo Exercícios
31. Resolução da Prova Dúvidas
32. Sistemas Dedutivos Motivação Formalismos
33. Método da Resolução em LP A Resolução em LP Exercícios
34. Método da Resolução em LPO Premissas, definições, resolvente, unificação, árvore de solução cláusula vazia a contradição
35. Transformação Clausal - LP Transformação de fórmulas em cláusulas em LP Exercícios
36. Exemplos do conteúdo corrente
37. Exercícios resolvidos e propostos Discussão Avaliação do aprendizado

Plano de ensino

38. Laboratório Experimentos de sala de aula Diretamente ao laboratório Prática em ação
39. Prova da Disciplina Uma prova envolvendo todo conteúdo da disciplina
40. Laboratorio de PICAT Exemplos Projeto Final
41. Fundamentos de Prolog Conceitos de Prolog Instancia, variavel, casamento, sequencia de execucao, etc Exemplos
42. Estudo Individual Video-aulas no site do professor Exercicios individuais Duvidas individuais com o professor
43. Formas Normais FNC FND Exemplos
44. Equivalencias entre Conectivos Substituicoes imediatas Exemplos
45. Particularizacoes UNIVERSAIS -- EXISTENCIAIS Generalizacoes UNIVERSAIS -- EXISTENCIAIS Exemplos
46. Regras de Inferencia na LPO Atomos Linguagem anotada Exemplos
47. Prova Todo conteudo do curso é avaliado aqui
48. Acompanhamento do projeto final Projeto Final Duvidas dos alunos Atendimento presencial e remoto
49. Álgebra das Proposições Exemplos
50. Relações Lógicas Relações Matemáticas Relações Lógicas Exemplos Conexão das relações com a lógica
51. Teorema Lógico Conceito de Teorema Lógico Definição a partir da relação de Equivalência Exemplos
52. A Linguagem de Primeira Ordem A Linguagem de Primeira Ordem; alfabeto, conectivos, regras, predicados, funtores Exemplos
53. Interpretação de Fórmulas em LPO Exemplos Exercícios
54. Quantificadores da LPO Funções Interpretações Exemplos
55. Equivalências na LPO Equivalências de quantificadores Exemplos
56. Introdução a Programação em Lógica. A linguagem PICAT e suas variações. Princípios de

Plano de ensino

funcionamento da máquina PICAT. Construção de predicados para problemas clássicos. Abordagem de eficiência da execução. Recursividade.
57. Exercícios pertinentes ao tópico no momento
58. Generalizações e Particularizações - Existencial e Universal Exercícios
59. PICAT - Video-aulas
60. 3a Avaliação -- Prova
61. Resolução da Prova
62. Revisão e Dúvidas dos alunos com conteúdo
63. Laboratório -- Projeto Final

Metodologia

1. Conteúdos expostos pelo professor com participação colaborativa dos estudantes, acompanhados pela proposição de exercícios práticos associados. Implementação de problemas clássicos da área. O conteúdo da disciplina poderá ser ministrado na modalidade de ensino a distância em até 20% do total de sua Carga Horária (MEC PORTARIA Nº 4.059, DE 10 DE DEZEMBRO DE 2004 publicado no DOU de 13/12/2004, Seção 1, p. 34).

Sistema de avaliação

1. Os estudantes serão avaliados com base no desempenho em avaliações escritas, exercícios de implementação e participação nas aulas e em eventos, conceituados de 0,0 a 10,0. Conforme regulamento da UDESC, o comparecimento às aulas deve ser de no mínimo 75%.
Ao longo do semestre, serão realizadas as seguintes avaliações:
 - 3 avaliações escritas com pesos iguais, provendo um total de 90% da MS;
 - 01 um laboratório: 10%A média geral ou semestral, MS, portanto, será obtida por meio da seguinte fórmula:
$$MS = 0.30 \times P1 + 0.30 \times P2 + 0.30 \times P3 + 0,10 \times Lab$$

Exame
Caso o discente não obtenha média M igual ou superior a 7,0, um exame escrito será aplicado objetivando aferir o conhecimento teórico do estudante.
Não há recuperação das provas por não comparecimento, exceto nos casos previstos no regulamento da UDESC.

Bibliografia básica

1. Guilherme Bittencourt. Inteligência Artificial - Ferramentas e Teorias, Editora da Unicamp, 10a. Escola de Computação, Instituto de Computação - UNICAMP, 1996.

ABE, Jair M. SCALZITTI, Alexandre. SILVA FILHO, João Inácio Introdução à lógica matemática para a Ciência da computação, São Paulo: Arte Ciência, 2001

Alencar Filho, E. - Iniciação à Lógica Matemática - Nobel, 1984.

Bibliografia complementar

1. CASANOVA, M. A., GIORNO, F.A.C & FURTADO, A.L., "Programação em Lógica e a Linguagem Prolog", Ed. Edgard Blucher, 1987

Castrucci, B. - Introdução à Lógica Matemática - Nobel, 1952.

Gersting, J. L. - Fundamentos Matemáticos para Ciência da Computação - LTC, 1995. (4ª edição)

SWI-Prolog: <http://www.swi-prolog.org/>

Souza, João Nunes de, "Lógica para Ciência da Computação: Fundamentos de linguagem, semântica e sistemas de dedução", Ed. Campus, 2002.

Kandel, Abraham, "Fuzzy Mathematical Techniques with Applications", Addison-Wesley, 1986.

Plano de ensino

Curso: CCI-BAC - Bacharelado em Ciência da Computação

Turma: CCI122-01C - CIENCIA DA COMPUTAÇÃO FASE 01C

Disciplina: AGT0001 - ALGORITMOS

Período letivo: 2017/1

Carga horária: 72

Professor: 3639428 - EVERLIN FIGHERA COSTA MARQUES

Ementa

1. Noções de arquitetura e programação de computadores. Algoritmo, fluxograma e pseudo-codificação. Entrada e saída de dados. Constantes e variáveis. Operadores e expressões. Desvios e laços. Vetores e matrizes. Programação estruturada. Experimentação em linguagem de alto nível.

Objetivo geral

1. Capacitar os estudantes a definir soluções de problemas através do desenvolvimento de algoritmos a serem executados por computador.

Objetivo específico

1. Conhecer noções de arquitetura e programação de computadores.
Construir soluções para problemas computacionais tanto na forma de algoritmo, fluxograma ou pseudo-código, envolvendo comandos básicos, vetores e matrizes.
Implementar um problema em pseudocódigo em uma ferramenta de linguagem de programação de alto nível.

Conteúdo programático

1. 4 h
 1. Noções de arquitetura e programação de computadores.
 - 1.1. Conceitos de Hardware e componentes do computador
 - 1.2. Unidades de Entrada e Saída
 - 1.3. Organização e tipos de Memórias
 - 1.4. Unidade Central de Processamento
 - 1.5. Conceito de Software e tipos de Software
 - 1.6. Conceito de Algoritmo e formas de representação
2. 16 h
 2. Sintaxe Básica de Pseudocódigo
 - 2.1. Estrutura de um algoritmo
 - 2.2. Tipos de dados
 - 2.3. Variáveis e constantes
 - 2.4. Operadores e expressões
 - 2.5. Operador de atribuição
 - 2.6. Comandos de entrada e saída
 3. Desvios e Laços
 - 3.1. Estruturas de seleção
 - 3.1.1. Seleção simples: (SE...ENTÃO)
 - 3.1.2. Seleção composta: (SE...ENTÃO...SENÃO)
 - 3.1.3. Múltiplas escolhas: (CASO)
3. 16 h
 4. Estruturas de repetição
 - 4.1. Teste no início (ENQUANTO...FAÇA)
 - 4.2. Teste no fim (REPITA...ATÉ)
 - 4.3. Teste no início e variável de controle (PARA...FAÇA)
4. 16 h
 5. Vetores e Matrizes
 - 5.1. Vetores unidimensionais
 - 5.2. Manipulação de strings (vetores de caracteres)
 - 5.3. Vetores bidimensionais (matrizes)

5. 16 h

Plano de ensino

6. Procedimentos e Funções
6.1. Passagem de parâmetro (por valor e por referência)
6.2. Recursividade

6. 4 h

7. Experimentação em linguagem de alto nível
7.1. Introdução à linguagem C

Metodologia

1. Aulas expositivas e dialogadas em sala
Aulas práticas em laboratório de informática
Uso de software para escrever, compilar e testar os algoritmos criados nas aulas

Obs: até 20% (vinte por cento) da carga horária poderá ser desenvolvida à distância com apoio do Ambiente Virtual de Aprendizagem Moodle, disponível na Universidade.

Sistema de avaliação

1. Do desempenho do discente:
A qualidade do desempenho do estudante será avaliada com base no desenvolvimento das seguintes atividades e com os seguintes critérios:
Prova 1 (escrita) = 20%
Prova 2 (escrita) = 25%
Prova 3 (escrita) = 25%
Exercícios avaliativos (escritos e práticos) = 10%
Trabalho final (prático) = 20%

Do desempenho da disciplina e do professor:
Os discentes terão, igualmente, a oportunidade de fazer uma avaliação mais completa do desempenho do professor e da disciplina através do sistema de avaliação (SIGA).

Bibliografia básica

1. DE OLIVEIRA, J.F.; MANZANO, José Augusto N. G. Algoritmos: Lógica para Desenvolvimento de Programação de Computadores. Editora Érica, 16a ed., 2004. ISBN 857194718X.
DE SOUZA, M.A.F., GOMES, M.M., SOARES, M.V., CONCILIO, R. Algoritmos e Lógica de Programação. Thomson Learning, 2004. ISBN 8522104646.
MEDINA, M., FERTIG, C. Algoritmos e Programação - Teoria e Prática. Novatec, 2005. ISBN 857522073X.

Bibliografia complementar

1. CORMEN, Thomas H. Algoritmos: teoria e prática. Rio de Janeiro: Campus, 2002. 916 p. ISBN 8535209263
DEITEL, Harvey M.; DEITEL, Paul J; KURBAN, Amir. Como programar em C. 2.ed. Rio de Janeiro: Livros Técnicos e Científicos, c1999. 486 p. ISBN 8521611919
GUIMARÃES, Angelo de Moura. Introdução à ciência da computação. Rio de Janeiro: Livros Técnicos e Científicos, 1984.
Instituto Brasileiro de Pesquisa em Informática. Dominando a linguagem C. Rio de Janeiro: IBPI, c1993. 236 p. ; ISBN 8585588012
RAMALHO, José Antônio. Introdução à informática: teoria e prática. São Paulo: Berkeley, 2000.
Artigos/materiais fornecidos pelo professor.

Plano de ensino

Curso: CCI-BAC - Bacharelado em Ciência da Computação

Turma: CCI122-01U - CIENCIA DA COMPUTAÇÃO FASE 01U

Disciplina: ALG1002 - ÁLGEBRA LINEAR E GEOMETRIA ANALÍTICA I

Período letivo: 2017/1

Carga horária: 72

Professor: 9549803 - Maisa Damazio Franco

Ementa

1. Vetores no R2 e R3. Produto escalar. Produto vetorial. Duplo produto vetorial e misto. Retas e planos no R3. Transformação de coordenadas no R2. Coordenadas polares cilíndricas e esféricas no R2 e no R3. Curvas e superfícies.

Objetivo geral

1. Proporcionar ao estudante a oportunidade de adquirir conhecimentos de Geometria Analítica e aplicá-los em sua área de atuação.

Objetivo específico

1. Familiarizar o aluno com vetores e suas operações e torná-lo apto a utilizá-los em problemas práticos.
Aplicar esses conceitos no estudo de reta e de plano.
Conhecer e aplicar transformação de coordenadas no R2. Conhecer os sistemas de coordenadas polares, cilíndricas e esféricas.
Tornar o aluno apto a fazer uma análise das cônicas e quádras.
O aluno deverá estar apto a representar curvas no espaço.

Conteúdo programático

1. 1. Vetores no R3
 - 1.1. Vetores no R3 - representação
 - 1.2. Alguns tipos de vetores
 - 1.3. Operações com vetores
 - 1.4. Adição e propriedades
 - 1.5. Subtração
 - 1.6. Produto por escalar e propriedades
 - 1.7. Base ortonormal
 - 1.8. Norma de um vetor
 - 1.9. Cálculo de versor
 - 1.10. Condição de paralelismo entre dois vetores
 - 1.11. Exercícios e aplicações
 - 1.12. Produto escalar
 - 1.13. Definição
 - 1.14. Propriedades
 - 1.15. Expressão cartesiana
 - 1.16. Interpretação geométrica
 - 1.17. Cossenos diretores de um vetor
 - 1.18. Exercícios e aplicações
 - 1.19. Produto vetorial
 - 1.20. Definição
 - 1.21. Propriedades
 - 1.22. Expressão cartesiana
 - 1.23. Interpretação geométrica
 - 1.24. Duplo produto vetorial
 - 1.25. Interpretação geométrica
 - 1.26. Exercícios e aplicações
 - 1.27. Produto misto
 - 1.28. Definição
 - 1.29. Expressão cartesiana
 - 1.30. Propriedades
2. 2. Retas no R3
 - 2.1. Interpretação geométrica
 - 2.2. Equação vetorial da reta
 - 2.3. Equação paramétrica da reta
 - 2.4. Equação simétrica da reta e equações reduzidas
 - 2.5. Equação da reta determinada por dois pontos distintos
 - 2.6. Cossenos diretores de uma reta
 - 2.7. Posição relativa entre as retas
 - 2.8. Paralelismo e perpendicularismo de retas

Plano de ensino

2.9. Ângulo entre duas retas 2.10. Interseção de retas.
3. 3. Planos no R3 3.1. Definição 3.2. Equação vetorial do plano 3.3. Equação paramétrica do plano 3.4. Equação geral do plano 3.5. Vetor normal do plano 3.6. Posições relativas de dois planos 3.7. Posições relativas de um plano a uma reta 3.8. Ângulo entre planos.
4. 4. Distâncias 4.1. Distância entre dois pontos 4.2. Distância de um ponto a uma reta 4.3. Distância entre duas retas 4.4. Distância de um ponto a um plano 4.5. Distância entre dois planos 4.6. Distância de uma reta a um plano.
5. 5. Estudo das Cônicas 5.1. Parábola 5.1.1. Definição 5.1.2. Elementos da parábola 5.1.3. Equações d parábola 5.2. Elipse 5.2.1. Definição 5.2.2. Elementos da elipse 5.2.3. Equações da elipse 5.3. Hipérbole 5.3.1. Definição 5.3.2. Elementos da hipérbole 5.3.3. Equações da hipérbole 5.3.4. Hipérboles equiláteras 5.4. Sistemas de coordenadas no R2. 5.5.1. Coordenadas polares 5.5.2. Transformação de coordenadas 5.5.3. Curvas polares 5.6 Parametrização de curvas.
6. 6. Estudo de Quádricas 6.1 Superfícies Centradas 6.1.1. Elipsóide 6.1.2. Hiperbolóide de uma folha 6.1.3. Hiperbolóide de duas folhas 6.2. Superfícies não Centradas 6.2.1. Parabolóide Elíptico 6.2.2. Parabolóide Hiperbólico 6.3. Superfície Conica 6.4. Superfície Cilíndrica 6.5. Sistemas de coordenadas em R3 6.5.1. Coordenadas Cilíndricas 6.5.2. Coordenadas Esféricas 6.5.3. Mudança de Coordenadas 6.6. Parametrização de superfícies

Metodologia

1. Aulas expositivas e dialogadas com resolução de exercícios orientados. Utilização de ferramentas tecnológicas. Atendimento individual ao aluno.
--

Sistema de avaliação

1. Quatro avaliações escritas individuais durante o semestre letivo, com pesos iguais

Bibliografia básica

1. 1. Steinbruch, A. e Winterle, P., "Geometria Analítica". Makron Books Editora. 2ª edição. 1987. 2. Venturi, J.J.. "Álgebra Vetorial e Geometria Analítica". Autores Paranaenses, 2009. Disponível, em http://geometriaa.dominiotemporario.com/livros/av.pdf sob licença do autor. 3. Venturi, J.J.. "Cônicas e Quádricas". Autores Paranaenses, 2003. Disponível, em http://geometriaa.dominiotemporario.com/livros/cq.pdf sob licença do autor
--

Plano de ensino

4. Boulos, P e Camargo, I., ``Geometria Analítica. Um Tratamento Vetorial". Makron Books Editora. 1990.

Bibliografia complementar

1. Boulos, P. e Camargo, I., "Introdução à Geometria Analítica no Espaço". Makron Books Editora. 1997.
2. Leithold, L.. O Cálculo com Geometria Analítica. Editora Harbra. 3ª edição. Volume 1. 1994.
3. Lehmann, C.H.. "Geometria Analítica". Editora Globo. 1982.

Plano de ensino

Curso: CCI-BAC - Bacharelado em Ciência da Computação

Turma: CCI122-01U - CIENCIA DA COMPUTAÇÃO FASE 01U

Disciplina: CDI1001 - CÁLCULO DIFERENCIAL E INTEGRAL I

Período letivo: 2017/1

Carga horária: 108

Professor: 9000176 - DEBORA ELOISA NASS KIECKHOEFEL

Ementa

1. Números, variáveis e funções de uma variável real. Limite e continuidade de função. Derivada diferencial. Teoremas sobre funções deriváveis. Análise das funções. Integral indefinida.

Objetivo geral

1. Desenvolver a capacidade de raciocínio crítico, lógico e dedutivo, utilizado no estudo do desenvolvimento e sua variação, tendo como objetivo deste estudo, as funções.

Objetivo específico

1. O aluno deverá ser capaz de:
 - ? Operar com equações e inequações com e sem valor absoluto.
 - ? Determinar o domínio de uma função.
 - ? Operar com funções.
 - ? Interpretar geometricamente a definição de limite.
 - ? Calcular limites de uma função usando limites notáveis e as regras de L' Hôpital.
 - ? Determinar se a função é contínua.
 - ? Derivar qualquer função.
 - ? Interpretar geométrica e fisicamente derivadas e diferenciais.
 - ? Resolver problemas com diferenciais.
 - ? Analisar a variação das funções e construir seus gráficos.
 - ? Resolver problemas utilizando técnicas de integração.

Conteúdo programático

1. 1. Números, variáveis e funções de uma variável real.
2. 1.1 Números;
3. 1.2 Desigualdades;
4. 1.3 Intervalos;
5. 1.4 Valor Absoluto;
6. 1.5 Função.
7. 2. Limite e Continuidade da função
8. 2.1 Limite de uma variável;
9. 2.2 Limite de uma função;
10. 2.3 Propriedades de limites;
11. 2.4 Cálculo de limites;
12. 2.5 Limites notáveis;
13. 2.6 Continuidade de uma função;
14. 2.7 Continuidade em intervalos;
15. 2.8 Propriedades das funções contínuas;
16. 2.9 Teorema do valor intermediário.
17. 3. Derivada
18. 3.1 Introdução à derivada;
19. 3.2 Reta tangente;
20. 3.3 Derivadas;

Plano de ensino

21. 3.4 Diferenciabilidade;
22. 3.5 Derivadas laterais;
23. 3.6 Regras de derivação;
24. 3.7 Derivação implícita;
25. 3.8 Derivada da função inversa;
26. 3.9 Derivada de uma função na forma paramétrica;
27. 3.10 Derivadas de ordem superior;
28. 3.11 Diferenciais e aproximação linear local;
29. 3.12 Interpretação mecânica da derivada;
30. 3.13 Taxa de variação e taxa relacionada
31. 4. Regra de L'Hôpital.
32. 5. Análise de Variação das Funções
33. 5.1 Funções crescentes e decrescentes;
34. 5.2 Máximos e mínimos;
35. 5.3 Teoremas sobre derivadas;
36. 5.4 Critérios para determinação dos extremos de uma função;
37. 5.5 Crescimento e decrescimento de uma função;
38. 5.6 Pontos de inflexão e concavidade;
39. 5.7 Assíntotas do gráfico de uma função;
40. 5.8 Aplicações da teoria dos máximos e mínimos de funções na solução de problemas.
41. 6. Integrais.
42. 6.1 Introdução à integral indefinida;
43. 6.2 Propriedades de integral indefinida;
44. 6.3 Tabela de integrais imediatas;
45. 6.4 Integração por substituição;
46. 6.5 Integração por partes;
47. 6.6 Integração de funções trigonométricas;
48. 6.7 Integrais por substituição trigonométrica;
49. 6.8 Integrais elementares que contém um trinômio quadrado;
50. 6.9 Integração de funções racionais por frações parciais;

Metodologia

1. Aulas expositivas e dialogadas com resolução de exercícios orientados. Atendimento individual ao aluno.
--

Sistema de avaliação

1. A média semestral será calculada com base na nota obtida nas 5 avaliações escritas individuais realizadas durante o semestre letivo, conforme a fórmula: $(1,25 \cdot P1 + 1,25 \cdot P2 + 2,5 \cdot P3 + 2,5 \cdot P4 + 2,5 \cdot P5) / 10$
--

Bibliografia básica

1. ANTON, H. Cálculo, um novo horizonte. Porto Alegre: Bookman, vol. 1, 6ª ed., 2000. FLEMING, D. M.; GONÇALVES, M. B. Cálculo A. São Paulo: Pearson Prentice Hall, 6ª ed. rev. e ampl., 2006. STEWART, J. Cálculo. São Paulo. Cengage Learning, vol. 1, 6ª ed, 2009.

Bibliografia complementar

Plano de ensino

1. KÜHLKAMP, N. Cálculo 1. Florianópolis. Editora UFSC, 3ª ed. rev. e ampl. 2006.
LEITHOLD, L. O Cálculo com Geometria Analítica. São Paulo. Editora HARBRA Ltda, 3ª ed., 1994.
SWOKOWSKI, E. W. Cálculo com Geometria Analítica. São Paulo. Makron Books Ltda, 2ª ed., 1994.
THOMAS, G. E. Cálculo. São Paulo. Pearson Addison Wesley, São Paulo, vol. 1, 10ª ed, 2002.
GUIDORIZZI, Hamilton Luiz. Um curso de cálculo. Volume 1. 5. ed. Rio de Janeiro: Livros Técnicos e Científicos, 2002.

Plano de ensino

Curso: CCI-BAC - Bacharelado em Ciência da Computação

Turma: CCI122-01U - CIENCIA DA COMPUTAÇÃO FASE 01U

Disciplina: MCI0001 - METODOLOGIA CIENTIFICA

Período letivo: 2017/1

Carga horária: 36

Professor: 9630970 - Maristela de Souza Borba Severino

Ementa

1. Contexto universitário; Diretrizes para a Leitura, Análise e Interpretação de Textos; Tipos de comunicação técnico-científicas (relatórios - manual, trabalho de conclusão de curso, dissertação, tese -, artigos, resenhas, resumo); Normatização do documento científico (NBR, SBC, IEEE, ACM); Elementos de informação (NBR: referências, figuras, tabelas, quadros, referência indireta e extensa - plágio); Elaboração e aplicação de modelos (template) de documentos técnico-científicos usando processador/editor de texto.

Objetivo geral

1. Oportunizar aos acadêmicos conhecimentos em metodologia científica necessários à leitura, interpretação, estruturação e produção de trabalhos científicos.

Objetivo específico

1. Proporcionar aos acadêmicos a construção de conhecimento em metodologia científica para que cada estudante esteja apto a
 - conhecer o contexto universitário em relação a ensino, pesquisa e extensão;
 - ler, analisar e interpretar textos para desenvolver autoria crítica;
 - conhecer os tipos de comunicação técnico-científicas: sumarização e resumo, resenha, artigos, papers, trabalho de conclusão de curso, relatório, dissertação, tese;
 - conhecer as normas contidas no manual para elaboração de trabalhos acadêmicos da UDESC, padrão ABNT em relação a formatações/produções de elementos pré-textuais, textuais, pós-textuais, páginas, títulos, figuras, tabelas, quadros, citações, referências, referência indireta e extensa;
 - refletir acerca da relação entre autoria, citações, plágio e autoplagio;
 - utilizar as normatizações da ABNT, SBC, IEEE, ACM;
 - elaborar e aplicar templates de documentos técnico-científicos por meio de processador/editor de texto nas produções acadêmicas;
 - apresentar em equipe, oralmente e com recursos, seminários durante o semestre letivo.

Aulas em 2017-1: segunda-feira das 19:00 às 20:40 - duração da hora-aula = 50 minutos

Fevereiro: 22

Março: 08, 15, 22, 29

Abril: 05, 12, 19, 26

Maior: 03, 10, 17, 24, 31

Junho: 07, 14, 21, 28

Exame final: 05/07/2017

Conteúdo programático

1. A ciência e o método científico no contexto universitário de produção científica: ciência, tecnologia e metodologia científica no ensino, pesquisa e extensão universitária
2. Diretrizes para a leitura, análise e interpretação de textos
 - Análise textual
 - Análise temática
 - Análise interpretativa
 - Problemática e síntese pessoal
3. Comunicação técnico-científica: sumarização e resumo
4. Comunicação técnico-científica: resenha
5. Comunicação técnico-científica: artigo e paper
6. Comunicação técnico-científica: relatório e trabalho de conclusão de curso - TCC
7. Comunicação técnico-científica: dissertação e tese
8. Normatização do documento científico - elementos de informação: manual para elaboração de trabalhos Acadêmicos da UDESC, padrão ABNT (2016) - elementos pré-textuais, textuais, pós-textuais
9. Normatização do documento científico - elementos de informação: manual para elaboração de trabalhos Acadêmicos da UDESC, padrão ABNT (2016) - páginas e títulos
10. Normatização do documento científico - elementos de informação: manual para elaboração de trabalhos Acadêmicos da

Plano de ensino

UDESC, padrão ABNT (2016) - ilustrações: figuras, tabelas, quadros
11. Normatização do documento científico - elementos de informação: manual para elaboração de trabalhos Acadêmicos da UDESC, padrão ABNT (2016) - citação: autoria e referência; plágio e autoplágio
12. Normatização do documento científico - elementos de informação: manual para elaboração de trabalhos Acadêmicos da UDESC, padrão ABNT (2016) - citações
13. Normatização do documento científico - elementos de informação: manual para elaboração de trabalhos Acadêmicos da UDESC, padrão ABNT (2016) - referências
14. Normatização do documento científico: NBR, SBC, IEEE, ACM
15. Elaboração e aplicação de modelos (template) de documentos técnico-científicos usando processador/editor de texto

Metodologia

1. As aulas são expositivo-dialogadas por meio de leituras, debates, análises de vídeos curtos e textos científicos, seminários e com o uso dos recursos disponibilizados, tais como materiais impressos de textos verbais e não verbais, quadro, computador/datashow, som. As atividades escritas e orais são realizadas individualmente e em equipe. As avaliações escritas são individuais.
--

Sistema de avaliação

1. 1 - AV_1 - ATIV_1 - valor 10,0 (dez), peso 1,0 (um): trabalhos realizados nos meses março e abril/2017 em casa e/ou na sala de aula, escritos e/ou orais, segundo orientações informadas na sala de aula e/ou na página on line da professora no SIGA UDESC. A nota final é a média das notas obtidas nos trabalhos solicitados/realizados. A nota/feedback de cada trabalho será semanal, ou seja, na próxima aula e a nota final/média será lançada no SIGA UDESC até 12/04/2017. Todo trabalho deve ter o registro da data de entrega, cabeçalho ou capa. Não são considerados trabalhos fora dos prazos, enviados por e-mail, bem como plagiados parcial ou integralmente. 2 - AV_2 - PROVA_1 - valor 10,0 (dez), peso 2,0 (dois): avaliação escrita individual e sem consulta de todos os conteúdos trabalhados. Data: 19/04/2017. 3 - AV_3 - APRES_OR - valor 10,0 (dez), peso 1,0 (um): apresentações orais em equipes sobre trabalho de conclusão de curso. As atividades/apresentações serão realizadas em datas as quais serão estipuladas durante o curso, ou seja, nos meses maio e/ou junho/2017. Todos os acadêmicos devem participar igualmente das apresentações e usar recursos, tal como power point. A nota da equipe é única. 4 - AV_4 - ATIV_2 - valor 10,0 (dez), peso 2,0 (dois): trabalhos realizados nos meses maio e junho/ 2017 em casa e/ou na sala de aula, escritos e/ou orais, segundo orientações informadas na sala de aula e/ou na página on line da professora no SIGA UDESC. A nota final é a média das notas obtidas nos trabalhos solicitados/realizados. A nota/feedback de cada trabalho será semanal, ou seja, na próxima aula e a nota final/média será lançada no SIGA UDESC até 07/06/2017. Todo trabalho deve ter o registro da data de entrega, cabeçalho ou capa. Não são considerados trabalhos fora dos prazos, enviados por e-mail, bem como plagiados parcial ou integralmente. 5 - AV_5 - PROVA_2 - valor 10,0 (dez), peso 4,0 (quatro): avaliação escrita individual e sem consulta de todos os conteúdos trabalhados durante o semestre letivo. Data: 21/06/2017. Para ser aprovado sem exame final, o acadêmico(a) deve ter, pelo menos, 75% da frequência e nota/média semestral 7,0 (sete). Exame final: 05/07/2017 de todos os conteúdos trabalhados no curso. Média final para aprovação: 5,0 (cinco). Observações: 1 - o acadêmico(a) deve acessar este curso na página on line da professora no SIGA UDESC para ver/obter os materiais e avisos disponibilizados. Para isso ir em , depois em ensino, depois em professores, depois na letra M, depois no nome da professora ou ir em http://www.joinville.udesc.br/portal/professores/maristela/; 2 - parâmetros para a entrega de cada atividade: o texto deve estar digitado e impresso com letra tamanho 12, arial ou times new roman, cabeçalho ou capa conforme material 2; registrar a data de entrega, a(s) perguntas/enunciados e a(s) respostas; todo parágrafo deve estar justificado; o texto deve estar coeso - com as concordâncias adequadas à língua padrão, coerente - fazer sentido ao leitor, bem argumentado/fundamentado, com as pontuações adequadas e segundo as normas da ABNT e/ou outras normas solicitadas; o texto resposta deve apresentar a(s) referência(s) textual(is), ou seja, informações sobre as obras/autores citados no texto segundo normas ABNT, bem como seguir outras orientações disponibilizadas durante o semestre letivo nas aulas e/ou na página on line da professora no SIGA UDESC no material 3 (três); 3 - falta na data de qualquer avaliação ou atividade-avaliação deve ser justificada na secretaria acadêmica do curso do acadêmico e solicitada a segunda chamada dessa avaliação no prazo previsto para essa solicitação, segundo normativas UDESC, resoluções acadêmicas; 4 - informações sobre frequências, solicitações de prova em segunda chamada encontram-se no portal do acadêmico, ou seja, entrar em , depois ir em ensino, portal do acadêmico, resoluções acadêmicas.
--

Bibliografia básica

1. ALVES, Rubem. Entre a ciência e a sapiência: o dilema da educação. 5. ed. São Paulo: Loyola, 2001. BARROS, Aidil de Jesus Paes de; LEHFELD, Neide Aparecida de Souza. Projeto de pesquisa: propostas metodológicas. 9. ed. Petrópolis: Vozes, 2000. BASTOS, Cleverson Leite; KELLER, Vicente. Aprendendo a aprender. 15. ed. Petrópolis: Vozes, 2001.
--

Plano de ensino

Bibliografia complementar

1. APPOLINÁRIO, Fábio. Metodologia da ciência. São Paulo: Cengage Learning, 2011.
- ASSOCIAÇÃO BRASILEIRA DE NORMAS TÉCNICAS. NBR 10520: informação e documentação - apresentação de citação de documentos. Rio de Janeiro: ABNT, 2002.
- ASSOCIAÇÃO BRASILEIRA DE NORMAS TÉCNICAS. NBR 14724: informação e documentação - trabalhos acadêmicos - apresentação. Rio de Janeiro: ABNT, 2005.
- ASSOCIAÇÃO BRASILEIRA DE NORMAS TÉCNICAS. NBR 6023: informação e documentação - referências - elaboração. Rio de Janeiro: ABNT, 2002.
- ASSOCIAÇÃO BRASILEIRA DE NORMAS TÉCNICAS. NBR 6024: numerações progressivas das seções de um documento. Rio de Janeiro: ABNT, 2003.
- ASSOCIAÇÃO BRASILEIRA DE NORMAS TÉCNICAS. NBR 6027: sumário. Rio de Janeiro: ABNT, 2003.
- ASSOCIAÇÃO BRASILEIRA DE NORMAS TÉCNICAS. NBR 6028: resumos. Rio de Janeiro: ABNT, 2003.
- BARROS, Aidil de Jesus Paes de; LEHFELD, Neide Aparecida de Souza. Fundamentos de metodologia científica: um guia para a iniciação científica. 2. ed. São Paulo: Pearson Makron Books, 2000.
- CERVO, Amado; BERVIAN, Pedro A. Metodologia científica. 5. ed. São Paulo: Prentice Hall, 2002.
- CRUZ, Carla; RIBEIRO, Uirá. Metodologia científica: teoria e prática. Rio de Janeiro: Axl Books do Brasil, 2003.
- FACHIN, Odília. Fundamentos de metodologia. 4. ed. São Paulo: Saraiva, 2003.
- GIL, Antônio Carlos. Como elaborar projetos de pesquisa. São Paulo: Atlas, 1991.
- HEERDT, Mauri Luiz; LEONEL, Vilson. Metodologia científica e da pesquisa: livro didático. Palhoça: UnisulVirtual, 2007. Disponível em: [/www.unisul.br](http://www.unisul.br). Acesso em: 8 fev. 2017.
- JUNCKES, Rosane Santana; OLIVEIRA, Nádia Fátima de. Multiculturalismo: metodologia da pesquisa científica. São Paulo: Know How, 2010.
- KÖCHE, José Carlos. Fundamentos da metodologia científica: teoria da ciência e prática da pesquisa. 14. ed. Petrópolis, RJ: Vozes, 1997.
- MARCONI, Marina de Andrade; LAKATOS, Eva Maria. Fundamentos da metodologia científica. 7. ed. São Paulo: Atlas, 2010.
- NERY, Guilherme et al. Nem tudo que parece é: entenda o que é plágio. Niterói, RJ: [Universidade Federal Fluminense, 2010?]. Cartilha. Disponível em: [/www.noticias.uff.br/arquivos/cartilha-sobre-plagio-academico.pdf](http://www.noticias.uff.br/arquivos/cartilha-sobre-plagio-academico.pdf). Acesso em: 8 fev. 2017.
- PINHEIRO, José Maurício dos Santos. Da iniciação científica ao TCC: uma abordagem para os cursos de tecnologia. Rio de Janeiro: Ciência Moderna Ltda., 2010.
- RUIZ, João Álvaro. Metodologia científica: guia para eficiência nos estudos. 5. ed. São Paulo: Atlas, 2002.
- SALOMON, Délcio Vieira. Como fazer uma monografia. 12. ed. São Paulo: Saraiva, 2010.
- SEVERINO, Antônio Joaquim. Metodologia do trabalho científico. 19. ed. São Paulo: Cortez, 1993.
- SILVA, Everaldo da et al. Metodologia do trabalho acadêmico. 3. ed. Curitiba: Juruá, 2010.
- UNIVERSIDADE DO ESTADO DE SANTA CATARINA. Manual para elaboração e normalização de trabalhos de conclusão de curso da UDESC: graduação e pós-graduação. Florianópolis: UDESC, 2016. Disponível em: [/www.cct.udesc.br/arquivos/id_submenu/2123/manual_de_trabalhos_academicos_da_udesc.pdf](http://www.cct.udesc.br/arquivos/id_submenu/2123/manual_de_trabalhos_academicos_da_udesc.pdf). Acesso em: 8 fev. 2017.
- RUIZ, João Álvaro. Metodologia científica: guia para eficiência nos estudos. 5. ed. São Paulo: Atlas, 2002.

Plano de ensino

Curso: CCI-BAC - Bacharelado em Ciência da Computação

Turma: CCI122-01U - CIENCIA DA COMPUTAÇÃO FASE 01U

Disciplina: TGS0001 - TEORIA GERAL DE SISTEMAS

Período letivo: 2017/1

Carga horária: 72

Professor: 2511223 - CLAUDIOMIR SELNER

Ementa

1. Introdução à Epistemologia. Visão Geral da Filosofia da Ciência. Histórico da TGS. Conceitos fundamentais da TGS. Características dos Sistemas. Classificações dos Sistemas. Cibernética. Desdobramentos atuais sobre TGS.

Objetivo geral

1. Contribuir para o desenvolvimento da consciência de que a natureza da ciência é efêmera, levando os alunos a compreenderem a proposta da Teoria Geral dos Sistemas a partir dessa consciência, proposta essa que é uma tentativa de unificação da forma de se perceber a realidade nas diversas expressões do saber científico.

Objetivo específico

1. - Compreender o conceito de "verdade"
- Compreender a complementaridade entre filosofia e ciência
- Compreender o jeito de pensar científico (a "filosofia" da ciência)
- Compreender o limite da ciência
- Entender o que é "sistema"
- Entender como a TGS alarga as fronteiras (diminui os limites) da ciência
- Entender a correlação entre TGS e Cibernética
- Compreender as contribuições atuais da TGS para o desenvolvimento de software
- Propiciar as condições para o aprendizado da Análise dos Sistemas

Conteúdo programático

1. Avaliação diagnóstica
- Identificação do conhecimento atual dos alunos sobre TGS
2. Introdução à Epistemologia
Estudo das teorias e princípios, busca pela verdade absolutamente certa (episteme), causalidade (Demócrito e Aristóteles), finalidade (Anaxágoras e Aristóteles), teoria como "óculos" para a realidade (Galileu, Kant, Einstein, Heisenberg, Morin), construção social da realidade, percepção da realidade, paradigma científico, rompimento epistemológico, causalidade e complementaridade (Bohr, Heisenberg, Schrödinger, Dirac).
3. Filosofia da Ciência
Visão geral, proposição e limites da ciência.
4. Histórico da TGS
Origem, propósito, significado e proposta da TGS dentro da filosofia da ciência.
5. Conceitos fundamentais da TGS
Conceito de sistemas, concepções cartesiana e mecanicista X enfoque sistêmico, proposta complementar ao princípio da causalidade (mecanicismo clássico) e ao método analítico cartesiano, super-sistema, sistema e subsistema.
6. Características dos Sistemas
Retroação, input/output de energia, entropia X entropia negativa, eqüifinalidade, endocausalidade, retroação, homeostase e estabilidade, diferenciação, autopoiesis, auto-referência, modelo de informação isomórfico ao da entropia negativa.
7. Classificações dos Sistemas
Sistemas fechados, sistemas abertos, sistemas psico-sociais, sistemas biológicos, sistemas sociais (tipos primitivos X organizações sociais), sistemas mecânicos (clock-work), tipos genéricos de sistemas de acordo com Katz & Kahn (produção, apoio, manutenção, adaptativos e gerenciais), sistemas de conhecimento, sistemas de informação
8. Cibernética
Insurgência das causas sobre seus efeitos, o pensamento artificial, retroinformação negativa, revitalização da teleologia, tectologia.
9. Desdobramentos atuais sobre TGS
Raciocínio sistêmico de Peter Senge (natureza cíclica dos sistemas, leis, arquétipos, feedback de reforço e de balanceamento, fontes de estabilidade e resistência ao

Plano de ensino

crescimento), nova teoria dos sistemas sociais de Niklas Luhmann, teoria dos sistemas psico-sociais de Maturana & Varela (tautologia cognoscitiva, sistemas operacionalmente fechados e auto-referenciados, autopoiesis), teoria da complexidade de Morin (sinergia, totalidade, organização), teoria do Caos, teoria dos jogos.
10. Debates em sala de aula Discussão e compartilhamento em sala de aula, das pesquisas feitas pelos alunos.
11. Avaliação da disciplina Avaliação final da disciplina, pelos alunos (conteúdo passado, forma adotada etc.)

Metodologia

1. A disciplina será ministrada através da exposição de parte (introdutória) dos conteúdos pelo professor e da promoção de seminários ("mesa redonda") com os alunos, a partir do aprofundamento da matéria através dos trabalhos que forem passados para serem feitos extra-classe. Caso se perceba que algum aluno não está conseguindo absorver os conteúdos, serão feitos acompanhamentos individuais, tanto em sala quanto via Internet. Além disso, poderão ser apresentados até 4 (quatro) filmes documentários, conforme haja tempo hábil para tal ("O ponto de Mutação", "Galáxia", "BBC - Brain Story" e "Waking Life")

Sistema de avaliação

1. Da absorção dos conteúdos pelos alunos: A absorção dos conteúdos pelos alunos será testada principalmente de duas formas: (i) das participações em sala de aula (compreendendo o número de presenças nas aulas e a participação com questionamentos e apresentação das suas percepções sobre a matéria durante as exposições do professor e sobretudo durante os seminários) e (ii) pelos trabalhos escritos a serem entregues. Adicionalmente, caso se perceba que os alunos não estão respondendo de forma adequada ao modelo (se não for possível caracterizar a absorção dos conteúdos ou francamente os alunos não estiverem interessados em aprender), serão aplicadas provas ad hoc dos conteúdos ministrados. Cada forma representa 50% da composição da nota final do aluno. Das aulas: Além da avaliação promovida pela própria instituição, os alunos terão a oportunidade de se expressar acerca dos conteúdos, da forma de ministrar as aulas e do comportamento do professor através de uma avaliação no último encontro em sala de aula.

Bibliografia básica

1. BERTALLANFY, L. Teoria geral dos sistemas. Petrópolis: Vozes, 2008. 360 p. KATZ & KAHN, D., R. Psicologia Social das Organizações. São Paulo: Atlas, 1974, 551p. SENGE, P. A quinta disciplina: teoria e prática da organização de aprendizagem. São Paulo: Nova Cultural, 1990. 352 p.
--

Bibliografia complementar

1. CHIAVENATO, Idalberto. Introdução a teoria geral da administração. 6. ed. rev. atual. Rio de Janeiro: Campus, c2000. 700 p. : ISBN 8535205578 (broch.) LAUDON, Kenneth C; LAUDON, Jane Price. Gerenciamento de sistemas de informação. 3.ed. Rio de Janeiro: Livros Técnicos e Científicos, 2001. xxi, 433 p. : ISBN 8521612672 (broch.)
--

Plano de ensino

Curso: CCI-BAC - Bacharelado em Ciência da Computação

Turma: CCI122-02A - CIENCIA DA COMPUTAÇÃO FASE 02A

Disciplina: LPG0001 - LINGUAGEM DE PROGRAMAÇÃO

Período letivo: 2017/1

Carga horária: 72

Professor: 6675298 - GUILHERME PIEGAS KOSLOVSKI

Ementa

1. Algoritmos em linguagem de alto nível. Sintaxe de operadores, expressões e instruções de controle. Tipos simples e estruturas compostas. Manipulação de dados em memória. Arquivos. Funções. Teste e documentação de programas.

Objetivo geral

1. Capacitar o aluno a programar computadores usando uma linguagem de programação.

Objetivo específico

1. - Conceituar princípios básicos e fundamentais de programação. - Proporcionar práticas de programação.

Conteúdo programático

1. Introdução a Linguagem de programação C - Características
2. Introdução a Linguagem de programação C - Tipos, Constantes e Variáveis
3. Introdução a Linguagem de programação C - Operadores, Expressões e Funções
4. Introdução a Linguagem de programação C - Funções de Entrada e Saída
5. Introdução a Linguagem de programação C - Teste e documentação de programas
6. Introdução a Linguagem de programação C - Estruturas de controle de fluxo
7. Introdução a Linguagem de programação C - Estruturas de controle de repetição
8. Introdução a Linguagem de programação C - Introdução a ponteiros
9. Introdução a Linguagem de programação C - Funções - valor
10. Introdução a Linguagem de programação C - Funções - referências
11. Introdução a Linguagem de programação C - Vetores
12. Introdução a Linguagem de programação C - Conversão de Tipos
13. Introdução a Linguagem de programação C - Arquivos de cabeçalho
14. Ponteiros - Aritmética de ponteiros
15. Ponteiros - Inicialização
16. Ponteiros - Endereços de elementos de vetores
17. Ponteiros - Ponteiros e strings
18. Ponteiros - Ponteiros para funções
19. Ponteiros - Ponteiros para ponteiros
20. Ponteiros - Alocação dinâmica de memória - conceito, funções, realocação, liberação
21. Tipos de dados definidos pelo usuário - Estruturas
22. Tipos de dados definidos pelo usuário - Union
23. Tipos de dados definidos pelo usuário - Enumerações
24. Tipos de dados definidos pelo usuário - typedef
25. Tipos de dados definidos pelo usuário - Diretivas de compilação
26. Noções de arquivos - File
27. Noções de arquivos- Trabalhando com arquivos - abrindo e fechando, modo texto e binário, entrada e saída formatada

Plano de ensino

28. Noções de arquivos - Leitura e gravação

29. Noções de arquivos - Acesso aleatório

Metodologia

1. A disciplina será desenvolvida através de aulas expositivas, com exercícios práticos em sala de aula, exercícios práticos em laboratórios e trabalhos práticos de laboratórios. Ocasionalmente ocorrerá o desenvolvimento de atividades através de um ambiente de auxílio para aprendizado a distância.

Sistema de avaliação

1. - Provas escritas (P1 e P2);
- Avaliações complementares (AC): trabalhos de implementação, lista de exercícios e relatórios de atividades em laboratório. A nota AC representa a média das notas de todas as avaliações complementares e poderá ser ponderada de acordo com a complexidade da atividade.
$$\text{Nota Final} = P1 * 0,30 + P2 * 0,30 + AC * 0,40$$

Bibliografia básica

1. DEITEL, P. DEITEL, H. C: como programar. 6a edic, a~o. Sa~o Paulo. Pearson Prentice Hall, 2011.
SCHILDT, H. C, completo e total. Sa~o Paulo. Makron Books, McGraw-Hill, 1996.
DE OLIVEIRA, J.F.; MANZANO, J.A.N.G. Algoritmos: Lo'gica para Desenvolvimento de Programac, a~o de Computadores. Editora E'rica, 16a ed., 2004. ISBN 857194718X.

Bibliografia complementar

1. Apostila de Linguagem C da UFMG disponí'vel na Internet em <http://www.ead.cpdee.ufmg.br/cursos/C/> GUIMARA~ES, A.; LAGES, N. Algoritmos e Estruturas de Dados. Editora LTC, 1994. ISBN 8521603789

Plano de ensino

Curso: CCI-BAC - Bacharelado em Ciência da Computação
Turma: CCI122-02B - CIENCIA DA COMPUTAÇÃO FASE 02B
Disciplina: LPG0001 - LINGUAGEM DE PROGRAMAÇÃO
Período letivo: 2017/1
Carga horária: 72
Professor: 3990362 - RUI JORGE TRAMONTIN JUNIOR

Ementa
1. Algoritmos em linguagem de alto nível. Sintaxe de operadores, expressões e instruções de controle. Tipos simples e estruturas compostas. Manipulação de dados em memória. Arquivos. Funções. Teste e documentação de programas.

Objetivo geral
1. Capacitar o aluno a programar computadores usando uma linguagem de programação.

Objetivo específico
1. - Conceituar princípios básicos e fundamentais de programação. - Proporcionar práticas de programação.

Conteúdo programático
1. 1. Introdução a Linguagem de programação C 1.1. Características 1.2. Tipos, Constantes e Variáveis 1.3. Operadores, Expressões e Funções
2. 2. Funções de Entrada e Saída
3. 3. Teste e documentação de programas
4. 4. Estruturas de seleção 4.1. switch ... case 4.2. if 4.3. if ... else
5. 5. Estruturas de iteração 5.1. for 5.2. do ... while 5.3. while
6. 6. Introdução a ponteiros
7. 7. Funções 7.1. Parâmetros passados por valor 7.2. Parâmetros passados por referência 7.3. Recursividade 7.4. Arquivos de cabeçalho
8. 8. Vetores 8.1. Unidimensionais 8.2. Multidimensionais
9. 9. Ponteiros 9.1. Conceito e funcionamento 9.2. Declaração e Utilização 9.3. Aritmética de ponteiros 9.4. Inicialização 9.5. Endereços de elementos de vetores 9.6. Ponteiros e strings 9.7. Ponteiros para funções 9.8. Ponteiros para ponteiros
10. 10. Alocação dinâmica de memória 10.1. Conceito 10.2. Funções de alocação 10.3. Realocação 10.4. Liberação 10.5. Alocação dinâmica de vetores
11. 11. Tipos de dados definidos pelo usuário 11.1. Estruturas

Plano de ensino

11.1.1. Criando e usando uma estrutura 11.1.2. Atribuições entre estruturas 11.1.3. Estruturas aninhadas 11.1.4. Passagem para funções 11.1.5. Ponteiros para estruturas 11.1.6. Vetor de estruturas 11.1.7. Alocação dinâmica de estruturas 11.2. Union 11.3. Enumerações 11.4. typedef 11.5. Diretivas de compilação
12. 12. Noções de arquivos 12.1. Introdução 12.2. Abrindo e fechando 12.3. Modo texto e binário 12.4. Entrada e saída formatada 12.5. Leitura e gravação 12.6. Lendo e gravando registros 12.7. Acesso aleatório

Metodologia

1. - Aulas expositivas e dialogadas em sala; - Aulas práticas em laboratório de informática; - Uso de software para escrever, compilar e testar os algoritmos criados nas aulas.
--

Sistema de avaliação

1. A qualidade do desempenho do estudante será avaliada com base no desenvolvimento das seguintes atividades e com os seguintes critérios: Prova 1 (escrita) = 20% Prova 2 (escrita) = 30% Prova 3 (escrita) = 30% Trabalho final (prático) = 20%

Bibliografia básica

1. DEITEL, P. DEITEL, H. C.: como programar. 6a edição. São Paulo. Pearson Prentice Hall, 2011. SCHILDT, H. C completo e total. São Paulo. Makron Books, McGraw-Hill, 1996. DE OLIVEIRA, J.F.; MANZANO, J.A.N.G. Algoritmos: Lógica para Desenvolvimento de Programação de Computadores. Editora Érica, 16a ed., 2004. ISBN 857194718X.

Bibliografia complementar

1. Apostila de Linguagem C da UFMG disponível na Internet em http://www.inf.ufsc.br/~fernando/ine5412/C_UFMG.pdf (acesso em fevereiro de 2014) GUIMARÃES, A.; LAGES, N. Algoritmos e Estruturas de Dados. Editora LTC, 1994. ISBN 8521603789

Plano de ensino

Curso: CCI-BAC - Bacharelado em Ciência da Computação

Turma: CCI122-02U - CIENCIA DA COMPUTAÇÃO FASE 02U

Disciplina: ALG2002 - ÁLGEBRA LINEAR E GEOMETRIA ANALÍTICA II

Período letivo: 2017/1

Carga horária: 72

Professor: 9549803 - Maisa Damazio Franco

Ementa

1. Matrizes, Sistemas de equações lineares, Espaço vetorial, Transformações lineares, Operadores Lineares, Autovalores e autovetores, Produto interno.

Objetivo geral

1. Reconhecer a álgebra linear como uma ferramenta que pode ser utilizada em diversas áreas da ciência e da tecnologia, proporcionando ao aluno a compreensão dos conteúdos da disciplina necessários a sua formação profissional.

Objetivo específico

1. - Identificar os vários tipos de matrizes, calcular determinantes, classificar sistemas lineares e resolver problemas diversos utilizando sistemas de equações lineares.
- Definir espaço vetorial com suas operações, propriedades e teoremas e resolver problemas envolvendo esses conceitos.
- Compreender o conceito de transformação linear, suas propriedades, operações, sua representação matricial.
- Reconhecer transformações lineares como uma importante ferramenta na resolução de problemas em diversas áreas do conhecimento.
- Possibilitar ao acadêmico, no estudo de produto interno, compreender a generalização do produto escalar, ampliando o conceito de distâncias, comprimentos, medida de ângulos, ortogonalidade, projeções ortogonais e bases ortonormais.
- Identificar o produto interno como uma ferramenta potencial na resolução de problemas.

Conteúdo programático

1. Apresentação do Plano de Ensino
Matrizes:
Introdução
Definição
Operações
2. Matrizes: Tipos especiais de matrizes e propriedades
3. Determinantes
4. Matriz linha reduzida
Matriz escalonada
5. Sistemas de Equações lineares
6. Matriz ampliada de um sistema
7. Sistemas lineares: Classificação de um sistema e resolução
8. Sistemas lineares: Método de escalonamento de Gauss
9. Sistemas lineares: método da inversa
10. Espaço vetorial: definição
11. Subespaço vetorial: definição
12. Espaço vetorial: dependência e independência linear
13. Espaço vetorial: Subespaço gerado por um conjunto de vetores
14. Espaço vetorial: Base e dimensão
15. Espaço vetorial: Matriz mudança de base
16. Espaço vetorial: Matriz mudança de base inversa
17. Transformação linear: definição e propriedades
18. Transformação linear: núcleo e imagem
19. Transformação linear: injetoras e sobrejetoras

Plano de ensino

20. Transformação linear: transformações induzidas por uma matriz
21. Transformação linear: composição de uma transformação
22. Matriz de uma transformação linear
23. Isomorfismo e inversa de uma transformação linear
24. Operadores lineares: transformações especiais no plano
25. Operadores lineares: Transformações especiais no espaço
26. Operadores lineares: Operadores auto-adjuntos e ortogonais
27. Operadores lineares: Operadores inversíveis
28. Autovalores e autovetores: definição
29. Autovalores e autovetores: Autovalores e autovetores de uma matriz
30. Autovalores e autovetores: Polinômio característico
31. Autovalores e autovetores: Cálculo de autovalores e autovetores
32. Autovalores e autovetores: Matrizes semelhantes
33. Autovalores e autovetores; diagonalização
34. Produto interno: Definição de produto interno
35. Produto interno: Ortogonalidade em espaços com produto interno
36. Produto interno: Complementos e projeções ortogonais
37. Produto interno; Bases ortonormais; processo de Gram-schmidt
38. Produto interno: Fatoração QR.

Metodologia

1. Aulas expositivas e dialogadas. Utilização de ferramentas tecnológicas para o desenvolvimento de atividades presenciais e à distância. Aulas experimentais com o uso de recursos tecnológicos. Aulas em ambiente virtual de aprendizagem- Moodle
--

Sistema de avaliação

1. Quatro avaliações escritas individuais durante o semestre letivo, com pesos iguais (P1, P2, P3, P4); - Trabalho em equipe -T1 e T2 (com diretrizes específicas)- Peso Provas: 8 Peso trabalhos: 2 Média=(8P+2T)/10

Bibliografia básica

1. ANTON, H. e RORRES, C. Álgebra linear com aplicações. São Paulo: Ed. Bookman, 2001. BOLDRINI, J. L. Álgebra Linear. São Paulo: Harbra, 2000. STEINBRUCH, A. e WINTERLE, P., Álgebra Linear. São Paulo: Ed. Makron Books. 1987.

Bibliografia complementar

1. LIMA, Elon L.: Álgebra Linear, Coleção Matemática Universitária, IMPA, Rio de Janeiro, RJ, 1996 LEON, Steven. Álgebra linear com aplicações. Rio de Janeiro: LTC, 1999. POOLE, David. Álgebra linear. Rio de Janeiro: Pioneira Thomson Learning, 2004. LIPSCHUTZ, Seymour. Álgebra linear: resumo da teoria, 600 problemas resolvidos, 524 problemas propostos . 2 ed. rev. São Paulo: Makron Books, 1972. LAY, David C; CAMELIER, Ricardo; IORIO, Valeria de Magalhães. Algebra linear e suas aplicações. 2. ed. Rio de Janeiro: Livros Técnicos e Científicos, 1999.



REPÚBLICA FEDERATIVA DO BRASIL
ESTADO DE SANTA CATARINA
Universidade do Estado de Santa Catarina - UDESC
CENTRO DE CIÊNCIAS TECNOLÓGICAS - UDESC/CCT



Plano de ensino

--

Plano de ensino

Curso: CCI-BAC - Bacharelado em Ciência da Computação

Turma: CCI122-02U - CIENCIA DA COMPUTAÇÃO FASE 02U

Disciplina: CDI2001 - CÁLCULO DIFERENCIAL E INTEGRAL II

Período letivo: 2017/1

Carga horária: 72

Professor: 2490099 - JONES CORSO

Ementa

1. Integral definida. Funções de várias variáveis. Integrais múltiplas. Séries numéricas. Séries de funções

Objetivo geral

1. Proporcionar ao estudante a oportunidade de adquirir conhecimentos de Cálculo Diferencial e Integral, bem como aplicar estes conceitos em sua área de atuação.

Objetivo específico

1. Aplicar conceitos e resolver problemas que envolvam integral definida.
Reconhecer e resolver problemas que envolvam funções de várias variáveis.
Reconhecer e resolver problemas que envolvam integrais múltiplas.
Reconhecer e resolver problemas que envolvam sequências e séries.

Conteúdo programático

1. Apresentação da Disciplina.
Plano de Ensino.
2. Somas Superiores e Inferiores.
Integral Definida.
3. Interpretação Geométrica de uma Integral Definida.
Propriedades da Integral Definida.
Teorema do Valor Médio para Integrais.
Teorema Fundamental do Cálculo.
4. Aplicações da Integral Definida: área.
Método da substituição na integral definida.
Método da integração por partes na integral definida.
5. Área em coordenadas cartesianas.
Integração em relação a x
Integração em relação a y
6. Área em coordenadas polares
7. Integrais Impróprias.
Integrais de funções descontínuas.
8. Comprimento de Arco:
Comprimento de Arco em coordenadas cartesianas.
Comprimento de Arco em coordenadas paramétricas.
Comprimento de Arco em coordenadas polares.
9. Volume de Revolução:
Revolução em torno de eixo coordenado
Revolução em torno de eixo paralelo.
10. Funções de Várias Variáveis:
Definição de Funções de várias variáveis.
Gráficos de Funções de duas variáveis.
Curvas e Superfícies de Nível.
11. Limite de funções:
Definição de Limite de Funções de duas variáveis.
Cálculo de limites pela definição.
Limites por Caminhos.
12. Continuidade de Funções de Várias Variáveis.
Derivadas Parciais.
Derivadas parciais de segunda ordem.
13. Interpretação geométrica de derivadas Parciais.
Derivação Implícita.
14. Regra da Cadeia para Derivada de Funções de Várias Variáveis.

Plano de ensino

15. Taxas Relacionadas e derivadas de funções compostas. Aplicações.
16. Diferencial Parcial de Funções de várias Variáveis. Diferencial Total de funções de várias variáveis. Aplicações.
17. Máximos e Mínimos: Pontos Críticos. Maximização e Minimização de Funções de duas variáveis.
18. Integral Dupla: Cálculo da Integral Dupla. Mudança da ordem de Integração.
19. Interpretação Geométrica da Integral Dupla. Cálculo de Volume de Sólidos. Cálculo de Área de Regiões Planas.
20. Diferencial de área em coordenadas polares. Determinante Jacobiano em coordenadas polares. Integral Dupla em Coordenadas Polares.
21. Integral Tripla Integral Tripla em Coordenadas Cartesianas.
22. Integral Tripla em Coordenadas Cilíndrica: Diferencial de área em coordenadas cilíndricas. Determinante Jacobiano em coordenadas cilíndricas. Cálculo de Integral Tripla em Coordenadas Cilíndrica
23. Integral Tripla em Coordenadas Esféricas: Diferencial de área em coordenadas esféricas. Determinante Jacobiano em coordenadas esféricas. Cálculo da Integral Tripla em Coordenadas Esféricas
24. Sequências Numéricas Gráfico e Limite de uma sequência Sequências Convergentes e Divergentes
25. Subsequências. Sequências Limitadas. Sequências Monótonas. Teoremas de Convergência de Sequências.
26. Séries Numéricas. Definição de Séries Numéricas. Séries Convergentes e Divergentes. Propriedades.
27. Condição Necessária para Convergência. Critério do Termo Geral. Série Harmônica. Séries Geométricas.
28. Critérios de Convergência: Critério de Comparação. Critério da Razão. Critério da Raiz.
29. Critério da Integral. Séries Hiper-Harmônicas (Séries-p). Séries Alternadas.
30. Séries Alternadas e Critério de Leibnitz. Séries de Termos de Sinais Quaisquer. Convergência Absoluta e Convergência Condicional.
31. Séries de funções. Séries de Potências. Intervalo e Raio de Convergência.
32. Séries de Taylor. Séries de MacLaurin.
33. Derivação de Séries de Potências. Integração de Séries de Potências.
34. Erro de Aproximação para Soma de Séries Alternadas. Série Binomial.

Plano de ensino

Metodologia

1. Aulas presenciais expositivas e dialogadas com resolução de exercícios orientados.
Utilização de ferramentas tecnológicas relacionados com os tópicos da disciplina.
Atendimento individual ao aluno.
Atividades à distância pela plataforma Moodle, no limite de 20% da carga horária da disciplina

Sistema de avaliação

1. A média semestral será calculada pela média aritmética das notas das quatro avaliações.

Bibliografia básica

1. FLEMMING, Diva Marília; GONÇALVES, Mirian Buss. Cálculo A: funções, limite, derivação, noções de integração. 5. ed. rev. ampl. São Paulo: Makron, 2004.
GONÇALVES, Mirian Buss; FLEMMING, Diva Marília. Cálculo B: funções de várias variáveis integrais duplas e triplas. São Paulo: Makron Books, 1999.
STEWART, James. Cálculo. São Paulo: Cengage Learning, 2009. volume 2.

Bibliografia complementar

1. ANTON, Howard. Cálculo: um novo horizonte. 6. ed. Porto Alegre: Bookman, 2000.
CARELLI, Enori; ZUCHI, Ivanete; MANDLER, Marnei; FIGUEIREDO, Elisandra, AGUIAR, Rogério. Apostila de Cálculo Diferencial e Integral II. Joinville: UDESC, 2015.
GUIDORIZZI, Hamilton Luiz. Um curso de cálculo. Volume 1, 2, 3 e 4. 5. ed. Rio de Janeiro: Livros Técnicos e Científicos, 2002.
THOMAS, George Brinton; WEIR, Maurice D; HASS, Joel; GIORDANO, Frank R. Cálculo, volume 2: George B. Thomas. 11. ed. São Paulo: Pearson/Addison Wesley, 2009.
SWOKOWSKI, Earl William; FARIAS, Alfredo Alves de. Cálculo com geometria analítica. Volumes 1 e 2. 2ª edição. São Paulo: Makron Books, 1995.

Plano de ensino

Curso: CCI-BAC - Bacharelado em Ciência da Computação

Turma: CCI122-02U - CIENCIA DA COMPUTAÇÃO FASE 02U

Disciplina: EST0007 - PROBABILIDADE E ESTATÍSTICA

Período letivo: 2017/1

Carga horária: 72

Professor: 3149242 - VOLNEI AVILSON SOETHE

Ementa

1. Probabilidade e Estatística passa a ter a seguinte ementa: "Análise Exploratória de Dados. Probabilidades. Variáveis aleatórias discretas e contínuas. Distribuições de Probabilidade Discretas e Contínuas. Distribuições de probabilidade conjuntas. Estimação de Parâmetros. Testes de hipóteses. Regressão e Correlação. Noções de amostragem

Objetivo geral

1. proporcionar ao aluno os conhecimentos necessários para avaliação descritiva de dados, realizando teste de hipóteses em análises de inferência, permitindo a identificação da correlação entre variáveis com realizando de estimativas

Objetivo específico

1. O aluno deverá ao final do semestre letivo ser capaz de utilizar os conceitos: a) para a avaliação descritiva de dados; b) para caracterizar o nível de confiança de parâmetros por meio de estimativas; c) validar hipóteses estatísticas de conjunto de dados; d) reconhecer o nível de associação entre variáveis e realizar projeções de cenários com base em metodologias de regressão.

Conteúdo programático

1. Apresentação do programa da disciplina
2. Estatística: origem, classificação, técnicas de amostragem, aplicações
3. Análise exploratória de dados.
Dados absolutos e dados relativos
Gráficos estatísticos
4. Elaboração de distribuição de frequência
Gráficos de frequência
5. Medidas de posição: média aritmética, geométrica e harmônica
Exercícios
6. Separatrizes: mediana, quartil e percentil
Moda
7. Medidas de dispersão: amplitude total, desvio médio absoluto, variância amostral e populacional
8. Desvio padrão amostral e populacional. coeficiente de variação. Erro padrão para a média
9. Medidas de assimetria. Coeficiente de assimetria.
10. Medidas de curtose. Coeficiente de curtose
11. Exercícios envolvendo análise exploratória de dados
12. Avaliação
13. Probabilidade: definições, chance de um evento, cálculo de probabilidade. Probabilidade de ocorrência de dois ou mais eventos. Probabilidade condicional.
14. Regra de Bayes e técnicas de contagem
15. Variáveis aleatórias: definição, classificação, esperança matemática, variância esperada
16. Distribuição binomial. Distribuição multinomial. Distribuição geométrica, hipergeométrica e Poisson
17. Exercícios envolvendo distribuições discretas de probabilidade
18. Distribuição normal de probabilidade. Aplicações
19. Distribuição normal para variáveis conjuntas. Aplicações
20. Distribuição inversa normal.
21. Exercícios de aplicação envolvendo distribuição normal
22. Aproximação teórica de distribuição binomial e Poisson pela Normal

Plano de ensino

23. Exercícios envolvendo distribuições discretas e contínuas
24. Avaliação
25. Estimativas para média com variâncias conhecidas e desconhecidas. Aproximação por t de Student.
26. Estimativas para proporção. Estimativa para diferença de médias e diferença de proporções.
27. Teste de hipóteses: definição, tipos de hipóteses, tipos de erros, teste para a média com variâncias conhecidas e desconhecidas
28. Teste de hipóteses para diferença de médias e diferença de proporções
29. Exercícios envolvendo estimativas e teste de hipóteses
30. Teste de aderência e independência pelo qui-quadrado.
31. Análise de correlação e regressão. Coeficiente de correlação e diagrama de dispersão
32. Regressão linear e quadrática. Exercícios.
33. Regressão exponencial. Exercícios.
34. Regressão múltipla e análise da variância residual
35. Exercícios envolvendo análise de regressão
36. Avaliação

Metodologia

1. exposição da matéria em quadro negro, incentivando a participação do aluno com a apresentação de diversos exemplos e exercícios. Apresentação de listas de exercícios contendo exemplos de exercícios resolvidos. Acompanhamento de dúvidas e questões fora da sala de aula. Incentivo ao uso de computadores e calculadoras científicas na determinação de estatísticas

Sistema de avaliação

1. Provas escritas, com média aritmética simples e listas de exercícios.
--

Bibliografia básica

1. ANGELINI, Flávio. MILONE, Giuseppe. Estatística Geral. São Paulo: Atlas, 1993. DANAIRE, D. MARTINS, G. A. Princípios da Estatística. 4ª Edição. São Paulo: Atlas, 1991. FONSECA, Jairo Simon. MARTINS, G. A. Curso de Estatística. 4ª Edição. São Paulo: Atlas, 1994.
--

Bibliografia complementar

1. MOORE, David S. A estatística básica e sua prática. 5. ed. Rio de Janeiro: LTC, 2011 MORETTIN, Luiz Gonzaga. Estatística básica: probabilidade e inferência, volume único. São Paulo: Pearson: Makron Books, c2010 KASMIER, Leonard J. Estatística aplicada à Economia e Administração. MacGraw-Hill: São Paulo, 1982. LIPSCHUTZ, Seymour. Probabilidade. 4ª Edição. São Paulo: Makron Books, 1993. SPRIEGEL, M. R. Estatística. Coleção Schaum. São Paulo: Mc Graw - Hill, 1993. BUSSAB, Wilton de Oliveira; MORETTIN, Pedro A. Estatística básica. 4.ed. São Paulo: Atual, 1987 TOLEDO, Geraldo Luciano. OVALLE, Ivo Izidoro. Estatística básica. São Paulo. Atlas. 1991.
--

Plano de ensino

Curso: CCI-BAC - Bacharelado em Ciência da Computação

Turma: CCI122-02U - CIENCIA DA COMPUTAÇÃO FASE 02U

Disciplina: FCC0002 - FÍSICA PARA CIÊNCIA DA COMPUTAÇÃO

Período letivo: 2017/1

Carga horária: 72

Professor: 3128571 - JACIMAR NAHORNY

Ementa

1. Eletrostática básica e circuitos resistivos. Capacitores e dielétricos. Indutores. Semicondutores. Análise de circuitos elétricos no domínio do tempo. Circuitos eletrônicos básicos.

Objetivo geral

1. Desenvolver no aluno a capacidade de análise crítica, interpretação física bem como resolução de problemas diversos.

Objetivo específico

1. O aluno deverá ser capaz de:
 - 1-Compreender os fundamentos da Física Geral.
 - 2-Descrever os fenômenos ligados a eletricidade, ao magnetismo e aos semicondutores.
 - 3-Conhecer as principais aplicabilidades em processos físicos e a Engenharia Industrial.

Conteúdo programático

1. Apresentação
Apresentação do plano de aulas
2. -Carga Elétrica e Campo Elétrico
Carga elétrica.
Quantização e conservação da carga.
Condutores isolantes e lei de Coulomb.
3. -Carga Elétrica e Campo Elétrico
Vetores.
4. -Carga Elétrica e Campo Elétrico
Campo Elétrico
Linhas de Campo.
5. Carga Elétrica e Campo Elétrico
Determinação do campo elétrico
6. Carga Elétrica e Campo Elétrico
Dipolos elétricos.
Campo de um dipolo.
7. Potencial Elétrico
Potencial elétrico.
Superfícies equipotenciais.
8. Revisão
9. Prova
1º Prova
10. Capacitância e Dielétricos
Capacitores Capacitância
11. Capacitância e Dielétricos
Capacitores
Cálculo da capacitância
12. Capacitância e Dielétricos
Associação de capacitores.
Armazenamento de energia em capacitores
13. Corrente Resistência e Força Eletromotriz
Corrente elétrica.
Resistividade e resistência elétrica.
Lei de Ohm
14. Corrente Resistência e Força Eletromotriz
Força Eletromotriz. Resistência interna
15. Energia e Potência em Circuitos

Plano de ensino

16. Revisão
17. Prova 2ª Prova semestral.
18. Circuitos de Corrente Contínua Associação de resistores.
19. Circuitos de Corrente Contínua Leis de Kirchhoff
20. Circuitos de Corrente Contínua Instrumentos de medidas elétricas
21. Circuitos de Corrente Contínua Circuitos RC.
22. Campo Magnético e Força Magnética Magnetismo. Campo magnético e linhas de campo.
23. Campo Magnético e Força Magnética Movimento de partículas em campos magnéticos.
24. Campo Magnético e Força Magnética Força magnética sobre condutores transportando corrente.
25. Campo Magnético e Força Magnética Fontes de campos magnéticos
26. Revisão
27. 3ª Prova semestral
28. Indução Magnética Indução magnética, Lei de Faraday, Lei de Lenz
29. Indutância Indutores, indutância, circuitos RL
30. Indutância Circuito LC, circuito RLC
31. Condução de Eletricidade nos Sólidos Condução de Eletricidade nos Sólidos
32. Condução de Eletricidade nos Sólidos Condutividade elétrica nos sólidos, níveis de energia dos sólidos
33. Condução de Eletricidade nos Sólidos Isolantes, metais e semicondutores. Dopagem. Junção p-n e diodos. Transistores.
34. Prova 4ª Prova semestral.
35. Feriado
36. Semana da Computação

Metodologia

1. A disciplina será trabalhada através de aulas expositivas e leituras adicionais. Cada item do plano de ensino será trabalhado de forma expositiva, procurando dar aos alunos exemplos de aplicação prática. Como parte de seu desenvolvimento os alunos, ao fim de cada tópico do conteúdo descrito acima, deverão solucionar listas de exercícios que constituirão uma base para as avaliações.

Sistema de avaliação

1. - Cinco prova de mesmo peso.
- A média semestral será calcula da seguinte forma:
Para alunos que não pediram segunda chamada durante o semestre a média será calculada com as quatro maiores notas das cinco provas.
Para alunos que pediram segunda chamada durante o semestre a média será calculada com as cinco notas das cinco provas.

Bibliografia básica

Plano de ensino

1. SEARS, F.; ZEMANSKI, M.W.; YOUNG, & FREEDMAN. Física III; Livros Técnicos e Científicos Editora S.A., RJ, 2003, Vol. 03. - 10a edição.
2. HALLIDAY D. e RESNICK, R. Fundamentos de Física; Livros Técnicos e Científicos Editora S.A., RJ, 2002, Vol. 03; 4a Edição (ou 6a Edição).
3. SILVA, R. P. Eletrônica Básica: um enfoque voltado à informática. Florianópolis: Editora da UFSC, 1995.
4. MALVINO, A.P. Eletrônica - Volume 1, McGraw-Hill, 1987.

Bibliografia complementar

1. Sears e Zemansky - Young e Freedman. Física III, volume 3, Eletromagnetismo, Editora Pearson Education do Brasil. 12a edição (2009).

Halliday, D. , Resnick, R. e Walker, J. Fundamentos de Física, Volume 3, Eletromagnetismo, Editora Livros Técnicos e Científicos. 8a edição (2009)

Nussenzveig, H. M. Curso de Física Básica, volume 3, Eletricidade e Magnetismo, Editora Edgard Blücher Ltda. 1a edição (1997)

Plano de ensino

Curso: CCI-BAC - Bacharelado em Ciência da Computação

Turma: CCI122-02U - CIENCIA DA COMPUTAÇÃO FASE 02U

Disciplina: MDI0001 - MATEMÁTICA DISCRETA

Período letivo: 2017/1

Carga horária: 72

Professor: 1033139444 - KARINA GIRARDI ROGGIA

Ementa

1. Teoria dos conjuntos, relações e funções, reticulados, contagem, princípio da multiplicação e adição, princípio de inclusão e exclusão; princípio das casas de pombo; primeiro e segundo princípios de indução matemática; Leis de composição interna; estruturas algébricas; grupos e subgrupos; homomorfismos e isomorfismos; anéis e subaneis; corpos.

Objetivo geral

1. Proporcionar ao aluno a oportunidade de apropriar-se de conceitos algébricos e da linguagem formal da matemática, e relacioná-los com temas e aplicações da Ciência da Computação.

Objetivo específico

1. Capacitar o aluno para expressar problemas e soluções em linguagem matemática, utilizando provas formais bem estruturadas. Capacitar o aluno no uso e reconhecimento de provas e estruturas definidas por indução matemática. Capacitar o aluno na identificação e resolução de problemas que envolvam:
 - conceitos da teoria dos conjuntos;
 - relações e funções;
 - processos de contagem e análise combinatória simples;
 - estruturas algébricas.

Conteúdo programático

1. Técnicas de Demonstração: Proposições, Conectivos e Tabelas-Verdade
2. Técnicas de Demonstração: Prova Direta
3. Técnicas de Demonstração: Prova por Contraposição
4. Técnicas de Demonstração: Prova por Redução ao Absurdo
5. Primeiro Princípio da Indução Matemática
6. Segundo Princípio da Indução Matemática
7. Teoria dos Conjuntos: Conceitos Básicos
8. Teoria dos Conjuntos: Diagramas de Venn
9. Teoria dos Conjuntos: Operações sobre Conjuntos
10. Relações: Relações Binárias
11. Relações: Propriedades de Relações
12. Relações: Relações de Equivalência
13. Relações: Relações de Ordem
14. Relações: Fechos
15. Funções: Propriedades de Funções (injeção, sobrejeção e bijeção)
16. Funções: Função Parcial
17. Funções: Composição de Funções
18. Contagem: Princípios da Multiplicação e da Adição
19. Contagem: Princípios de Inclusão e Exclusão
20. Contagem: Princípio das Casas de Pombo
21. Estruturas Algébricas: Operações Binárias
22. Estruturas Algébricas: Propriedades das Operações Binárias

Plano de ensino

23. Estruturas Algébricas: Semigrupos, Monóides e Grupos
24. Homomorfismos
25. Reticulados: Limitantes de Conjuntos Parcialmente Ordenados
26. Reticulados: Reticulados como Relação de Ordem e como Álgebra
27. Reticulados: Tipos de Reticulados
28. Álgebra de Boole

Metodologia

1. A disciplina será desenvolvida através de aulas expositivo-dialogadas, com exercícios práticos orientados em sala de aula e exercícios a serem passados para resolução extra-classe. O conteúdo da disciplina poderá ser ministrado na modalidade de ensino a distância em até 20% do total de sua Carga Horária (MEC PORTARIA Nº 4.059, DE 10 DE DEZEMBRO DE 2004 publicado no DOU de 13/12/2004, Seção 1, p. 34).

Sistema de avaliação

1. Do desempenho do aluno:
A qualidade do desempenho do aluno será avaliada com base no desenvolvimento das seguintes atividades e com os seguintes critérios:
a) Avaliação contínua englobando assiduidade, participação em aula e resolução de exercícios (AC);
b) Provas individuais (P1, P2 e P3).
A Média Semestral (MS) será calculada pela fórmula
$$MS = 0,3 * (P1 + P2 + P3) + 0,1*AC$$

Do desempenho da disciplina e do professor:
Os estudantes terão, igualmente, a oportunidade de fazer uma avaliação do desempenho do professor e da disciplina. As informações sobre esta atividade serão fornecidas pelo coordenador do curso.

Das regras para revisão das avaliações:
Depois da publicação das notas pelo professor, os alunos têm 07 dias corridos para solicitar a revisão com o professor. Esta revisão será feita na sala do professor, preferencialmente em horário de atendimento aos alunos, ou em um horário do qual o professor possa atendê-los.

Bibliografia básica

1. DOMINGUES, Hygino H.; IEZZI, Gelson. Álgebra moderna. 4 ed. São Paulo: Atual, 2003.

GERSTING, Judith L. Fundamentos matemáticos para a ciência da computação. 5 ed. Rio de Janeiro: LTC, 2008.

MENEZES, Paulo. B. Matemática discreta para computação e informática. 2 ed. Porto Alegre: Bookman, 2005.

Bibliografia complementar

1. LOVÁSZ L.; PELIKÁN J.; VESZTERGOMI K. Matemática discreta. Textos Universitários. 2 ed. Rio de Janeiro: SBM, 2010.

MENEZES, Paulo B.; TOSCANI, Laura V.; LÓPEZ, Javier G. Aprendendo Matemática Discreta com Exercícios. Porto Alegre: Bookman, 2009.

Plano de ensino

Curso: CCI-BAC - Bacharelado em Ciência da Computação

Turma: CCI122-02U - CIENCIA DA COMPUTAÇÃO FASE 02U

Disciplina: SNA0001 - SISTEMAS DE NUMERAÇÃO E ÁLGEBRA DE BOOLE

Período letivo: 2017/1

Carga horária: 36

Professor: 3398030 - EDINO MARIANO LOPES FERNANDES

Ementa

1. Sistemas de numeração. Conversão de bases. Aritmética binária. Álgebra de Boole. Teoremas e postulados de Boole. Funções booleanas. Formas canônicas. Mapas de Karnaugh.

Objetivo geral

1. Capacitar o aluno a dominar os conceitos básicos de sistemas de numeração e álgebra de Boole.

Objetivo específico

1. - Conceituar sistemas de numeração e seu histórico;
- Conceituar conversões entre diferentes sistemas;
- Conceituar operações aritméticas em diferentes sistemas;
- Introduzir conceitos básicos da representação binária da informação numérica e não numérica (textual);
- Conceituar Álgebra de Boole, Teoremas e postulados de Boole;
- Conhecer Funções Booleanas e formas canônicas;
- Conceituar e utilizar Mapas de Karnaugh.

Conteúdo programático

1. Sistemas Numéricos
1.1 Sistemas de Numeração:
1.1.1 Objetivo;
1.1.2 Tipos;
1.1.3 Base;
1.1.4 Potência.
2. 1.5 Principais sistemas;
1.6 Conversão entre sistemas.
3. 2.3 Representação de Inteiros com e sem sinal;
2.4 Representação de reais - Ponto flutuante.
4. 3.2 Tabela verdade;
3.3 Funções AND, OR e NOT.
5. 4.3 Aplicações da Álgebra Booleana
4.4 Formas Canônicas
6. 5.3 Mapeamento de funções fora da forma padrão e incompletamente especificadas;
5.4 Mapas para 5 e 6 variáveis.
7. Avaliação Teórica.
8. Apresentação Trabalho em Grupos.
9. Realização de Atividades Práticas e Resolução de Exercícios.
10. Semana da Computação.

Metodologia

1. Aulas expositivas e dialogadas com aulas práticas para desenvolvimento de exercícios.

Sistema de avaliação

1. A nota final da disciplina será composta por duas Provas (P1 e P2) e uma nota de Trabalhos (T) - Sendo este composto por exercícios desenvolvidos durante o semestre.
A média final da disciplina é dada pela seguinte fórmula:
 $Média = P1 \cdot 0,4 + P2 \cdot 0,4 + T \cdot 0,2$

Plano de ensino

Bibliografia básica

1. LOURENÇO, A. C. Sistemas Numéricos e Álgebra Booleana. Editora Érica.
MELO, M. O. Eletrônica Digital. Editora da UDESC. Florianópolis, 2002.
BIGNELL, James. DONOVAN, Robert. Eletronica Digital. Cengage Learning, 2009.
LORIN, H. Introdução à Arquitetura e Organização de Computadores. Ed. Campus.

Bibliografia complementar

1. TANENBAUM, Andrew S. Organização Estruturada de Computadores. Ed. Prentice Hall. 5a Edição.

Plano de ensino

Curso: CCI-BAC - Bacharelado em Ciência da Computação

Turma: CCI122-03U - CIENCIA DA COMPUTAÇÃO FASE 03U

Disciplina: ANN0001 - ANÁLISE NUMÉRICA

Período letivo: 2017/1

Carga horária: 72

Professor: 3149269 - FERNANDO DEEKE SASSE

Ementa

1. Análise numérica: características e importância. Máquinas digitais: precisão, exatidão e erros. Aritmética de ponto flutuante. Sistemas de numeração. Resolução computacional de sistemas de equações lineares. Resolução de equações algébricas e transcendentais. Resolução de sistemas de equações não lineares. Aproximação de funções: interpolação polinomial, interpolação spline, ajustamento de curvas, aproximação racional e por polinômios de Chebyshev. Integração numérica: Newton-Cotes e quadratura gaussiana

Objetivo geral

1. Conhecer os métodos numéricos utilizados para resolver problemas de física básicos que não possuem soluções exatas. Ensinar a modelar problemas numericamente e aplicar ou implementar os códigos numéricos necessários.

Objetivo específico

1. Ao ter completado a disciplina com sucesso o estudante deverá dominar os seguintes tópicos:
Programação estruturada.
Entender os conceitos de programação estruturada básicos envolvendo tomadas de decisão, loops, funções e parâmetros implementados em uma linguagem de programação numérica.
Métodos numéricos
. Entender os métodos numéricos mais comuns usados em física, quando usar cada método, e como implementar os métodos básicos de maneira estruturada usando uma linguagem de programação. Conhecer as bibliotecas numéricas disponíveis no sistema numérico utilizado.
Acurácia numérica.
Estimar o erro inerente aos diversos métodos numéricos.
Eficiência numérica.
Estimar a eficiência de um método numérico específico quando mais de uma opção é disponível para resolver uma certa classe de problemas.
Estabilidade numérica
. Entender as propriedades de convergência e limitações dos diversos métodos numéricos.

Conteúdo programático

1. Erros computacionais
Análise de erro em operações aritméticas
2. Fórmula de Taylor
Dedução da fórmula de Taylor e fórmula de Lagrange para o resto.
3. Sistemas de ponto flutuante
Sistema de ponto flutuante na forma geral
4. Padrão IEEE 754
Precisão simples, dupla, simples e estendida
5. Equações algébricas
Determinação gráfica de raízes de equações algébricas
Uso de comandos de Maple para determinação de raízes
6. Convergência de métodos iterativos
Convergência linear e superlinear
7. Método da bissecção
Heurística do método
Algoritmo
8. Métodos de ponto fixo
Teoria e exemplos
Iteração linear
Algoritmo e implementação computacional
Análise de convergência

Plano de ensino

9. Método de Newton-Raphson Método de Newton-Raphson como um caso especial de método de ponto fixo Dedução geométrica Exemplo Análise de convergência
10. Método da secante Introdução geométrica Análise de convergência Exemplos
11. Raízes de polinômios Uso do método de Newton-Raphson para determinação de raízes complexas de polinômios.
12. Raízes complexas de equações não polinomiais.
13. Método de Gauss simples para sistemas lineares Introdução ao método Implementação computacional e exemplos
14. Método de Gauss com pivotação Pivotação parcial de linhas
15. Implementação do método de Gauss com pivotação Implementação do método de Gauss com pivotação de linhas em Maple e Scilab Aplicações
16. Pivotação parcial com escalas Pivotação parcial com escalas e vetor de índices
17. Fatoração LU (1) Introdução à fatoração LU Exemplos passo a passo
18. Fatoração LU (2) Implementação de um programa de fatoração LU
19. Sistemas mal condicionados Introdução Exemplos Necessidade da computação algébrica para resolver sistemas mal condicionados
20. Método iterativo de Jacobi para sistemas lineares Descrição do método Análise de convergência Exemplos
21. Implementação do Método de Jacobi Implementação do método iterativo de Jacobi Aplicações
22. Método iterativo de Gauss-Seidel Introdução ao método iterativo de Gauss-Seidel Exemplos Análise de convergência
23. Implementação do Método de Gauss-Seidel Implementação do Método de Gauss-Seidel em Maple e Scilab Aplicações
24. Prova 1: Sistemas numéricos, erros e raízes de equações algébricas, sistemas lineares.
25. Interpolação linear Funções de base Exemplos com Scilab
26. Interpolação de Lagrange Base monomial Construção dos polinômios interpolantes de Lagrange Implementação em Maple e Scilab
27. Interpolação de Newton Introdução Implementação Aplicações
28. Diferenças divididas Definição Construção da tabela de diferenças divididas Determinação do grau do polinômio interpolante Erro na interpolação

Plano de ensino

Aplicações
29. Pontos de Chebyshev Fenômeno de Runge Pontos de Chebyshev
30. Splines Splines lineares, quadráticos e cúbicos Spline cúbico natural e not-a-knot
31. Método dos mínimos quadrados Ajuste de uma reta a pontos dados, usando o método dos mínimos quadrados. Ajuste de uma função da pontos dados, usando o método dos mínimos quadrados.
32. Sistemas não lineares Método iterativo de Newton para sistemas não-lineares.
33. Diferenciação Numérica Fórmulas avançadas e retardadas de diferenciação numérica
34. Extrapolação de Richardson Dedução do método e aplicação Implementação
35. Prova 2: Métodos diretos e iterativos, interpolação, ajuste, diferenciação numérica.
36. Métodos de Simpson-Kepler Métodos de Simpson 1/3 e 3/8 Dedução da fórmula de integração aproximada, simples e composta Fórmula de erro para a fórmula simples e composta Implementação
37. Método trapezoidal recursivo Dedução do método de integração numérica trapezoidal recursivo Implementação
38. Método de Romberg Dedução do método integração de Romberg Implementação
39. Método de Simpson adaptativo Teoria e implementação de um procedimento recursivo
40. Quadratura gaussiana Introdução ao método da quadratura gaussiana com nós arbitrários. Nós gaussianos
41. Método trapezoidal recursivo
42. Equações diferenciais ordinárias Método de Euler Implementação em Maple e Scilab Comandos de Maple para a solução de EDOs numéricas
43. Método da série de Taylor Método da série de Taylor para problemas de valor inicial Aplicação a EDOs de qualquer ordem.
44. Método de Euler modificado, método de Heun
45. Métodos de Runge-Kutta Métodos de segunda ordem para PVI de qualquer ordem Método de Runge-Kutta de quarta ordem
46. Métodos de Adams-Bashforth e Adams-Moulton
47. Implementação numérica e uso de bibliotecas para solução numérica de EDOs
48. Prova 3: Integração numérica e equações diferenciais.

Metodologia

1. Cada semana envolverá duas aulas teóricas e duas práticas (implementação computacional). O desenvolvimento inicial dos algoritmos básicos será feita em Maple. Na parte prática será preferida a implementação em um sistema numérico (C/C++, FORTRAN/Matlab/Python ou equivalentes). Das 72h da disciplina, 14h ocorrerão na modalidade de ensino à distância, distribuídas ao longo o semestre, através do Moodle. O aluno terá que assistir vídeos, seguir tutoriais, resolver e submeter tarefas no prazo estipulado. Serão estabelecidos 7 módulos no Moodle, cada um associado a 2h de carga horária.

Sistema de avaliação

Plano de ensino

- | |
|--|
| 1. As provas serão sem consulta e sua média aritmética terá peso de 90% na nota semestral. O restante será atribuído à média aritmética de trabalhos submetidos através do Moodle. |
|--|

Bibliografia básica

- | |
|---|
| 1. ALBRECHT, Peter. Análise Numérica - Um Curso Moderno. LTC e Editora da Universidade de São Paulo. 1973. Rio de Janeiro.
BARROS, Ivan de Queiroz. Introdução ao Cálculo Numérico. Ed. Edgard Blücher Ltda. 1972.
CLÁUDIO, Dalcídio Moraes; MARINS, Jussara Maria. Cálculo Numérico Computacional. São Paulo: Atlas, 1989. |
|---|

Bibliografia complementar

- | |
|--|
| 1. FORSYTHE, G. et al. Computer Methods for Mathematical Computations. Englewood Cliffs: Prentice-Hall, 1978.
HAMMER, R.; HOCKS, M.; KULISH, U. et al. Numerical Toolbox for Verified Computing I: Basic Numerical Problems. Berlin, Springer-Verlag, 1993.
JAJA, J. An Introduction to Parallel Algorithms. Reading: Addison-Wesley, 1992.
OLIVEIRA, P.W.; DIVERIO, T.A.; CLAUDIO, D.M. Fundamentos de Matemática Intervalar. Porto Alegre: Sagra-Luzzatto, 1999.
SANTOS, Vitorino Ruas de Barros. Curso de Cálculo Numérico. Ao Livro Técnico S.A.. 1972. Rio de Janeiro.
TRAUB, J.F. Complexity of Sequential and Parallel Numerical Algorithms. New York: Academic Press, 1973. |
|--|

Plano de ensino

Curso: CCI-BAC - Bacharelado em Ciência da Computação

Turma: CCI122-03U - CIENCIA DA COMPUTAÇÃO FASE 03U

Disciplina: AOC0003 - ARQUITETURA E ORG. DE COMPUTADORES

Período letivo: 2017/1

Carga horária: 72

Professor: 3398030 - EDINO MARIANO LOPES FERNANDES

Ementa

1. Funções e Portas Lógicas. Circuitos digitais Combinacionais e Sequenciais. Registradores. Noções de Arquitetura e Organização de computadores. Estrutura e funcionamento da CPU. Hierarquia de memórias. Conjunto, formato e armazenamento de instruções. Noções de linguagem Assembly. Dispositivos de entrada e saída. Pipeline de instruções. Arquiteturas RISC e CISC. Noções de processamento paralelo. Noções de Microcontroladores.

Objetivo geral

1. Introduzir conhecimentos básicos sobre portas lógicas e circuitos digitais e sua relação com os sistemas computacionais, capacitando o aluno a compreender o funcionamento e a organização dos computadores digitais, através da descrição dos elementos constituintes de um sistema de computação e da unidade central de processamento e da análise do fluxo elementar de seus dados, estrutura, organização e funcionamento geral dos computadores.

Objetivo específico

1. - Conhecer funções e portas lógicas digitais;
- Capacitar o aluno a compreender os circuitos digitais combinacionais e sequenciais básicos;
- Capacitar o aluno a entender o funcionamento geral de computadores digitais;
- Habilitar o aluno a compreender melhor a funcionalidade e utilização dos blocos constituintes de computadores digitais, bem como, dos dispositivos de entrada e saída;
- Compreensão da organização interna de arquiteturas de microprocessadores modernos;
- Apresentar software de baixo nível, seu conjunto de instruções, interrupções e modos de endereçamento;
- Fornecer uma visão geral de conceitos avançados em arquitetura de computadores.
- Introduzir conhecimentos de processamento paralelo e microcontroladores

Conteúdo programático

1. Apresentação do Plano de Ensino
2. Capítulo I - Circuitos Lógicos Combinacionais
 - 1.1 Revisão de sistemas numéricos e álgebra Booleana
3. 1.2 Portas lógicas
 - 1.3 Funções lógicas
4. 1.4 Circuitos Combinacionais
5. 1.5 Formas Canônicas
6. 1.6 Mapas de Karnaugh
7. 1.7 Projetos de Circuitos
8. 1.8 Circuitos Aritméticos
9. Atividades de Laboratório
10. Capítulo II - Circuitos Lógicos Sequenciais
 - 2.1 Flip-Flops
11. 2.2 Contadores
12. 2.3 Comparadores
 - 2.4 Registradores
13. Atividades de Laboratório
14. Apresentação de Trabalhos em Grupo
15. Resolução de Exercícios

Plano de ensino

16. Semana da Computação
17. Capítulo III - O Sistema de Computação
3.1 O conceito de Processamento de Dados
18. 3.2 Representação de informações
19. 3.3 Conceito de arquivos e registros
3.4 Medidas de desempenho de sistemas computacionais
20. 3.5 A arquitetura e Organização do computador
21. 3.6 A influência do Sistema Operacional
22. Capítulo IV - A Unidade Central de Processamento
4.1 A placa-mãe (Motherboard)
23. 4.2 Barramentos
24. 4.3 Circuitos de apoio (IRQ's, DMA, etc.)
4.4 Funções básicas da CPU
25. 4.5 A Unidade Lógica e Aritmética
4.6 Os registradores da CPU
26. 4.7 Função controle
4.8 Função Processamento
27. 4.9 Instruções de máquina
4.10 Formato das instruções
28. 4.11 Arquitetura RISC e CISC
4.12 Pipeline de instruções
29. 4.13 Linguagem de montagem (Assembly)
4.14 Classificação de instruções
4.15 Modos de endereçamento
30. 4.16 Instruções de transferência de dados
31. 4.17 Atividades de Laboratório
32. Capítulo V - Dispositivos de Armazenamento
5.1 Memórias
33. 5.2 Representação das informações nas memórias
5.3 Operações realizadas nas memórias
34. 5.4 Hierarquia de memórias
5.5 Memória principal
35. 5.6 Memória cache

Plano de ensino

5.7 Memória secundária
36. 5.8 Cálculo e Capacidade de memória
5.9 Tipos e nomenclatura de memórias
37. Capítulo VI - Noções de Processamento Paralelo
6.1 Organização SMP
38. 6.2 Clusters
6.3 Tipos de Sistemas com Processadores Paralelos
39. 6.4 Processadores Multicore
40. Capítulo VII - Noções de Microcontroladores
7.1 Conceitos
41. 7.2 SMP's
7.3 PIC's
7.4 FPGA
7.5 Arduíno
42. 7.4 Projetos com Microcontroladores
43. Capítulo VIII- Dispositivos de entrada e saída
8.1 Formas de Comunicação entre CPU e dispositivos de E/S
8.2 Teclado, Mouse, Joystick, Mesa digitalizadora, Scanner
44. 8.3 Impressoras, Plotter e Monitores de Vídeo
8.4 Fita Magnética, Discos Magnéticos, Óticos, CD, DVD, Cartão de memória, Pen drive, Unidades de Estado Sólido
45. 8.5 Motherboard
8.6 Padrões de barramento
46. 8.7 Microprocessadores PC e dispositivos móveis
8.8 Montagem e Manutenção de Microcomputadores
47. Avaliação Teórica

Metodologia

1. A disciplina será trabalhada através da temática Teórico-prática, com atividades realizadas em grupo, assim como uma forte carga de exercícios e aulas expositivas dialogadas.
As aulas serão seguidas de atividades práticas em laboratório. As leituras e os trabalhos em grupos serão significativos para elaboração de trabalho escrito e apresentado durante o transcorrer da disciplina. A disciplina poderá oferecer 20 % de sua carga horária na modalidade à distância.

Sistema de avaliação

1. Do desempenho do aluno:
A qualidade do desempenho do aluno será avaliada com base no desenvolvimento das seguintes atividades e com os seguintes critérios:
 - a) Provas sobre o(s) capítulo(s) teóricos (P1 e P2) - (60 %)
 - b) Elaboração e apresentação de trabalho em grupo (TG) - (20 %)
 - c) Escrita e Apresentação de Artigo (AR) - (10%)
 - d) Relatório de experiências de laboratório (RE) - (10%)

Do desempenho da disciplina e do professor:

Plano de ensino

Os discentes terão, igualmente, a oportunidade de fazer uma avaliação mais completa do desempenho do professor e da disciplina através do sistema de avaliação eletrônico.

Bibliografia básica

1. MELO, M. Eletrônica Digital. Makron Books.2003.
STALLINGS, W. Arquitetura e Organização de Computadores. 8 ed. Prentice Hall. São Paulo, 2010.
TANENBAUM, Andrew S. Organização estruturada de computadores. 5. ed. São Paulo: Pearson, 2007.

Bibliografia complementar

1. ZUFFO, J.A. Fundamentos de Arquitetura e Organização de Microprocessadores. Edgard Blücher.
MALVINO, A. Microcomputadores e Microprocessadores. Ed. McGraw-Hill do Brasil, 1985.
HENNESSY J. L.; PATTERSON, D. A. Arquitetura de Computadores - Uma Abordagem Quantitativa. Rio de Janeiro: Ed. Campus, 2003.

Plano de ensino

Curso: CCI-BAC - Bacharelado em Ciência da Computação
Turma: CCI122-03U - CIENCIA DA COMPUTAÇÃO FASE 03U
Disciplina: EDA0001 - ESTRUTURA DE DADOS
Período letivo: 2017/1
Carga horária: 72
Professor: 3321045 - GILMARIO BARBOSA DOS SANTOS

Ementa
1. Representação e manipulação de tipos abstratos de dados. Estruturas lineares. Introdução a estruturas hierárquicas. Métodos de classificação. Análise de eficiência. Aplicações.

Objetivo geral
1. capacitar o aluno a desenvolver soluções computacionais eficientes através da utilização de algoritmos eficientes e estrutura(s) de dados adequada(s).

Objetivo específico
1. Implementar as principais estruturas de dados (lista, fila, pilha, árvore) Analisar os principais algoritmos que tratam conjuntos de dados (ordenação, busca) Capacitar os alunos a avaliar o melhor algoritmo para solucionar certo problema

Conteúdo programático
1. Apresentação da disciplina
2. Revisão da Linguagem C: ponteiros, aritmética de ponteiros, ponteiro-para-ponteiro
3. Revisão da Linguagem C: manipulação de structs, unions, vetor, encadeamento simples/duplo
4. Introdução ao modelo TDA
5. TDA-PE: Criação, buscaNoTopo, destruição, remoção, inserção, testes de status
6. TDA PE: Criação, buscaNoTopo, destruição, remoção, inserção, testes de status Laboratório TDA-PE: compilação dos códigos disponibilizados
7. TDA PDSE/PDDE
8. TDA-MPE: Multi-pilha: uso de Union, inicialização, acesso a metadados do TDA-MPE
9. TDA-MPE
10. Revisão e exercícios gerais
11. Revisão e exercícios gerais
12. Prova #1
13. Filas Estáticas: adaptando a partir das pilhas
14. FDSE/FDDE: adaptadas a partir da PDSE/PDDE
15. Variedades de FDSEs
16. Fila de Prioridade
17. Aplicações da Fila de prioridade como simulador de fila bancária
18. Exercícios.
19. prova#2
20. Listas LDSE/LDDE
21. Listas: LESE
22. Multilistas: Grafos, matrizes esparsas... Variações de LDDEs

Plano de ensino

23. Exercícios..
24. Exercícios: Listas
25. Prova#3
26. Árvores em geral: conceitos...
27. ABB: busca sequencial, ABB versus busca binária
28. ABB-AVL
29. B-tree
30. Ordenação/Especificação do trabalho Alg. de Ordenação Especificação do trabalho final
31. Exercícios
32. Exercícios
33. Prova#4
34. Orientação de trabalho
35. Orientação de trabalho final
36. Entrega/entrevista de trabalho
37. Resultados finais
38. Unio-and-finding
39. Bloom-filter
40. skip-list
41. Exercícios
42. Prova#5
43. Avaliação escrita
44. Outra atividade, especificada no complemento.
45. ProvaDeRevisãoDaLingC

Metodologia

1. A disciplina será ministrada através de aula expositivas da teoria e aulas em laboratório de forma a provocar discussões acerca dos conteúdos e produzir implementações (programação) de soluções para os problemas propostos.

Vinte por cento da carga horária da disciplina poderá ser desenvolvido com auxílio do sistema Moodle (moodle2.joinville.udesc.br)

Sistema de avaliação

1. O desempenho do aluno será avaliado com base no desenvolvimento das seguintes atividades e com os seguintes critérios:
 - a) participação ativa nas aulas e nos trabalhos;
 - b) elaboração e apresentação de trabalhos individuais ou em grupo;
 - c) avaliações individuais (provas e trabalhos). A média semestral (MS) é calculada sobre as notas de pelo menos 3 avaliações as quais incluem pelo menos um trabalho prático.A média MT das notas dos trabalhos pode ser ponderada de acordo com o grau de dificuldade dos mesmos. Não está descartada a atribuição de nota (extra) como reconhecimento à participação destacada e de qualidade do(a) aluno(a) durante as atividades do curso;

Plano de ensino

d) não é permitido o uso de aparelhos de comunicação (celulares e similares) durante as provas.

Bibliografia básica

1. Tenenbaum, Aaron M. et al. Estruturas de Dados Usando C. Ed. Makron Books.
Horowitz, Ellis. & Sahni, Sartaj. Fundamentos de Estruturas de Dados. Editora Campus.
Szwarcfiter, J. L. et al. Estruturas de Dados e seus Algoritmos. Ed. LTC.

Bibliografia complementar

1. Aitken, P. & Jones, B. Guia do Programador C, Ed Berkeley Brasil.
Azeredo, P. A. Métodos de Classificação de Dados e Análise de suas Complexidades. Ed. Campus
Cormen, Thomas H. et al. Introduction to Algorithms. MIT Press.
Kernigham, B. W. A Linguagem de Programação C, Ed. Campus.
Preiss, Bruno R. Estruturas de Dados e Algoritmos - Padrões e projetos orientados a objetos com Java, Editora Campus.
Schildt, H. C Avançado - Guia do Usuário, Ed. McGraw Hill.
Schildt, H. C Completo e Total, McGraw Hill
Ward, R. Depurando em C, Ed Campus.
Veloso, Paulo. et al. Estruturas de Dados. Editora Campus.
Como também o próprio material disponibilizado no sítio da disciplina.

Plano de ensino

Curso: CCI-BAC - Bacharelado em Ciência da Computação

Turma: CCI122-03U - CIENCIA DA COMPUTAÇÃO FASE 03U

Disciplina: LFA0001 - LINGUAGENS FORMAIS E AUTÔMATOS

Período letivo: 2017/1

Carga horária: 72

Professor: 1033139444 - KARINA GIRARDI ROGIA

Ementa

1. Alfabetos e Linguagens. Linguagens, gramáticas e expressões regulares, autômatos finitos. Linguagens e gramáticas livres de contexto e autômatos de pilha. Linguagens sensíveis ao contexto. Implementação dos conceitos para a solução de problemas básicos.

Objetivo geral

1. Capacitar o aluno para a aplicação sistematizada e formalizada de conceitos e resultados relativos às linguagens, gramáticas, autômatos e reconhecedores.

Objetivo específico

1. Apresentar os principais métodos de tratamento sintático de linguagens abstratas, com a respectiva associação às linguagens típicas da Ciência da Computação;
Conceituar as linguagens regulares, autômatos finitos e expressões regulares;
Apresentar a equivalência entre os modelos;
Conceituar autômatos a pilha determinísticos e não determinísticos;
Conceituar os conceitos sobre gramáticas livres de contexto;
Introduzir os conceitos de linguagens sensíveis ao contexto;
Apresentar a hierarquia de Chomsky

Conteúdo programático

1. Contextualização das linguagens formais e conceitos básicos.
2. Alfabeto, palavra, linguagem e gramática.
3. Autômato Finito Determinístico
4. Autômato Finito Não-Determinístico
5. Autômato Finito com Movimentos Vazios
6. Expressões Regulares
7. Gramáticas Regulares
8. Lema do Bombeamento
9. Propriedades de Linguagens Regulares
10. Minimização de Autômatos Finitos
11. Autômatos Finitos com Saída
12. Gramáticas Livre de Contexto (GLC)
13. Árvores de Derivação
14. GLC Ambíguas
15. Simplificação de GLC
16. Formas Normais
17. Autômato com Pilha
18. Propriedades de LLC
19. Algoritmos de Reconhecimento: Autômato Descendente
20. Algoritmos de Reconhecimento: CYK
21. Algoritmos de Reconhecimento: Early
22. Máquina de Turing (geral e com fita limitada)

Plano de ensino

23. Linguagens Sensíveis ao Contexto

24. Gramáticas Irrestitas

25. Hierarquia de Chomsky

26. Gramáticas de Grafos

Metodologia

1. A disciplina será desenvolvida através de aulas expositivo-dialogadas, com exercícios práticos orientados em sala de aula, além de desenvolvimento extra-classe de trabalho final prático. O conteúdo da disciplina poderá ser ministrado na modalidade de ensino a distância em até 20% do total de sua Carga Horária (MEC PORTARIA Nº 4.059, DE 10 DE DEZEMBRO DE 2004 publicado no DOU de 13/12/2004, Seção 1, p. 34).

Sistema de avaliação

1. Do desempenho do aluno:
A qualidade do desempenho do aluno será avaliada com base no desenvolvimento das seguintes atividades e com os seguintes critérios:
a) Provas individuais (P1, P2 e P3).
b) Trabalho final (T)
A Média Semestral (MS) será calculada pela fórmula
$$MS = (P1 + P2 + P3 + T)/4$$

Do desempenho da disciplina e do professor:
Os estudantes terão, igualmente, a oportunidade de fazer uma avaliação do desempenho do professor e da disciplina. As informações sobre esta atividade serão fornecidas pelo coordenador do curso.

Das regras para revisão das avaliações:
Depois da publicação das notas pelo professor, os alunos têm 07 dias corridos para solicitar a revisão com o professor. Esta revisão será feita na sala do professor, preferencialmente em horário de atendimento aos alunos, ou em um horário do qual o professor possa atendê-los.

Bibliografia básica

1. HOPCROFT, J. E., ULLMAN, J. D. e MOTWANI, R. Introdução à Teoria de Autômatos, Linguagens e Computação. Ed. Campus, 2002.
MENEZES, P. F. B. Linguagens Formais e Autômatos. Série Livros Didáticos nº3. 4ª edição. Ed. Sagra Luzzato, 2002.
VIEIRA, N. J. Introdução aos Fundamentos da Computação: Linguagens e Máquinas. 1a ed.: Rio de Janeiro: Thomson, 2006.

Bibliografia complementar

1. SIPSER, M. Introdução à Teoria da Computação. 2a ed. São Paulo: Cengage Learning, 2012.
SUDKAMP, T. A. Languages and Machines: An introduction to the Theory of Computer Science. Second edition. Addison Wesley, 1997.

Plano de ensino

Curso: CCI-BAC - Bacharelado em Ciência da Computação

Turma: CCI122-03U - CIENCIA DA COMPUTAÇÃO FASE 03U

Disciplina: MEP0003 - METODOLOGIA DA PESQUISA

Período letivo: 2017/1

Carga horária: 36

Professor: 9630970 - Maristela de Souza Borba Severino

Ementa

1. Pesquisa tecnológica. Ciência e tecnologia. Criação e absorção da tecnologia. Métodos de pesquisa. Projeto de pesquisa. Fases do projeto. Comunicação científica.

Objetivo geral

1. Oportunizar aos acadêmicos conhecimentos em metodologia da pesquisa, fundamentais para produção de trabalhos científicos e práticas discentes em pesquisa.

Objetivo específico

1. Refletir acerca da relação entre pesquisa tecnológica, ciência, criação e absorção da tecnologia. Conhecer, identificar e usar os métodos de pesquisas em atividades, no projeto e no relatório da pesquisa. Conhecer e usar as normalizações do trabalho científico em atividades, no trabalho da pesquisa, tais como formatações/produções dos elementos pré-textuais, textuais e pós-textuais, páginas, títulos, letras, citações, referências, segundo as normas da Associação Brasileira de Normas Técnicas. Produzir, em equipe um projeto de pesquisa teórica e/ou documental, apresentá-lo escrita e oralmente e desenvolvê-lo. Apresentar, em equipe e em forma escrita e oral, o relatório da pesquisa teórica e/ou documental desenvolvida. Comunicar-se adequada e eficazmente na apresentação dos trabalhos científicos.

Aulas em 2017-1: terça-feira das 13:30 às 15:10 - duração da hora-aula: 50 minutos

Fevereiro: 21

Março: 07, 14, 21, 28

Abril: 04, 11, 18, 25

Maior: 02, 09, 16, 23, 30

Junho: 06, 13, 20, 27

Exame final: 11/07/2017

Conteúdo programático

1. Pesquisa tecnológica: ciência e pesquisa, criação e absorção da tecnologia.
2. Pesquisa: teórica, documental, campo; exploratória, descritiva, explicativa.
3. O projeto e o relatório da pesquisa: elementos pré-textuais, textuais, pós-textuais; obrigatórios e opcionais.
4. Pesquisa-piloto / pré-teste; validade interna e externa; confiabilidade interna e externa; triangulação.
5. Tipos de pesquisas/método(s) de pesquisa - método(s) de procedimento(s): estudo de caso, ex-post facto, base etnográfica, pesquisa-ação, histórico, comparativo, tipológico, estruturalista, estatístico.
6. Metodologia e métodos da pesquisa - métodos de abordagens: base indutiva, base dedutiva, dialético, hipotético-dedutivo.
7. Metodologia e métodos da pesquisa - formas de abordagens dos dados: qualitativa e/ou quantitativa.
8. Planejamento/projeto da pesquisa e fases:
 - elementos pré-textuais: capa; folha de rosto; resumo e palavras-chave; sumário;
 - elementos textuais: introdução: tema, problema/pergunta, hipótese(s) e variáveis, justificativa, objetivos: geral e específicos; fundamentação teórica; abordagem metodológica: metodologia - método(s) de procedimento(s), método de abordagem, natureza do estudo: básica ou aplicada - contexto do estudo ou objeto de investigação do estudo, instrumentos da coleta dos dados, procedimentos na análise dos dados; cronograma;
 - elementos pós-textuais: referências; apêndices - se houver; anexos - se houver.
9. Normalização do trabalho científico, segundo normas da ABNT: estrutura; formatações das páginas, títulos, letras.
10. Normalização do trabalho científico, segundo normas da ABNT: citações, plágio e autoplágio.
11. Normalização do trabalho científico, segundo normas da ABNT: citações e referências.
12. O relatório da pesquisa:
 - elementos pré-textuais: capa; folha de rosto; resumo e palavras-chave; sumário;
 - elementos textuais: introdução: tema, problema/pergunta, hipótese(s) e variáveis, justificativa, objetivos: geral e específicos; fundamentação teórica; abordagem metodológica: metodologia - método(s) de procedimento(s), método de abordagem, natureza do estudo: básica ou aplicada - contexto do estudo ou objeto da investigação do estudo, instrumentos da

Plano de ensino

coleta dos dados, procedimentos na análise dos dados, cronograma; análise e interpretação dos dados; considerações finais: resposta da(s) pergunta(s), implicações, limitações do estudo, sugestões de estudos futuros;
--- elementos pós-textuais: referências; glossário - opcional; apêndices - se houver/opcional; anexos - se houver/opcional, índice - opcional.

13. Comunicação do trabalho científico.

Metodologia

1. As aulas são expositivo-dialogadas por meio de leituras, debates de textos científicos e ilustradas por meio de recursos audiovisuais. As atividades escritas/orais são realizadas individualmente e/ou em equipe. Há apresentações orais em equipes dos projetos e dos relatórios das pesquisas escritos com o uso dos recursos disponibilizados, tais como materiais impressos, textos verbais e não verbais, quadro, computador/datashow, som, vídeos de curta duração. Haverá o uso do moodle na postagem/entrega dos trabalhos da pesquisa: projeto e relatório da pesquisa escritos e slides.

Sistema de avaliação

1. 1 - AV_1 - PROVA_1 - Valor 10,0 (dez), peso 2,0 (dois): avaliação escrita individual e sem consulta de todo o conteúdo trabalhado. Data: 18/04/2017.

2 - AV_2 - ATIV - Valor 10,0 (dez), peso 1,0 (um): atividades realizadas durante o semestre letivo em casa e/ou na sala de aula/moodle, escritas segundo orientações informadas, entregues nas datas estipuladas na sala de aula e/ou na página on line da professora no SIGA UDESC. A nota final é a média das notas obtidas nas atividades solicitadas/realizadas. A nota/feedback de cada trabalho/atividade será semanal, ou seja, na próxima aula e a nota final/média será lançada no SIGA UDESC até 30/05/2017. Todo trabalho deve ter o registro da data de entrega, cabeçalho ou capa, segundo materiais trabalhados, avisos e outras informações disponibilizadas na sala de aula e/ou página on line da professora. Não são considerados trabalhos entregues fora do prazo, enviados por e-mail, plagiados parcial ou integralmente.

3 - AV_3 - PESQUISA - Valor 10,0 (dez), peso 3,0 (três): produção escrita e apresentação oral de uma pesquisa teórica e/ou documental. Atividade realizada em equipe.
-- 14/03/2017: os acadêmicos devem entregar à professora na sala de aula os nomes completos dos integrantes das equipes.
-- 03/04/2017: os acadêmicos devem postar/informar no moodle UDESC-CCT até as 14 horas:
--- capa,
--- folha de rosto,
--- palavras-chave,
--- introdução: tema, problema/pergunta, hipótese(s) - se houver; justificativa; objetivos: geral e específicos,
--- referências: inicialmente, pelo menos, cinco obras da coleta/análise dos dados da pesquisa.
-- 28/04/2017 - até as 14 horas: todos os acadêmicos da equipe devem entregar/postar no moodle UDESC-CCT o projeto da pesquisa escrito.
-- 28/04/2017 - até as 14 horas: todos os acadêmicos da equipe devem postar no moodle UDESC-CCT os slides da apresentação oral do projeto da pesquisa escrito.
-- 02/05/2017 - a partir dessa data: apresentar oralmente o projeto da pesquisa escrito.
-- 05/06/2017 - até as 14 horas: todos os acadêmicos da equipe devem entregar/postar no moodle UDESC-CCT o relatório da pesquisa escrito.
-- 05/06/2017 - até as 14 horas: todos os acadêmicos da equipe devem postar no moodle UDESC-CCT os slides da apresentação oral do relatório da pesquisa escrito.
-- 06/06/2017 - a partir dessa data: apresentar oralmente o relatório da pesquisa escrito.

-- Projeto da pesquisa: 10% da nota; relatório da pesquisa: 90% da nota.
-- Texto escrito: 70% da nota; apresentação oral: 30% da nota.
-- A pesquisa teórica e/ou documental deve ser produzida com materiais/textos públicos e sem restrições de usos.
-- Todos os integrantes/membros da equipe devem postar no moodle UDESC-CCT, ou seja, enviar via moodle UDESC-CCT - o mesmo trabalho da equipe - projeto e relatório da pesquisa e seus slides - e somente em formato doc para texto e ppt para slides.
-- Não são considerados: (1) trabalhos diferentes de uma mesma equipe, (2) entregues fora do prazo, (3) enviados por e-mail, (4) plagiados parcial ou integralmente.
-- Todos os integrantes da equipe devem participar/contribuir igualmente nas apresentações orais e produções escritas do projeto da pesquisa e relatório da pesquisa.
-- As orientações são realizadas em equipe.
-- O projeto da pesquisa e o relatório da pesquisa devem estar de acordo com as normas da ABNT e metodológicas trabalhadas na disciplina, bem como modelos/informações disponibilizado(a)s na página on line da professora no SIGA UDESC, inclusive materiais 2, 3, 4, 5, 6, 7.
Material 2: Manual para elaboração e normalização de trabalhos de conclusão de curso da UDESC: graduação e pós-graduação- versão 2016.
Material 3: projeto da pesquisa - modelo e orientações.
Material 4: projeto da pesquisa - slides.
Material 5: relatório da pesquisa - modelo e orientações.
Material 6: relatório da pesquisa - slides.
Material 7: orientações para a escrita acadêmica - atualizado durante o curso: cabeçalho, capa e escrita acadêmica.

4 - AV_4 - PROVA_2 - Valor 10,0 (dez), peso 4,0 (quatro): avaliação escrita individual e sem consulta dos conteúdos a serem

Plano de ensino

informados durante o semestre letivo. Data: 27/06/2017.

Para ser aprovado sem exame final, o acadêmico(a) deve ter, pelo menos, 75% de frequência do total de 36 horas-aula e nota/média semestral sete.

Exame final: 11/07/2017 de todos os conteúdos trabalhados no curso. Média final para aprovação: 5,0 (cinco), segundo normativas UDESC.

Observações

- 1 - Todos acadêmicos devem se cadastrar/inscrever no moodle UDESC-CCT. Orientações para cadastro no moodle: ir em , entrar em serviços, depois em portal corporativo, CINF-Suporte, Serviços, CADASTRO ON-LINE PARA TODOS ALUNOS - REDE SEM FIO/VPN CAPES/MOODLE2 (Formulário). Preencher o formulário - CPF + senha de acesso ao SIGA UDESC- e depois de duas horas, acessar o moodle, ou seja, ir em /moodle2.joinville.udesc.br/>, entrar no seu curso/disciplina - DCC/Metodologia da Pesquisa - MEP0003, preencher nome do usuário que é seu CPF e sua senha - a que você utiliza para entrar no SIGA UDESC - fazer o autocadastro e navegar pela página.
- 2 - O plano de ensino ou material um, materiais, cronograma estão disponibilizados no moodle UDESC-CCT.
- 3 - Parâmetros para a entrega de cada atividade solicitada: o texto deve estar digitado e impresso com letra tamanho 12, arial ou times new roman, cabeçalho ou capa conforme material 2, ou seja, Manual para elaboração e normalização de trabalhos de conclusão de curso da UDESC: graduação e pós-graduação (2016); registrar a data de entrega, bem como a(s) perguntas/enunciados e a(s) respostas; todo parágrafo deve estar justificado; texto: coeso, ou seja, com concordâncias adequadas à língua padrão; coerente, isto é, fazer sentido ao leitor; bem argumentado/fundamentado; com as pontuações adequadas; escrito segundo as normas da Associação Brasileira de Normas Técnicas (ABNT); apresentar a(s) citação(ões) e referência(s) textual(is), bem como elementos pré-textuais, textuais e pós-textuais adequados à produção, segundo normas da ABNT e orientações do curso.
- 4 - Qualquer trabalho plagiado/copiado parcialmente ou totalmente é desqualificado sem direito a reescrita/reapresentação.
- 5 - Falta na data de qualquer avaliação ou atividade-avaliação deve ser justificada na secretaria acadêmica do curso do aluno e solicitada a segunda chamada dessa avaliação no prazo previsto para essa solicitação, segundo normativas UDESC.
- 6 - Informações sobre frequências, solicitações de prova em segunda chamada encontram-se no portal do acadêmico, ou seja, entrar em , depois ir em ensino, portal do acadêmico, resoluções acadêmicas.

Bibliografia básica

1. DEMO, Pedro. Pesquisa. 2. ed. São Paulo : Cortez, 1991.
GALLIANO, Alfredo Guilherme. O método científico: teoria e prática. São Paulo: Harbra & Row do Brasil, 1986.
MARCONI, Marina de Andrade; LAKATOS, Eva Maria. Fundamentos da metodologia científica. 6. ed. São Paulo: Atlas, 2005.

Bibliografia complementar

1. APPOLINÁRIO, Fábio. Metodologia da ciência. São Paulo: Cengage Learning, 2011.
BAPTISTA, Makilim Nunes; CAMPOS, Dinael Corrêa de Campos. Metodologias de pesquisa em ciências: análises quantitativas e qualitativas. Rio de Janeiro: LTC, 2015.
BARROS, Aidil Jesus da Silveira; LEHFELD, Neide Aparecida de Souza. Fundamentos de metodologia científica: um guia para a iniciação científica. 2. ed. São Paulo: Makron Books, 2000.
CERVO, Amado Luiz; BERVIAN, Pedro Alcino. Metodologia científica. 4 ed. São Paulo: Makron Books do Brasil, 1996.
CRUZ, Carla; RIBEIRO, Uirá. Metodologia científica: teoria e prática. Rio de Janeiro: Axcel Books do Brasil, 2003.
FACHIN, Odília. Fundamentos de metodologia. 4. ed. São Paulo: Saraiva, 2003.
GIL, Antônio Carlos. Como elaborar projetos de pesquisa. São Paulo: Atlas, 1991.
HEERDT, Mauri Luiz; LEONEL, Vilson. Metodologia científica e da pesquisa: livro didático. Palhoça: UnisulVirtual, 2007. Disponível em: . Acesso em: 12 jul. 2016.
JUNCKES, Rosane Santana; OLIVEIRA, Nádia Fátima de. Multiculturalismo: metodologia da pesquisa científica. São Paulo: Know How, 2010.
KÖCHE, José Carlos. Fundamentos da metodologia científica: teoria da ciência e prática da pesquisa. 14. ed. Petrópolis, RJ: Vozes, 1997.
NERY, Guilherme et al. Nem tudo que parece é: entenda o que é plágio. Niterói, RJ: [Universidade Federal Fluminense, 2010?]. Cartilha. Disponível em: /www.noticias.uff.br/arquivos/cartilha-sobre-plagio-academico.pdf>. Acesso em: 8 fev. 2017.
PINHEIRO, José Maurício dos Santos. Da iniciação científica ao TCC: uma abordagem para os cursos de tecnologia. Rio de Janeiro: Ciência Moderna Ltda., 2010.
RUIZ, João Álvaro. Metodologia científica: guia para eficiência nos estudos. 5. ed. São Paulo: Atlas, 2002.
SALOMON, Dêlcio Vieira. Como fazer uma monografia. 12. ed. São Paulo: Saraiva, 2010.
SEVERINO, Antônio Joaquim. Metodologia do trabalho científico. 20. ed. São Paulo: Cortez, 1996.
SILVA, Everaldo da et al. Metodologia do trabalho acadêmico. 3. ed. Curitiba: Juruá, 2010. WAZLAWICK, Raul Sidnei. Metodologia da pesquisa para ciência da computação. 2. ed. Rio de Janeiro: Elsevier, 2014.
UNIVERSIDADE DO ESTADO DE SANTA CATARINA. Manual para elaboração e normalização de trabalhos de conclusão de curso da UDESC: graduação e pós-graduação. Florianópolis: UDESC, 2016. Disponível em: /www.cct.udesc.br/arquivos/id_submenu/2123/manual_de_trabalhos_academicos_da_udesc.pdf>. Acesso em: 8 fev. 2017.

Plano de ensino

Curso: CCI-BAC - Bacharelado em Ciência da Computação

Turma: CCI122-03U - CIENCIA DA COMPUTAÇÃO FASE 03U

Disciplina: POO0001 - PROGRAMAÇÃO ORIENTADA A OBJETOS

Período letivo: 2017/1

Carga horária: 72

Professor: 3198057 - CARLA DIACUI MEDEIROS BERKENBROCK

Ementa

1. Conceitos de orientação a objetos. Decomposição de programas. Generalização e especialização. Agregação e composição. Herança e polimorfismo. Projeto orientado a objetos. Estudo de uma linguagem.

Objetivo geral

1. Oferecer meios para que o aluno entenda o funcionamento e conceitos básicos sobre programação orientada a objetos, projete um sistema orientado a objetos e conheça a linguagem de programação Java.

Objetivo específico

1. - Dar subsídios aos acadêmicos para compreenderem o funcionamento e utilização dos conceitos associados a POO, como troca de mensagens, atributos, associações, agregações e composições;
- Apresentar aos acadêmicos os conceitos básicos sobre generalizações, interfaces, classes abstratas, polimorfismo e outros;
- Introduzir uma linguagem de programação orientada a objetos;
- Permitir que os acadêmicos apliquem os conceitos aprendidos na linguagem Java;
- Capacitar ao desenvolvimento de programas orientados a objeto.

Conteúdo programático

1. Introdução
Modelos
Programação OO
Classes, Objetos, instâncias, referências e métodos
2. Criando classes e aplicações
Sintaxe básica
Métodos em classes
Modificadores de acesso
Escrevendo classes e aplicações interativas
3. Programação Orientada a objetos
Herança
Polimorfismo
Notação
Exercícios Resolvidos
4. Desenvolvimento em Java
Generalização e especialização
Agregação e composição
Herança
5. Projeto orientado a objetos
Realização de Projeto

Metodologia

1. A disciplina será trabalhada utilizando a temática teórico-prática, com ênfase a discussão ao debate em grupo, assim como uma forte carga de leitura e aulas expositivas complementaram o trabalho. O conteúdo da disciplina poderá ser ministrado na modalidade de ensino a distância em até 20% do total de sua Carga Horária (MEC PORTARIA No 4.059, DE 10 DE DEZEMBRO DE 2004 publicado no DOU de 13/12/2004, Seção 1, p. 34). Serão aplicadas técnicas de aprendizado colaborativo, envolvendo dispositivos móveis, visando a fixação do conteúdo.

Sistema de avaliação

1. O desempenho do aluno será avaliado com base no desenvolvimento das seguintes atividades e com os seguintes critérios:
a) participação ativa nas aulas (avaliação que auxilia o aluno nas notas das avaliações abaixo)
b) Avaliações Escritas Individuais (AE1 e AE2).
c) Exercícios (EXE)
d) Seminários (SEM)
e) Trabalho prático em dupla, com avaliação individual (TP1)

Plano de ensino

Bibliografia básica

1. DEITEL, H. M.; DEITEL, P. J. Java: Como Programar. 4 ed. Porto Alegre : Bookman, 2003.
HORSTMANN, C. S.; CORNELL, G. Core Java 2. São Paulo : Makron Books, 2003.
BOOCH, G.; RUMBAUGH, J.; JACOBSON, I. UML Guia do Usuário. Rio de Janeiro : Campus, 2000.

Bibliografia complementar

1. SANTOS, R. Introdução à Programação Orientada a Objetos usando Java. Rio de Janeiro : Elsevier, 2003.
LARMAN, C. Utilizando UML e padrões. Porto Alegre : Bookman, 2002.
METSKER, S. J. Padrões de Projeto em Java. Porto Alegre : Bookman, 2004.
HAGGAR, Peter. Practical Java: Programming Language Guide. Reading, MA : Addison Wesley, 2000.
ARNOLD, K., HOLMES, D. The Java programming language. 3 ed. Boston, MA : Addison Wesley, 2000.
PAGE-JONES, M; PASCHOA, C. R. Fundamentos do desenho orientado a objeto com UML. São Paulo : Makron Books, 2001.

Plano de ensino

Curso: CCI-BAC - Bacharelado em Ciência da Computação

Turma: CCI122-03U - CIENCIA DA COMPUTAÇÃO FASE 03U

Disciplina: PPR0001 - PROJETO DE PROGRAMAS

Período letivo: 2017/1

Carga horária: 36

Professor: 211020727 - DIEGO BUCHINGER

Ementa

1. Modularização. Coesão e acoplamento. Métodos baseados em dados. Métodos baseados no tempo. Métodos baseados em funções. Métodos baseados em objetos. Padrões de Projeto.

Objetivo geral

1. Apresentar aos alunos os métodos para projeto e modularização de programas.

Objetivo específico

1. - Entendimento dos aspectos fundamentais do projeto de programas
- Compreensão dos diferentes tipos de arquiteturas de sistemas
- Domínio das principais técnicas para projeto e modularização de sistemas
- Compreensão de padrões de projeto

Conteúdo programático

1. 1. Plano de Ensino e Método de Avaliação
2. 2. Fundamentos de projeto de programas
 - 2.1. Introdução e posicionamento
 - 2.2. Processo de Software
 - 2.3. Aspectos fundamentais do projeto de programas
3. 3. Projeto de Dados
 - 3.1 Métodos de Projeto Orientado a Dados
 - 3.2 Conceitos básicos: Entidades, relações e atributos
 - 3.3 Diagramas Entidade-Relacionamento
4. 4. Projeto Arquitetural
 - 4.1 Conceitos Básicos / revisão orientação a objetos
 - 4.2 Organização do Sistema
 - 4.3 Estilos de decomposição modular
 - 4.4 Modelos de controle
 - 4.5 Diagramas de classes (UML)
5. 5. Projeto Procedimental
 - 5.1 Diagramas de eventos
 - 5.2 Diagramas de estados
 - 5.3 Diagrama de Fluxo de Dados
6. 6. Projeto de Interfaces
7. 7. Padrões de Projeto (Design Patterns) e boas práticas

Metodologia

1. Aulas expositivas com atividades práticas para fixação. Desenvolvimento de um trabalho completo envolvendo projeto e implementação de um sistema.

Sistema de avaliação

1. Os alunos serão avaliados através das seguintes atividades e pesos:
 - * Exercícios: [14%]
 - Fundamentos e Introdução [2%]
 - Requisitos [2%]
 - Projeto de dados [2%]
 - Diagrama de fluxo de dados [2%]
 - Projeto arquitetural básico [2%]

Plano de ensino

- Projeto em camadas [2%]
- Projeto de interface [2%]
- *Pesquisa: [8%]
- Relatório sobre UML [8%]
- *Trabalho Prático: [58%]
- Requisitos e diagramas iniciais [5%]
- Entrega do projeto final [20%]
- Plano de implementação e testes [2%]
- Anotações da execução da implementação e teste [2%]
- Entrega da implementação e apresentação [20%]
- Avaliação do projeto e implementação da outra equipe [3%]
- Nota da avaliação (por outra equipe) do projeto [3%]
- Nota da avaliação (por outra equipe) da implementação [3%]
- * Prova [20%]

Bibliografia básica

1. BEZERRA, E. Princípios de Análise e Projetos de Sistemas com UML. Rio de Janeiro: Campus, 2003.
PRESSMAN, R.S. Engenharia de Software. São Paulo: Makron Books, 2002.
SOMMERVILLE, I. Engenharia de Software. São Paulo: Addison Wesley, 2003.

Bibliografia complementar

1. WARNIER, J. Lógica de Construção de Programas. Rio de Janeiro: Campus, 1984.
JACKSON, M. Princípios de Projeto de Programas. Rio de Janeiro: Campus, 1988.
PAGE-JONES, M. Projeto Estruturado de Sistemas. São Paulo: McGraw-Hill, 1988..

Plano de ensino

Curso: CCI-BAC - Bacharelado em Ciência da Computação

Turma: CCI122-04U - CIENCIA DA COMPUTAÇÃO FASE 04U

Disciplina: AMS0001 - ANÁLISE E MODELAGEM DE SISTEMAS

Período letivo: 2017/1

Carga horária: 36

Professor: 9672753 - LEANDRO ISRAEL PINTO

Ementa

1. Ciclo de vida; Paradigmas de desenvolvimento; Análise de sistemas de informação; Análise de sistemas de tempo real; Ferramentas de modelagem.

Objetivo geral

1. Capacitar o aluno a compreender o funcionamento e conceitos básicos sobre análise e modelagem de sistemas, formular e interpretar os artefatos existentes.

Objetivo específico

1. - Capacitar ao desenvolvimento de artefatos de sistema
- Noções básicas sobre análise e modelagem de sistema
- Desenvolvimento de artefatos de sistema

Conteúdo programático

1. 1. Visão geral
2. 2. Processo de desenvolvimento de Software
 - 2.1 Desenvolvimento em Cascata
 - 2.2 Desenvolvimento Iterativo Incremental
 - 2.2 Técnicas de Desenvolvimento
 - 2.3 Desenvolvimento Ágil
3. 3. Requisitos de Software
4. 4. Modelagem UML
 - 4.1 Introdução
 - 4.2 Estruturação Básica
 - 4.3 Classes, relações e mecanismos básicos
5. 5. Diagramas UML
 - 5.1 Diagrama de Caso de Uso
 - 5.2 Diagrama de Classes
 - 5.3 Diagrama de Pacotes
 - 5.4 Diagrama de Sequência
 - 5.5 Diagrama de Máquina de Estados
 - 5.6 Diagrama de Atividades
 - 5.7 Diagrama de Componentes
 - 5.8 Diagrama de Implantação
6. 6. Modelagem Ágil

Metodologia

1. A disciplina será trabalhada através da temática Teórico-prática, com exercícios e aulas expositivas dialogadas. As aulas serão seguidas de atividades práticas em laboratório. Os trabalhos serão apresentados durante o transcorrer da disciplina.

Sistema de avaliação

1. Do desempenho do aluno:
A qualidade do desempenho do aluno será avaliada com base no desenvolvimento das seguintes atividades e com os seguintes critérios:
 - a) Trabalhos práticos (30%+20%=50 %)
 - b) Exercícios (15 %)
 - c) Trabalho Final (35 %)

Do desempenho da disciplina e do professor:

Os estudantes terão, igualmente, a oportunidade de fazer uma avaliação mais completa do desempenho do professor e da disciplina através do sistema de avaliação.

Plano de ensino

Bibliografia básica

1. BEZERRA, E. Princípios de Análise e Projeto de Sistemas com UML. Editora Campus. 2ª. Ed. 2007.
BOOCH, G.; RUMBAUGH, J. e JACOBSON, I., UML - Guia do Usuário. Rio de Janeiro, Campus, 2000.
MCMENAMIN, S., PALMER, J., Análise Essencial de Sistemas. São Paulo. McGraw-Hill, 1991.

Bibliografia complementar

1. PAGE-JONES, M. Projeto Estruturado de Sistemas. São Paulo. McGraw-Hill, 1998.
PRESSMAN, R. Engenharia de Software. 5ª. Edição. São Paulo. McGraw-Hill, 2003.
RUMBAUGH, J. ET. All. UML - Guia do Usuário. Rio de Janeiro. Campus, 2001.

Plano de ensino

Curso: CCI-BAC - Bacharelado em Ciência da Computação

Turma: CCI122-04U - CIENCIA DA COMPUTAÇÃO FASE 04U

Disciplina: BAN1001 - BANCO DE DADOS I

Período letivo: 2017/1

Carga horária: 72

Professor: 3809552 - REBECA SCHROEDER FREITAS

Ementa

1. Conceitos básicos; modelos de dados; aspectos de modelagem de dados; projeto e aplicações de Banco de Dados.

Objetivo geral

1. Habilitar o aluno a identificar os principais problemas relacionados ao projeto de banco de dados relacionais, bem como relacioná-los ao conjunto de estratégias de modelagem conceitual, lógica e física estabelecidas como boas práticas para alavancar o desempenho de bancos de dados relacionais.

Objetivo específico

1. - Compreender os conceitos relacionados a bancos de dados e sistemas gerenciadores de bancos de dados;
- Compreender e aplicar técnicas de modelagem conceitual e o correto uso de seus construtores;
- Compreender e identificar os principais construtores e restrições de integridade aplicadas ao modelo lógico relacional;
- Identificar a aplicação de diferentes regras de mapeamento de modelos conceituais para o modelo lógico relacional;
- Compreender e estruturar corretamente sentenças eficientes de manipulação de dados relacionais através da álgebra relacional, cálculo relacional e da linguagem SQL;
- Compreender e aplicar as principais técnicas de modelagem física para bancos de dados relacionais.

Conteúdo programático

1. Introdução a Banco de Dados (BD) e a Sistemas Gerenciadores de Banco de Dados (SGBD)
 - Modelos de Dados
 - Arquitetura de SGBDs
 - Componentes de uma SGBD
 - Interfaces
2. Projeto Conceitual de Banco de Dados
 - Modelo Entidade-Relacionamento Estendido
 - Entidades
 - Atributos
 - Relacionamentos
 - Generalização
3. Projeto Lógico de Banco de Dados Relacional
 - Conversão de Entidades, Atributos, Relacionamentos e Generalização
 - Cardinalidades e Multiplicidade em relacionamentos
 - Restrições de Integridade
4. Normalização de Dados
 - Formas normais
 - Produção de esquemas livres de redundância
 - Dependências Funcionais
 - Projetos de engenharia reversa
5. Álgebra Relacional
 - Seleção
 - Projeção
 - Produto Cartesiano
 - Atribuição
 - Renomeação
 - Otimização Algébrica
 - União
 - Diferença
 - Intersecção
 - Junções
 - Divisão
 - Atualizações
6. Cálculo Relacional
 - Cálculo de Tuplas

Plano de ensino

- Quantificador Universal - Quantificador Existencial
7. Linguagem de Consulta Comercial: SQL - DDL (Data Definition Language) - DML (Data Manipulation Language)
8. Projeto de Bancos de Dados não-relacionais - Conceitos - Modelos - Mapeamento das técnicas relacionais para as não-relacionais

Metodologia

1. A disciplina será ministrada através da exposição dos conteúdos pelo professor, da promoção de exercícios e de um trabalho final que visam a fixação do conteúdo pelos alunos, sendo que até 20% do conteúdo programático poderá ser ministrado na forma de ensino a distância através da ferramenta moodle. A prática dos conceitos apresentados será realizada através do uso de sistemas gerenciadores de banco de dados e de sua correta manipulação.
--

Sistema de avaliação

1. Do desempenho dos alunos: O desempenho dos alunos será avaliado com base no desenvolvimento das seguintes atividades: 1) avaliações individuais: - Prova 1: 25% - Prova 2: 30% - Prova 3: 25% 2) avaliações em grupo: - Trabalho 1: 10% - Trabalho 2: 10% Do desempenho do professor e da disciplina: O desempenho do professor e da disciplina será avaliado pela avaliação promovida pela própria instituição ao término do semestre. Adicionalmente, os alunos terão a liberdade de se expressar e sugerir mudanças durante todo o semestre acerca da disciplina, de seu formato e da condução da mesma pelo professor.

Bibliografia básica

1. CHEN, P. Gerenciamento de Banco de Dados. São Paulo: McGraw-Hill, 1990. DATE, C. J. Introdução a Sistemas de Banco de Dados. 7ª. Edição. São Paulo: Campus, 2000. ELMASRI, R., NAVATHE, S. B., Sistemas de Banco de Dados - Fundamentos e Aplicações. 3ª. Edição. Rio de Janeiro: LTC, 2000.

Bibliografia complementar

1. HEUSER, C. A. Projeto de Banco de Dados, 2001. SILBERSCHATZ, A; KORTH, H. F.; SUDARSHAN, S. Sistema de Banco de Dados, 2005.
--

Plano de ensino

Curso: CCI-BAC - Bacharelado em Ciência da Computação

Turma: CCI122-04U - CIENCIA DA COMPUTAÇÃO FASE 04U

Disciplina: COM0002 - COMPILADORES

Período letivo: 2017/1

Carga horária: 72

Professor: 3949795 - CRISTIANO DAMIANI VASCONCELLOS

Ementa

1. Análises léxica, sintática e semântica; Ferramentas para construção de compiladores; Geração e otimização de código intermediário; Ambientes em tempo de execução.

Objetivo geral

1. Conhecer as fases de compilação de programas, representações intermediárias de código e o ambiente em tempo de execução.

Objetivo específico

1. Entender como é o processo de geração de código, o que é executado nas fases de análise e síntese do processo de compilação. Entender o funcionamento de compiladores.

Conteúdo programático

1. Etapas da geração de um arquivo executável:
Pré-processamento, compilação, montagem e ligação.
2. Revisão sobre teoria de linguagens:
Classificação de linguagens e seus reconhecedores; linguagens regulares; linguagens regulares; Backus-Naur Form.
3. Análise léxica:
 - Especificação de tokens, utilização de gramáticas regulares.
 - Autômatos finitos deterministas.
 - Autômatos finitos não-deterministas.
 - Projeto de um analisador léxico (scanner).
4. Análise sintática top-down:
Método de descendente recursivo. Eliminação da recursividade à esquerda. Fatoração à esquerda.
5. Análise sintática top-down:
Método LL(1).
6. Análise sintática bottom-up:
Métodos: SLR(1), LR(1) e LALR(1).
7. Geradores de Analisadores Sintáticos.
8. Análise semântica.
9. Ambiente em tempo de execução.
10. Geração e otimização de código intermediário.

Metodologia

1. A disciplina será ministrada através de aula expositivas, exercícios e desenvolvimento de um compilador para linguagem de programação simplificada, fazendo um paralelo com linguagens reais, através de discussões em sala. Até 20% do conteúdo programático poderá ser ministrado na forma de ensino a distância.

Sistema de avaliação

1. O desempenho do aluno será avaliado com base no desenvolvimento das seguintes atividades e com os seguintes critérios:
 - a) avaliações individuais (provas).
 - b) exercícios.
 - c) Trabalho de conclusão da disciplina.
 - d) Prova Prática.

Plano de ensino

Média = $(40 \times \text{Prova} + 25 \times \text{Trabalho} + 25 \times \text{Prova Prática} + 10 \times \text{Exercícios}) / 100$

Bibliografia básica

1. Aho, Alfred V.; Lam, Monica S.; Sethi, Ravi; Ullman, Jeffrey D. Compiladores: Princípios, Técnicas e Ferramentas. Pearson. Cooper, Keith D.; Torczon, Linda. Construindo compiladores. Rio de Janeiro: Elsevier, 2014.

Bibliografia complementar

1. Grune, Dick; Projeto Moderno de Compiladores: Implementação e Aplicações; Campus, 2001.
Bryant, Randal E.; O'Hallaron, David R. Computer Systems: A Programmer's Perspective. Prentice Hall.
The JavaTM Virtual Machine Specification. Tim Lindholm, Frank Yellin.
flex: The Fast Lexical Analyzer. <http://flex.sourceforge.net/>
The Yacc-compatible Parser Generator. <http://www.gnu.org/software/bison/manual/bison.pdf>

Plano de ensino

Curso: CCI-BAC - Bacharelado em Ciência da Computação

Turma: CCI122-04U - CIENCIA DA COMPUTAÇÃO FASE 04U

Disciplina: PRA0001 - PROJETO DE ARQUIVOS

Período letivo: 2017/1

Carga horária: 72

Professor: 1033140358 - GLAUCO VINICIUS SCHEFFEL

Ementa

1. Dispositivos de armazenamento. Organizações básicas de arquivos. Gerenciamento de espaço. Métodos de indexação. Árvores balanceadas. Espalhamento. Tópicos especiais.

Objetivo geral

1. Ao final desta disciplina o discente deverá ser capaz de planejar, modelar, construir, gerenciar, testar e manter códigos eficientes para armazenamento e busca de dados em dispositivos de armazenamento permanente de dados.

Objetivo específico

1. O egresso será capaz de:
 1. Usar algoritmos de ordenação de forma apropriada dependendo da mídia de armazenamento;
 2. Criar estruturas indexadas para buscas eficientes
 3. Realizar buscas em Strings;
 4. Testar códigos desenvolvidos;
 5. Trabalhar com árvores balanceadas;
 6. Criar algoritmos de espalhamento;

Conteúdo programático

1. Dispositivos de armazenamento. Organizações básicas de arquivos. Gerenciamento de espaço. Métodos de indexação. Árvores balanceadas. Espalhamento. Tópicos especiais.

Metodologia

1. Aulas Expositivas, Debate, Leitura, Dinâmicas de grupo, Trabalho em equipe e estudos de caso.

Sistema de avaliação

1. Duas provas e dois trabalhos sendo:
 1. Trabalho de abertura e fixação de conhecimentos previamente adquiridos (20% da nota final)
 2. Prova (20% da nota final)
 3. Trabalho Complementar a prova. (20% da nota final)
 4. Prova (40% da nota final)

Bibliografia básica

1. DOS SANTOS, C.S.; AZEREDO, P.A. Tabelas: Organizações e Pesquisa. UFRGS, 2001. HOROWITZ, E., SAHNI, S. Fundamentos de Estruturas de Dados. Editora Campus, 1987. ISBN 8570014228.
ZIVIANI, N. Projeto de Algoritmos com Implementações em Pascal e C. Editora Thomson Learning, 2004. ISBN 8522103909.

Bibliografia complementar

1. WIRTH, Niklaus.; LEE, Cheng Mei. Algoritmos e estruturas de dados. Rio de Janeiro: Livros Técnicos e Científicos, c1999. 255 p. ISBN 8521611900 (broch.)
VELOSO, Paulo A. S; SANTOS, Clesio Saraiva dos; AZEREDO, Paulo; FURTADO, Antonio Luz. Estruturas de dados. 1. ed. Rio de Janeiro: Campus, c1984. 228 p. : ISBN 8570013523 (broch.)

Plano de ensino

Curso: CCI-BAC - Bacharelado em Ciência da Computação

Turma: CCI122-04U - CIENCIA DA COMPUTAÇÃO FASE 04U

Disciplina: REC0001 - REDES DE COMPUTADORES

Período letivo: 2017/1

Carga horária: 72

Professor: 3375552 - CHARLES CHRISTIAN MIERS

Ementa

1. Introdução às redes de comunicações; Modelo de referência OSI; Camada Física (técnicas de transmissão analógica e digital); Técnicas de multiplexação; Camada de enlace de dados; Camada de Rede; Camada de transporte; Modelo TCP/IP (Camada de Aplicação); Redes locais e metropolitanas; Projeto de redes.

Objetivo geral

1. Compreender os conceitos básicos de redes de computadores.

Objetivo específico

1. -Compreender a definição e a utilização de redes de computadores;
-Compreender os modelos e arquiteturas existentes;
-Compreender as funcionalidades e serviços das camadas do modelo de referência OSI;

Conteúdo programático

1. 1. Introdução às redes de computadores
 - a. Conceitos básicos: comutação de dados, sistema de comunicação, meios de transmissão, protocolos, tempo de propagação e de transmissão, atrasos, tipos de conexão, vazão
 - b. Apresentação da evolução das arquiteturas e sistemas: camadas e protocolos, entidade, serviços, interfaces, endereçamento.
 - c. Funções de um protocolo: controle de erros, controle de fluxo, segmentação e remontagem, endereçamento, multiplexação, encapsulamento
 - d. Apresentação das topologias de rede
 - i. Redes geograficamente distribuídas
 - ii. Redes locais e metropolitanas
 - e. Meios de transmissão
 - f. Comutação de circuitos e de pacotes
 - g. Arquiteturas e padrões
 - i. Modelo de referência OSI da ISO
 - ii. Noções Arquitetura TCP/IP
 - i. Comparação entre modelo OSI e arquitetura TCP/IP
2. 2. Camada de aplicação
 - a. Princípios e serviços
 - b. Protocolo HTTP
 - c. Aplicações: TELNET, DNS, FTP, NFS, correio eletrônico
 - d. Arquitetura Peer-to-Peer e Cliente-Servidor
3. 3. Camada de transporte
 - a. Princípios e serviços
 - b. Algoritmos Go-Back-N e Selective Repeat
 - c. Comunicação confiável
 - d. Estabelecimento e encerramento de conexões
 - e. Protocolo UDP
 - f. Protocolo TCP
 - g. Princípios de Controle de Congestionamento
4. 4. Camada de rede
 - a. Introdução
 - b. Comutação
 - c. Endereçamento
 - d. Roteamento
 - e. Protocolo IP
 - f. Algoritmos de roteamento
 - g. Roteamento na Internet
5. 5. Camada de enlace e redes locais
 - a. Introdução
 - b. Detecção e correção de erros de transmissão
 - c. Tipos de Serviços
 - d. Protocolos de Acesso Múltiplos

Plano de ensino

e. Endereçamento f. Ethernet g. Equipamentos de redes: hubs e switches h. Redes Virtuais i. Protocolo ARP j. Protocolo PPP k. Noções de redes sem fio
6. AP - Avaliação Progressiva
7. TE1 - Explicação, orientação e execução TE1
8. EAD - Atividade extra-sala remota
9. Semana da computação
10. Segunda chamada AP
11. Apresentação TE1

Metodologia

1. O método empregado será de aulas expositivas mesclado com trabalhos, que devem ser apresentados em sala de forma a dar o conhecimento necessário ao aluno e induzir o mesmo a pesquisar/aprofundar nos assuntos através de trabalhos. As aulas poderão ser ministradas em modo presencial ou semi-presencial (até 20% da carga total no modo semi-presencial).

Sistema de avaliação

1. Provas escritas (Avaliações Progressivas, AP1 e AP2); Trabalhos em grupo (TE1); Listas de exercícios (LE1+LE2); Participação efetiva nas aulas (presença, pontualidade, atenção e principalmente contribuição significativa nos estudos realizados). $MS = (TE1*6+AP1*2+AP2*2)/10$ Os estudantes terão, igualmente, a oportunidade de efetuar, ao andamento da disciplina uma avaliação mais completa do desempenho do professor e da disciplina. O formulário para esta atividade será preparado para a ocasião ou fornecido pelo coordenador do curso.
--

Bibliografia básica

1. KUROSE, J. & ROSS, K. Computer Networking: A Top-Down Approach Featuring the Internet. Addison-Wesley, 2010, terceira edição. TANENBAUM, Andrew. Redes de Computadores. 4a. Edição. Editora Campus, Ltda. 2011. Quinta edição. PETERSON, Larry L.; DAVIE, Bruce S. Redes de Computadores: Uma Abordagem de Sistemas, 3a Ed. Rio de Janeiro: Campus, 2004. FOROUZAN, Behrouz, MOSHARRAF, Firouz, Redes de Computadores. Uma abordagem Top-Down, McGraw Hill, 2014
--

Bibliografia complementar

1. STALLINGS, William. Data and Computer Communications, 8th Ed. New Jersey: Pearson, 2007 SOARES, L.F.G. et al. Redes de Computadores - Das LANs, MANs e WANs às redes ATM. 2a Edição Editora Campus. 1995. 693 p. Request for Comments (RFCs), Disponível em http://www.ietf.org/rfc.html McKeown, N., Anderson, T., Balakrishnan, H., Parulkar, G., Peterson, L., Rexford, J., Shenker, S., and Turner, J. (2008). OpenFlow: enabling innovation in campus networks. SIGCOMM Comput. Commun. Rev., 38(2): 69-74. Handley, M. (2006). Why the Internet only just works. BT Technology Journal, 24:119- 129.
--

Plano de ensino

Curso: CCI-BAC - Bacharelado em Ciência da Computação

Turma: CCI122-04U - CIENCIA DA COMPUTAÇÃO FASE 04U

Disciplina: SOFT001 - ENGENHARIA DE SOFTWARE

Período letivo: 2017/1

Carga horária: 72

Professor: 3809552 - REBECA SCHROEDER FREITAS

Ementa

1. Processos de Software; Modelos, métricas, estimativas e alocação de recursos; Processo individual de software (PSP- Personal Software Process); Qualidade e sua administração; Alocação e administração de Pessoal e recursos; Ambientes de uso de software; Ferramentas de desenvolvimento de software.

Objetivo geral

1. Habilitar o aluno a identificar os principais problemas da produção de software, bem como relacioná-los ao conjunto de procedimentos, métodos e ferramentas estabelecidos pela Engenharia de Software para promover a melhoria contínua do produto e do processo de software.

Objetivo específico

1. - Compreender os conceitos relacionados à Engenharia de Software;
- Compreender e aplicar técnicas, métodos, boas práticas e ferramentas para a produção de software;
- Identificar a aplicação de diferentes modelos de processo de software;
- Compreender e aplicar métodos para o levantamento de estimativas aplicadas a projetos de software;
- Compreender e avaliar processos de verificação e validação de software, bem como modelos de qualidade para melhoria contínua do software e de seu processo.

Conteúdo programático

1. Introdução à Engenharia de Software e Modelos de Processo de Software;
 - Modelos Prescritivos
 - Modelos Ágeis
 - Processo Unificado
2. Modelos de Processo Contemporâneos
 - Modelos Ágeis - XP e Scrum
3. Gerência de Projetos
 - Planejamento
 - Declaração de Escopo
 - Estimativas de Esforço
 - Gerenciamento de Riscos
4. Gerenciamento e Configuração de Software
 - Itens de Configuração de Software
 - Rastreabilidade
 - Baseline e Release
 - Controle de Versão
 - Repositório
 - Políticas de Compartilhamento de Itens
 - Auditoria de Configuração
 - Ferramentas para Controle de Versão
5. Ferramentas Case
 - Características e Funcionalidades
 - Ferramentas específicas
 - Geração de Código
6. Verificação e Validação de Software
 - Erro, Defeito e Falha
 - Teste de Funcionalidade
 - Teste de Unidade
 - Teste de Integração
 - Teste de Sistema
 - Teste de Aceitação
 - Teste de Ciclo de Negócio
 - Teste de Regressão
 - Testes Suplementares

Plano de ensino

- Teste de Interface com Usuário
- Teste de Performance (Carga, Estresse e Resistência)
- Teste de Segurança
- Teste de Recuperação de Falha
- Teste de Instalação
- Teste Estrutural
- Complexidade Ciclomática
- Grafo de Fluxo
- Caminhos Independentes
- Casos de Teste
- Múltiplas Condições
- Caminhos Impossíveis
- Particionamento de Equivalência
- TDD - Desenvolvimento Orientado a Testes

7. Qualidade de Software
- Qualidade de Produto e suas métricas
 - Trabalho 4: análise comparativa entre - Modelo de Qualidade SquaRE - ISO/IEC 25010:2011
 - Modelo de Qualidade de Dromey
 - Gestão da Qualidade
 - Qualidade de Processo
 - ISO/IEC 90003
 - ISO/IEC 15504 - SPICE
 - CMMI
 - MPS-BR
 - Melhoria de Processo de Software (SEI-IDEAL)
 - Linha de Processo de Software

Metodologia

1. A disciplina será ministrada através da exposição dos conteúdos pelo professor e da promoção de exercícios e trabalhos que visam a fixação do conteúdo pelos alunos, sendo que até 20% do conteúdo programático poderá ser ministrado na forma de ensino a distância através da ferramenta moodle. A prática dos conceitos apresentados será realizada através do uso de ferramentas oportunas e do uso de jogos de simulação para ambientes de desenvolvimento de software.

Sistema de avaliação

1. Do desempenho dos alunos:
O desempenho dos alunos será avaliado com base no desenvolvimento das seguintes atividades:
- 1) avaliação individual:
- Prova: 40%
- 2) Trabalhos em grupo:
- Entrega da primeira parte do Projeto da Disciplina: 20%
 - Entrega da segunda parte do Projeto da Disciplina: 40%
- Do desempenho do professor e da disciplina:
O desempenho do professor e da disciplina será avaliado pela avaliação promovida pela própria instituição ao término do semestre. Adicionalmente, os alunos terão a liberdade de se expressar e sugerir mudanças durante todo o semestre acerca da disciplina, de seu formato e da condução da mesma pelo professor.

Bibliografia básica

1. BEZERRA, E.. Princípios de Análise e Projeto de Sistemas com UML. 2ª. Ed. Rio de Janeiro: Campus. 2007.
- BOOCK, G., RUMBAUGH, J., JACOBSON, I., UML: guia do usuário. Rio de Janeiro. Campus, 2000.
- SOMMERVILLE, I., Engenharia de Software. 8ª. Ed. São Paulo: Pearson, 2007.
- LARMAN, C., Utilizando UML e padrões: uma introdução a análise e ao projeto orientados a objetos. 2ª. Ed. Porto Alegre: Bookmann, 2002.
- ROCHA, a. R. C. da.; MALDONADO, J. C.;WEBER, K. C. Qualidade de Software: Teoria e Prática. São Paulo: Prentice Hall, 2001.

Bibliografia complementar

1. PRESSMAN, R. S. Engenharia de Software. 6ª. Ed. São Paulo: McGraw-Hill, 2006.
- MEDEIROS, E. Desenvolvendo Software com UML 2.0: definitiva. São Paulo: Makron Books, 2009.
- BOURQUE, P. e DUPUIS, R. (Eds). Guide to IEEE Software Engineering Body of Knowledge (SWEBOOK). 2004 Version. [S.l.]: IEEE Computer Society, 2004. Disponível em: <http://www.computer.org/portal/web/swebok/htmlformat> . Acesso em: 14 fev. 2011.
- WEINBERG, Gerald M. Software com qualidade: Pensando e idealizando sistemas. São Paulo: Makron Books. 1993.
- CHRISSIS, A. B.; KONRAD, M.; SHRUM, S. CMMI: Guidelines for Process Integration and Product

Plano de ensino

Improvement. 2nd Ed. New Jersey: Addison-Wesley. 2009.

SEI. Software Engineering Institute. CMMI for Development (CMMI-DEV), Version 1-2, Technical report CMU/SEI-2006-TR-008. Pittsburgh, PA: Software Engineering Institute, Carnegie Mellon University, 2006.

Disponível em: <http://www.sei.cmu.edu/library/abstracts/reports/06tr008.cfm> Acesso em 14 fev. 2011.

SOFTEX. MPS.BR - Melhoria de Processo do Software Brasileiro. Guia Geral: 2009. Disponível em

HTTP://www.softex.br/mpsbr/_guias/guias/MPS.BR_Guia_Geral_2009.pdf Acesso em: 14 fev. 2011

Plano de ensino

Curso: CCI-BAC - Bacharelado em Ciência da Computação

Turma: CCI122-04U - CIENCIA DA COMPUTAÇÃO FASE 04U

Disciplina: SOP0001 - SISTEMAS OPERACIONAIS

Período letivo: 2017/1

Carga horária: 72

Professor: 3877850 - RAFAEL RODRIGUES OBELHEIRO

Ementa

1. Introdução. Conceitos de processos e memória. Gerência de processo/processador. Comunicação entre processos. Alocação de recursos. Gerenciamento de memória: memória virtual, paginação, segmentação e swap. Sistemas de arquivos. Gerenciamento de dispositivos de entrada e saída.

Objetivo geral

1. Capacitar o aluno a compreender os princípios de projeto e implementação de Sistemas Operacionais.

Objetivo específico

1. - Conceituar sistemas operacionais;
- Compreender os princípios de multiprogramação;
- Compreender o gerenciamento de processos de um SO;
- Compreender os princípios de programação concorrente;
- Compreender o gerenciamento de memória de um SO;
- Compreender o gerenciamento de E/S de um SO;
- Compreender o gerenciamento de arquivos de um SO;
- Introduzir os princípios de projeto de um SO.

Conteúdo programático

1. Introdução a SO
Apresentação do plano de ensino
2. Introdução
Fundamentos de SO
Histórico de SO
Conceitos de SO
Organização de SO
3. Fundamentos de SO
4. Histórico de SO
5. Tipos de SO
6. Conceitos básicos de SO
Visão geral das funcionalidades de um SO
Noções de gerência de processos
Noções de gerência de memória
Noções de gerência de E/S
Noções de deadlocks
Noções de sistemas de arquivos
7. Princípios de hardware
Revisão de conceitos básicos de hardware do ponto de vista de um SO
8. Organização de SO
Organização interna de SO
Arquiteturas monolíticas, em camadas, máquinas virtuais, cliente-servidor, etc.
9. Gerência de processos
Conceitos de processos e threads
Implementação de processos e threads
Comunicação interprocessos
Escalonamento de processos
10. Processos
Conceito de processo
Criação e encerramento de processos
Diagrama de estados de processos
Blocos de controle de processo

Plano de ensino

11. Threads Conceito de thread Uso de threads Implementação de threads de usuário e de núcleo
12. Programação com threads
13. Comunicação interprocessos: condições de disputa, regiões críticas, soluções com espera ocupada
14. Comunicação interprocessos: sleep e wakeup
15. Comunicação interprocessos: semáforos e mutexes
16. Comunicação interprocessos: monitores
17. Comunicação interprocessos: passagem de mensagens
18. Comunicação interprocessos no Linux IPC usando threads: mutexes e variáveis de condição IPC usando processos: memória compartilhada e semáforos POSIX
19. Escalonamento de processos Escalonamento em lote: FCFS, SJF, SRTN Escalonamento interativo: round-robin, prioridades, filas múltiplas, fração justa
20. Escalonamento de processos no Linux
21. Deadlocks Conceitos de deadlocks Modelagem de deadlocks Tratamento de deadlocks
22. Gerência de entrada e saída Organização do software de E/S Escalonamento de disco
23. Princípios de hardware de E/S
24. Princípios de software de E/S Camadas de software de E/S
25. Discos magnéticos Princípios de funcionamento Tempos de acesso a disco Algoritmos de escalonamento de disco
26. Gerência de E/S no Linux Princípios de gerência de E/S no Linux Escalonamento de disco no Linux
27. Gerência de memória Gerência de memória contígua Paginação Segmentação
28. Gerência de memória contígua Gerência de memória com partições fixas Gerência de memória com partições variáveis Swapping
29. Memória virtual Paginação Algoritmos de substituição de páginas Segmentação
30. Gerência de memória no Linux
31. Sistemas de arquivos Arquivos Diretórios Implementação de sistemas de arquivos
32. Sistemas de arquivos no Linux
33. Exercícios de revisão

Plano de ensino

34. Prova
35. Trabalho Trabalho prático de implementação
36. Devolução e correção da prova

Metodologia

1. Aula expositiva dialogada. Desenvolvimento de atividades individuais e em grupo. Atividades práticas (implementações) em laboratório. Até 14 horas-aula (menos que 20% da carga horária da disciplina) poderão ser desenvolvidas na modalidade à distância.

Sistema de avaliação

1. Três provas (P1, P2, P3). Ocasionalmente, exercícios propostos em aula poderão ser avaliados e compor a nota da prova que cobre o assunto correspondente.

Média final = $0,3 \cdot P1 + 0,35 \cdot P2 + 0,35 \cdot P3$

Bibliografia básica

1. TANENBAUM, Andrew S. Sistemas Operacionais Modernos, 3ª Ed. São Paulo: Pearson, 2010.

OLIVEIRA, Rômulo S.; CARISSIMI, Alexandre da S.; TOSCANI, Simão S.; Sistemas Operacionais, 2ª Ed. Porto Alegre: Instituto de Informática da UFRGS: Sagra Luzzatto, 2001.

Bibliografia complementar

1. MACHADO, Francis Berenger; MAIA, Luiz Paulo. Arquitetura de Sistemas Operacionais, 3ª Ed. Rio de Janeiro: LTC, 2002.

SILBERSCHATZ, Avi; GALVIN, Peter; GAGNÉ, Greg. Fundamentos de Sistemas Operacionais, 8ª Ed. Rio de Janeiro: LTC, 2010.

STALLINGS, William. Operating Systems: internals and design principles, 6th Ed. Prentice-Hall, 2009.

STUART, Brian L. Princípios de Sistemas Operacionais. Rio de Janeiro: Cengage, 2011.

TANENBAUM, Andrew S. Organização Estruturada de Computadores, 5ª Ed. São Paulo: Pearson, 2007.

TANENBAUM, Andrew S. Sistemas Operacionais Modernos, 2ª Ed. Rio de Janeiro: Pearson Brasil, 2003.

TANENBAUM, Andrew S.; WOODHULL, Albert S. Sistemas Operacionais: Projeto e Implementação, 2ª Ed. Porto Alegre: Bookman, 2000.

Plano de ensino

Curso: CCI-BAC - Bacharelado em Ciência da Computação

Turma: CCI122-04U - CIENCIA DA COMPUTAÇÃO FASE 04U

Disciplina: TEG0001 - TEORIA DOS GRAFOS

Período letivo: 2017/1

Carga horária: 72

Professor: 2646943 - OMIR CORREIA ALVES JUNIOR

Ementa

1. Noções básicas de grafos. Representação de grafos, grafos infinitos. Isomorfismo de grafos. Distâncias. Coloração. Grafos acíclicos e expansão de grafos em árvores. Planaridade. Problemas do caminho mínimo. Problemas Eulerianos e Hamiltonianos. Fluxo em redes. Algoritmos de Busca em Grafos. Introdução ao estudo de estruturas combinatórias.

Objetivo geral

1. Apresentar os principais conceitos da área de grafos aos acadêmicos e aplicar os conhecimentos no desenvolvimento, implementação e análise dos algoritmos aplicados na Teoria de Grafos.

Objetivo específico

1. Ao final da disciplina os estudantes vão ter noções gerais para serem capazes de:
 - Apresentar os conceitos básicos de Grafos.
 - Apresentar os aspectos de planaridade, isomorfismo e coloração.
 - Apresentar os conceitos relacionados a coloração e suas aplicações.
 - Apresentar as estruturas de dados de como um grafo possa ser representado computacionalmente
 - propor exercícios práticos de implementação.
 - Apresentar os conceitos de Árvores.
 - Apresentar diversos algoritmos, suas aplicações e propor exercícios de implementação aos alunos.

Conteúdo programático

1. aula_01 : Introdução ao curso de Teoria dos grafos
Apresentação do plano de ensino
Objetivo do Curso
procedimentos de avaliação
referencias bibliográficas
Introdução a TEG
Definição de Grafos e Dígrafos
Graus de Entrada e de saída, laços e arcos. Nós e arcos adjacentes e independentes
Grafos regulares
Teorema do Aperto de mão
exercícios
2. Aula_02: Conceitos básicos de Teoria dos Grafos
Grafos Completos, valorados, rotulados e acíclicos
Operações de arcos e nós:
Inclusão e exclusão de um nó
Inclusão e exclusão de um arco
Fusão e explosão de nós
exercícios
3. Aula_03 : Conceitos básicos de Teoria dos Grafos
Definição de subgrafos
subgrafos induzidos por nós e arcos
Percurso elementar, simples e ciclos
Grafo conectado
grau de entrada e de saída de um nó
Componente de um grafo
Cortes de nós e arestas
Conectividade de um grafo
comprimento
exercícios
4. Aula_4 : Conceitos básicos de teoria dos grafos
Maximal e Máximo
Cliques: Clique máximo e maximal
Grafos K-Partidos
Grafo bipartido completo

Plano de ensino

Grau de um nó de um dígrafo União de grafos excentricidade de um grafo Raio, diâmetro e mediana de um grafo nós periféricos de um grafo resolução de exercícios
5. Aula_05: Representação Computacional de Grafos Representação Computacional de grafos Matrizes adjacentes Trabalho prático em equipe na aula: implementação de grafos utilizando matrizes
6. aula_06: Representação Computacional de grafos representação computacional de grafos Estrutura de Listas Trabalho prático em grupo em sala: Implementar grafos utilizando Listas
7. Aula 07 - Operações de arcos e nós Inclusão e exclusão de um arco Inclusão e exclusão de um nó Fusão e explosão de um nó excentricidade de um nó raio e diâmetro de um grafo nó periférico de um grafo centro de um grafo mediana de um grafo exemplos exercícios
8. 08 - Isomorfismo e planaridade Definição de grafos isomorfos propriedades de grafos isomorfos exemplos Definição de grafos planares fórmula de Euler grafos homeomorfos teorema de Kuratowski exemplos exercícios
9. Aula 9 - Grafos Eulerianos e Hamiltonianos Definição de Grafos eulerianos e semi-eulerianos exemplos Complemento de um Grafo União de Grafos Componentes de um Grafo Algoritmo de Fleury Exercícios Definição de Grafos Hamiltonianos exemplos Trabalho prático: Implementar o algoritmo de Fleury
10. Aula 10 : Coloração de grafos definição de coloração número cromático Teorema das quatro cores exemplos grafo de Petersen grafos com sinais resolução de exercícios
11. Aula_11: Primeira avaliação TEG0001: primeira avaliação escrita. Duração 01:40 minutos
12. Aula_12: Entrega primeira prova e coloração Entrega e correção da primeira avaliação
13. Aula 13: Árvores Introdução arvores enraizadas árvores disjuntas grau de uma árvore vértices internos árvore cheia

Plano de ensino

nível de um vértice árvore enraizada balanceada centro de uma árvore exercícios Árvore geradora árvore binária representação computacional de uma árvore (listas e matrizes) Endereçamento Global de uma árvore Algoritmo de percurso: Pré-ordem resolução de exercícios implementação algoritmo Pré-ordem
14. Aula 14: Árvores (continuação) Algoritmos de percurso: Simétrico Pós-ordem resolução de exercícios Implementação do algoritmo Simétrico
15. Aula_15: Árvores (continuação) Árvores de decisão Ordenação de elementos busca sequencial decisão binária árvores de jogos códigos de prefixo resolução de exercícios
16. Aula_16: Busca em Profundidade e em nível Busca em Profundidade em Grafos Árvores geradoras com busca em profundidade exemplos Busca em Nível em Grafos Árvores geradoras com busca em nível exemplos resolução de exercícios
17. Aula_17 - Busca em Nível em grafos (conitnuação) códigos de Huffman exemplo das n rainhas Busca em nível em grafos direcionados resolução de exercícios
18. Aula_18: Caminho mínimo de um grafo G Introdução arestas ponderadas Comprimento ou peso de um caminho Método de relaxação exemplos Caminho mínimo entre um nó U e todos os demais nos de G Algoritmo de DIJSKTRA exemplo resolução de exercícios
19. Aula_19: Segunda avaliação de TEG0001 Segunda avaliação de TEG0001 avaliação escrita duração 01h:40 min
20. Aula 20 - Caminhos mínimos (conitnuação) Caminho mínimo entre nó U e todos os demais nos V de G introdução Algoritmo de Bellmann-Ford exemplos resolução de exercícios entrega e correção da segunda avaliação
21. Aula_21: Caminho mínimo entre todos os nós Algoritmo de Floyd-Wharshall Fecho Transitivo Direto e Indireto exemplos resolução de exercícios
22. Aula_22 - Árvore geradora mínima Algoritmo de PRIM

Plano de ensino

Algoritmo de Kruskal exemplos exercícios Fecho transitivo direto e indireto de um grafo exercícios
23. Aula_23: Conectividade de Grafos conectividade de arestas e vértices conjunto de corte vértices e arestas disjuntos Teorema de Menger exemplos Coloração de arestas Índice cromático Teorema de Vizing Teorema de Shannon exemplos exercícios
24. Aula_24: Fluxo máximo em grafos(redes) rede em fluxo fluxo e restrição de capacidade conservação de fluxo cortes Caminho aumentado (augmenting path) redes residuais corte mínimo , fluxo máximo capacidade residual exemplos exercícios
25. Aula_25: Método de Ford-Fulkerson determinação do fluxo máximo utilizando o método de Ford-Fulkerson exemplos exercícios
26. Aula_26: Método de Dinitz determinação do fluxo máximo utilizando o algoritmo de Dinitz exemplos exercícios
27. Aula_27: Algoritmo de Edmonds-Karps exemplos Fluxo máximo com custo mínimo algoritmo de Roy, Busaker e Gowen Problema do Caixeiro Viajante Problema da Mochila exemplos exercícios
28. Aula_28: Emparelhamento Matching (emparelhamento/casamento) emparelhamento máximo emparelhamento maximal cadeia M-aumentante Fluxo máximo e emparelhamento Método Hungaro Árvores de Steiner exemplos exercícios
29. Aula_35: Revisão geral do curso Revisão geral dos assuntos abordados no curso preparação para a terceira prova resolução de exercícios
30. Aula_36: Terceira avaliação terceira avaliação de TEG0001 duração 01:40
31. Aula 31: Correção terceira avaliação

Plano de ensino

correção da terceira avaliação resolução de exercícios preparação para o exame
32. Aula_32: Apresentação de trabalhos em equipe:
33. Aula_33: Apresentação de trabalhos em equipe
34. Aula_34: Apresentação de trabalhos em equipe:
35. Aula_35: resolução de exercícios resolução de exercícios utilizando os algoritmos: Ford-Fulkerson Custo mínimo e fluxo máximo Corte mínimo e fluxo máximo
36. Aula_36: apresentação trabalhos em equipe
37. Aula_20.1: Caminho mínimo (continuação) continuação algoritmo de Bellmann-Ford resolução de exercícios algoritmo CPM (Critical Path Method) resolução de exercícios
38. Aula_29: otimização combinatória problema do caixeiro Viajante problema da Mochila resolução de exercícios

Metodologia

1. Cada um dos temas listados no programa da disciplina será abordado visual e conceitualmente por meio da articulação crítico-reflexivo, com aulas expositivas e/ou dialogadas utilizando recursos data-show, que deverá ser sucedido da realização de demonstrações práticas e a realização de exercícios utilizando ferramentas para interação com grafos e ambientes para desenvolvimento de programas. Propiciando aos discentes: análise, interpretação e construção de novos conhecimentos com apoio das realizações de pesquisas bibliográficas.

Considerações importantes sobre a metodologia da disciplina:

-Os alunos terão até 15 minutos de tolerância após o início da aula para entrar na sala. De outra forma não será considerada a presença.

-É necessário que cada aluno providencie uma calculadora para ser usada nas provas e nas aulas. Não será permitido o uso de calculadoras de celulares ou de qualquer dispositivo com acesso a Internet ou que possibilite o envio de mensagens (sms) durante a realização das provas.
-Será utilizado o sistema Moodle2 onde serão disponibilizados arquivos utilizados na disciplina, bem como, avisos e dúvidas quanto aos assuntos da disciplina.

Link: moodle2.joinville.udesc.br

Sistema de avaliação

1. Os alunos serão avaliados através dos seguintes Instrumentos de Avaliação:

- Participação ativa nas aulas e nos trabalhos;
- Avaliações individuais, Provas (PR);
- Elaboração e apresentação de Trabalhos (TR) em grupo;

Os seguintes critérios serão observados para fins de avaliação:

- Domínio dos conteúdos discutidos, participação nas atividades, responsabilidade e pontualidade;
- Prazos de entrega de trabalhos;
- Frequência suficiente (75%).

A média final das avaliações (MF) será calculada através da seguinte fórmula:

$$MF = NP1 \cdot 0,30 + NP2 \cdot 0,35 + PR \cdot 0,20 + TR \cdot 0,15$$

onde: NP_i - Nota da prova i (i=1, 2 ou 3); PR- Projetos e TR-Trabalhos.

. TRABALHOS: O conjunto de trabalhos é composto de:

- . trabalhos em grupo de 02 alunos para implementar algoritmos diversos
- . projeto final do curso

Plano de ensino

Critérios de aprovação:

- Os alunos com MF igual ou superior a 7,0 e com 75% de frequência estão aprovados.
- Os alunos com média inferior a 7,0 estarão em Exame Final.

Os alunos com frequência suficiente poderão fazer a Prova de Exame (PEX), compreendendo toda a matéria, e a média final será calculada com a fórmula: $MF \cdot 0,6 + PEX \cdot 0,4$.

Do desempenho da disciplina e do professor:

A coordenação do curso fará a avaliação durante o semestre.

Bibliografia básica

1. LUCCHESI, C. L. et alii. Aspectos Teóricos da Computação, Parte C: Teoria dos Grafos, projeto Euclides, 1979.
- SANTOS, J. P. O. et alii. Introdução à Análise Combinatória. UNICAMP; 1995.
- SZWARCFITER, J. L. Grafos e Algoritmos Computacionais. Campus, 1986.
- GERSTING, Judith L. Fundamentos Matemáticos para a Ciência da Computação. Rio de Janeiro. 3a Ed. Editora.

Bibliografia complementar

1. 1.) CORMEN, T. Introduction to Algorithms, third edition, MIT press, 2009 (*)
- 2.) ROSEN, K. Discrete Mathematics and its applications, seventh edition, McGraw Hill, 2011. (*)
- 3.) WEST, Douglas, B. Introduction to Graph Theory, second edition, Pearson, 2001. (*)
- 4.) BONDY, J.A., MURTY, U.S.R., Graph Theory with applications, Springer, 1984 (*)
- 5.) SEDGEWICK, R. Algorithms in C - part 5 - Graph Algorithms, third edition, 2002, Addison-Wesley. (*)
- 6.) GOLDBARG, M., GOLDBARG E., Grafos: Conceitos, algoritmos e aplicações. Editora Elsevier, 2012. (*)
- 7.) BONDY, J.A., MURTY, U.S.R., Graph Theory with applications, Springer, 1984
- 8.) FEOFIOFF, P., KOHAYAKAWA, Y., WAKABAYASHI, Y., uma introdução sucinta à teoria dos grafos. 2011. (www.ime.usp.br/~pf/teoriadosgrafos)
- 9.) DIESTEL, R. Graph Theory, second edition, springer, 2000
- 10.) FURTADO, A. L. Teoria de grafos. Rio de janeiro. Editora LTC. 1973.
- 11.) WILSON, R.J. Introduction to Graph Theory. John Wiley & Sons Inc., 1985
- 12.) BOAVENTURA NETTO, P. O. Grafos: Teoria, Modelos, Algoritmos. Edgard Blucher, SP, quinta edição.

Plano de ensino

Curso: CCI-BAC - Bacharelado em Ciência da Computação

Turma: CCI122-05U - CIENCIA DA COMPUTAÇÃO FASE 05U

Disciplina: BAN2001 - BANCO DE DADOS II

Período letivo: 2017/1

Carga horária: 72

Professor: 3921492 - FABIANO BALDO

Ementa

1. Sistemas de Gerenciamento de Banco de Dados (SGBD): arquitetura e aspectos operacionais; Projeto e implementação de aplicações de Banco de Dados. Tópicos em bancos de dados e linguagens de consulta não convencionais.

Objetivo geral

1. Aprofundar os conhecimentos e habilidades do aluno no projeto e manipulação de esquemas de dados, assim como familiarizá-lo com os principais módulos que compõem os sistemas de gerenciamento de banco de dados (SGBDs).

Objetivo específico

1. a) Utilizar a linguagem de consulta estruturada (SQL) para manipular dados;
b) Conhecer os principais módulos de operação dos SGBDs;
c) Melhorar o desempenho do banco de dados;
d) Conhecer novos modelos de banco de dados e suas linguagens de consulta.

Conteúdo programático

1. Apresentação do Plano de Ensino
2. Revisão sobre Modelos de Dados e SGBDs
3. Revisão sobre Modelo Entidade-Relacionamento
4. Revisão sobre Projeto de Banco de Dados
5. Prática - Projeto ER
6. Mapeamento ER para Relacional
7. Prática - Mapeamento ER para Relacional
8. DDL - Criação, alteração e exclusão de estruturas do BD
9. Prática - Criação de Esquemas usando DDL
10. DML - Comandos Básicos de Consulta
11. Prática - Consultas Simples
12. DML - Comandos Avançados de Consulta
13. Prática - Consultas Avançadas
14. Prática - Consultas Avançadas (Exercício Complementar)
15. Transações
16. Processamento de Transações - Recuperação de BDs
17. Prática - Inserção de Dados em Múltiplas Tabelas
18. Processamento de Transações - Controle de concorrência
19. Prática - Processamento de Transações (Recuperação de Banco de Dados)
20. Prática - Processamento de Transações (Controle de Concorrência)
21. Especificação de Visões
22. Prática - Criação de Visões
23. Implementação de Funções
24. Prática - Especificação de Funções

Plano de ensino

25. Implementação de Gatilhos
26. Prática - Especificação de Gatilhos
27. Banco de Dados Objeto-Relacional
28. Prática - Criação de esquema Objeto-Relacional Criação de esquema de Banco de Dados Objeto-Relacional no PostgreSQL
29. Otimização de Consultas
30. Indexação de dados
31. Big Data e NoSQL
32. Banco de Dados Documento - MongoDB
33. Prática - Consultas no MongoDB
34. Curso de SQL On-line (a distância)
35. Prova 1
36. Prova 2
37. Apresentação Trabalho Final
38. Revisão para Prova 1
39. Revisão para Prova 2
40. Semana da Computação
41. Apresentação de Seminário
42. Banco de Dados Chave-Valor - Redis
43. Prática - Consultas no Redis
44. Banco de Dados Família de Coluna - Cassandra
45. Prática - Consultas no Cassandra
46. Banco de Dados Grafo - Neo4J
47. Prática - Consultas no Neo4J

Metodologia

1. Aulas expositivas acompanhadas de trabalhos práticos relacionados aos conteúdos apresentados na aula; Aulas práticas em laboratório que objetivam a implementação dos conceitos apresentados nas aulas teóricas; Listas de exercícios para auxiliar na fixação do conteúdo apresentado; Seminários para proporcionar a busca de forma autônoma pelo conhecimento; Provas teóricas para avaliar o conteúdo conceitual aprendido; Trabalhos para avaliar a capacidade de uso dos conceitos aprendidos; Até 20% do conteúdo programático poderá ser ministrado na forma de ensino a distância, com o auxílio da ferramenta Moodle.

Sistema de avaliação

1. Os alunos serão avaliados com base no seu desempenho nas provas, trabalhos e listas de exercícios, sendo que ao final do semestre o aluno deverá ter comparecimento mínimo de 75% às aulas e desempenho mínimo de 70% nas avaliações.
O grau de desempenho do aluno será avaliado com base no desenvolvimento das seguintes atividades e com os seguintes critérios:
 - a) Listas de exercícios individuais (10% da média);
 - b) 1 seminário em dupla (20% da média);
 - c) 1 trabalho de implementação em dupla (20% da média)
 - d) 2 provas individuais (50% da média, 25% cada).

Bibliografia básica

1. ELMASRI, R.; NAVATHE, S. B. Sistemas de banco de dados. 6 ed. São Paulo: Pearson Education, 2011.
RAMAKRISHNAN, R.; GEHRKE, J. Sistemas de gerenciamento de banco de dados. São Paulo : McGraw-Hill, 2008.
SADALAGE, P. J.; FOWLER, M. NoSQL essencial: Um guia conciso para o mundo emergente da persistência poliglota. São Paulo: Novatec, 2014.

Bibliografia complementar

Plano de ensino

1. SILBERSCHATZ, A. Sistema de banco de dados. Rio de Janeiro: Campus, 2012.
HEUSER, C. A. Projeto de Banco de Dados. 4 ed. Porto Alegre: Bookman, 2001.
DATE, C. J. Introdução a sistemas de banco de dados. 8 ed. São Paulo: Campus, 2004.
GARCIA-MOLINA, H.; ULLMAN, J. D.; WIDOM, J. Database systems: The complete book . 2. ed. Upper Saddle River, N.J.: Pearson/Prentice Hall, 2009.
NASSU, E. A.; SETZER, V. W. Bancos de dados orientados a objetos. São Paulo: E. Blucher, 1999.

Plano de ensino

Curso: CCI-BAC - Bacharelado em Ciência da Computação

Turma: CCI122-05U - CIENCIA DA COMPUTAÇÃO FASE 05U

Disciplina: CAL0001 - COMPLEXIDADE DE ALGORITMOS

Período letivo: 2017/1

Carga horária: 72

Professor: 211020727 - DIEGO BUCHINGER

Ementa

1. Estudo de complexidade via métodos de desenvolvimento de algoritmos. Modelos de computação e ferramentas para notação para análise de algoritmos. Algoritmos iterativos e recursivos. Solubilidade de problemas. Intratabilidade de problemas. Análise da complexidade de algoritmos clássicos na área da computação.

Objetivo geral

1. Analisar a complexidade de tempo e espaço de algoritmos. Identificar o melhor caso, o pior caso e o caso médio de execução de algoritmos. Identificar problemas tratáveis e intratáveis.

Objetivo específico

1. Capacitar o aluno a analisar a complexidade de tempo e espaço de algoritmos e ser capaz de identificar problemas considerados intratáveis.

Conteúdo programático

1. Apresentação do planejamento e da ementa da disciplina.
Introdução à disciplina.
2. Conceitos Básicos de Complexidade:
 - Notação O grande
 - Ordens de complexidade
 - Análise de complexidade com uma variável
 - Análise de complexidade de tempo de algoritmos recursivos
 - Somatórios
 - Complexidade de Espaço
 - Notação Assintótica
 - Teorema Mestre
3. Análise de Algoritmos de Ordenação: Merge Sort, Quick Sort, Heap Sort, Counting Sort e Bucket Sort
4. Análise de Complexidade de Estruturas de Dados Elementares: pilha, fila, lista encadeada, árvore binária e árvore rubro negra
Análise de Complexidade com múltiplas variáveis
Hash
5. Análise de Complexidade de operações elementares com inteiros de 'n' bits
Números primos, aritmética modular e algoritmo de criptografia RSA
6. Abordagens para Resolução de Problemas:
 - Indução matemática
 - Divisão e conquista
 - Algoritmos gulosos
 - Algoritmos de tentativa e erro
 - Programação dinâmica
 - Algoritmos de aproximação
7. Conceitos de Teoria da Computação vinculados com complexidade:
 - Problemas tratáveis e intratáveis;
 - Classes de problemas: P, NP, NP-Completo e NP-Difícil;
 - Redução de problemas;
 - Problemas NP-Completo: SAT, 3-CNF-SAT, Clique, Cobertura de Vértices, Ciclo Hamiltoniano, Caixeiro Viajante, Subset-Sum
 - Algoritmos pseudo-polinomiais

Metodologia

1. Aulas expositivas acompanhadas de trabalhos práticos relacionados aos conteúdos apresentados na aula. Aulas práticas em laboratório que objetivam a implementação ou pesquisa dos conceitos apresentados nas aulas teóricas. Listas de exercícios para auxiliar na fixação do conteúdo apresentado. Seminários para proporcionar a busca de forma autônoma pelo conhecimento. Provas teóricas para avaliar o conteúdo conceitual aprendido. Trabalhos para avaliar a capacidade do uso dos conceitos aprendidos. Até 20% do conteúdo programático poderá ser ministrado na forma de ensino a distância.

Sistema de avaliação

Plano de ensino

1. Os alunos serão avaliados com base no seu desempenho nas provas, trabalhos e listas de exercícios, sendo que ao final do semestre o aluno deverá ter comparecimento mínimo de 75% às aulas e desempenho mínimo de 70% nas avaliações. O grau de desempenho do aluno será avaliado com base no desenvolvimento das seguintes atividades e com os seguintes critérios:
 - a) Provas: 40%
 - a.1) Prova 1 (20%)
 - a.2) Prova 2 (20%)
 - b) Trabalhos: 45%
 - b.1) Trabalho/Pesquisa algoritmos de ordenação (7%)
 - b.2) Apresentação trabalho complexidade alg. Grafos: (8%)
 - b.3) Trabalho com relatório comparativo entre busca binária, árvore e hash (12%)
 - b.4) Implementação e análise do algoritmo de criptografia RSA (18%)
 - c) Exercícios: 15%
 - c.1) Análise de algoritmos simples I (1,5%)
 - c.2) Somatórios e algoritmos recursivos (3%)
 - c.3) Análise de algoritmos II - teorema mestre e complexidade de espaço (3%)
 - c.4) Implementação e análise de algoritmos - guloso, backtracking e programação dinâmica (5%)
 - c.5) Exercícios de classes de problemas (2,5%)

Bibliografia básica

1. - Algoritmos. Thomas H. Cormen, Charles E. Leiserson, Ronald L. Rivest, Clifford Stein. Campus.
- Projeto de Algoritmos com Implementações em Pascal e C. Nívio Ziviani. Cengage Learning.
- Data structures and Algorithms. Data structures and algorithms. Alfred V. Aho, John E. Hopcroft, Jeffrey D. Ullman. Addison Wesley, 1987.

Bibliografia complementar

1. - The Status of the P Versus NP Problem. Lance Fortnow. Communications of the ACM, Vol. 52 No. 9, Pages 78-86.
- Algorithms. Sanjoy Dasgupta, Christos Papadimitriou, Umesh Vazirani. McGraw Hill.

Plano de ensino

Curso: CCI-BAC - Bacharelado em Ciência da Computação

Turma: CCI122-05U - CIENCIA DA COMPUTAÇÃO FASE 05U

Disciplina: CGR0001 - COMPUTAÇÃO GRÁFICA

Período letivo: 2017/1

Carga horária: 72

Professor: 6651070 - ANDRE TAVARES DA SILVA

Ementa

1. Conceitos Básico; Dispositivos Gráficos; Sistemas de Cores; Transformações geométricas; Primitivas gráficas; Visibilidade; Rendering (modelos de iluminação, shading, textura, antialiasing).

Objetivo geral

1. Capacitar o aluno a dominar os conceitos básicos de Computação Gráfica 2D e 3D. Capacitar o aluno a implementar soluções envolvendo técnicas de Computação Gráfica.

Objetivo específico

1. CONCEITUAR computação gráfica e seu histórico.
INTRODUZIR conceitos básicos, aspectos e técnicas de construção de objetos 2D e 3D.
CONCEITUAR projeção e suas variações: paralela e perspectiva.
INTRODUZIR conceitos avançados de síntese de imagens como rendering, shading e iluminação.
CONCEITUAR técnicas de modelagem geométrica de objetos.
INTRODUZIR conceitos básicos e tipos de animações computacional.
DESENVOLVER protótipo de um sistema de visualização gráfica de objetos.

Conteúdo programático

1. Introdução Introdução a Computação Gráfica
Aplicações
Dispositivos Gráficos
Padrões de Cores (RGB, HSV, CMYK, outros)
Atualidades
2. Revisão de matemática
Transformações Geométricas e Projetivas
Câmera Virtual, Introdução ao OpenGL
3. Rendering
"Rasterização" e Preenchimento de polígonos
Modelos de Iluminação
"Shading"
4. Relacionamento espacial (Visibilidade)
5. Mapeamento de Texturas
Tópicos complementares, experimentos com OpenGL
6. Introdução Geral à Modelagem Geométrica
Curvas de Bézier, B-Spline
Geometria Sólida Construtiva (CSG)
Representação por Decomposição: Quadrees e Octrees;
Tópicos complementares, experimentos com OpenGL
7. Introdução ao Processamento e Análise de Imagens.
Introdução à Visão Computacional.

Metodologia

1. A disciplina será ministrada através de aula expositivas da teoria e aulas em laboratório de forma a provocar discussões acerca dos conteúdos e produzir implementações (programação) de soluções para os problemas propostos.

Sistema de avaliação

1. - Provas (2 provas previstas - 2 individuais e s/ consulta);
- Trabalhos individuais ou em grupos de 2 alunos, com o desenvolvimento de soluções para problemas sugeridos;
- Trabalho final individual ou em dupla;
Nota Final = $P1 * 0.3 + P2 * 0.3 + TC * 0.2 + TF * 0.2$
onde:
P1 - Prova 1
P2 - Prova 2
TC - Trabalhos complementares [Exercícios, Trabalhos (2 previstos), entre outros]

Plano de ensino

TF - Trabalho Final da Disciplina

Os estudantes terão, igualmente, a oportunidade de efetuar, ao andamento da disciplina uma avaliação mais completa do desempenho do professor e da disciplina. O formulário para esta atividade será preparado para a ocasião ou fornecido pelo coordenador do curso.

Bibliografia básica

1. ANGEL, E.. Interactive Computer Graphics: a top-down approach with OpenGL. 2.ed. Reading: Addison- Wesley, 2000.
- AZEVEDO, E. e CONCI, A . Computação Gráfica - Teoria e Prática. Editora Campus, 2003.
- FOLEY, J. et al. Computer Graphics: Principles and Practice. 2. Ed. Reading: Addison-Wesley, 1990.
- HEARN, D. e BAKER, P.. Computer Graphics - C Version. 2 ed. Prentice Hall, 1997.

Bibliografia complementar

1. Artigos dos principais eventos e revistas da área (SIGGRAPH, EUROGRAPHICS, SIBGRAPI, SBGAMES, WRVA, SVR,...).

Plano de ensino

Curso: CCI-BAC - Bacharelado em Ciência da Computação

Turma: CCI122-05U - CIENCIA DA COMPUTAÇÃO FASE 05U

Disciplina: PAP0002 - PARADIGMAS DE PROGRAMAÇÃO

Período letivo: 2017/1

Carga horária: 72

Professor: 3949795 - CRISTIANO DAMIANI VASCONCELLOS

Ementa

1. Visão comparativa dos paradigmas de linguagens de programação: imperativo, funcional, lógico e orientado a objetos. Sintaxe e semântica de linguagens de programação. Sistemas de tipos, modularização e abstrações.

Objetivo geral

1. Conhecer os paradigmas de programação e características presentes em diferentes linguagens.

Objetivo específico

1. Comparar características presentes em diferentes linguagens de programação, fornecendo subsídios a escolha da linguagem mais adequada para implementar a solução de um determinado problema.

Conteúdo programático

1. Paradigmas de Programação. Evolução das principais linguagens de programação.
2. Linguagens imperativas e linguagens funcionais.
3. Introdução a programação funcional:
 - tipos de dados primitivos;
 - declaração de funções;
 - recursão;
 - listas e tuplas;
 - polimorfismo paramétrico (universal);
 - avaliação estrita e avaliação preguiçosa.
4. Cálculo Lambda. Funções parciais e computabilidade.
5. Funções de ordem superior.
6. Tipos de dados algébricos.
7. Polimorfismo de sobrecarga.
8. Monadas.
9. Abstração de dados e modularização:
 - módulos;
 - tipos de dados abstratos;
 - programação orientada a objetos.
10. Paradigma lógico e unificação.

Metodologia

1. A disciplina será ministrada através de aula expositivas, exemplos e exercícios práticos de implementação. Até 20% do conteúdo programático poderá ser ministrado na forma de ensino a distância.

Sistema de avaliação

1. O desempenho do aluno será avaliado com base no desenvolvimento das seguintes atividades e com os seguintes critérios:
 - a) Avaliação individual (prova).
 - b) Exercícios Práticos.
 - c) Seminários.

Média = $(40 \cdot \text{Prova} + 40 \cdot \text{Exercícios} + 20 \cdot \text{Seminários}) / 100$

Bibliografia básica

1. John C. Michell; Concepts in Programming Languages. Cambridge University Press.
Robert W. Sebesta; Conceitos de Linguagens de Programação. Bookman.

Plano de ensino

Cláudio Cesar de Sá, Marcio Ferreira Silva; Haskell Uma Abordagem Prática, Novatec.
O'Sullivan, Bryan; Stewart, Donald; Goerzen, John. Real World Haskell, O'Reilly.

Bibliografia complementar

1. Lipovaca, Miran. Learn You a Haskell for Great Good!: A Beginner's Guide.
Why Functional Programming Matters. John Hughes Technical Report.
Paul Hudak, John Hughes, Simon Peyton Jones, Philip Wadler; History of Haskell: being lazy with class, The Third ACM SIGPLAN History of Programming Languages Conference. San Diego, California, June 9-10, 2007.
Ronald Garcia, Jaakko Jarvi, Andrew Lumsdaine, Jeremy G. Siek, and Jeremiah Willcock. 2003. A comparative study of language support for generic programming. In Proceedings of the 18th annual ACM SIGPLAN conference on Object-oriented programming, systems, languages, and applications (OOPSLA '03). ACM, New York, NY, USA.

Plano de ensino

Curso: CCI-BAC - Bacharelado em Ciência da Computação

Turma: CCI122-05U - CIENCIA DA COMPUTAÇÃO FASE 05U

Disciplina: PES0001 - PESQUISA OPERACIONAL

Período letivo: 2017/1

Carga horária: 72

Professor: 2939118 - CARLOS NORBERTO VETORAZZI JUNIOR

Ementa

1. Programação linear: formulação; solução gráfica; solução algébrica; método simplex; transportes; designação. Programação de projetos: conceitos fundamentais; montagem de redes; análise do caminho crítico, durações probabilísticas. Introdução à Teoria das filas: conceitos fundamentais; solução analítica. Introdução à simulação. Uso do computador para solução de problemas de pesquisa operacional.

Objetivo geral

1. CAPACITAR o aluno na formulação e resolução de problemas clássicos de pesquisa operacional

Objetivo específico

1. CONCEITUAR pesquisa operacional
CAPACITAR o aluno na formulação e e solução de problemas de programação linear.
CAPACITAR o aluno na solução de problemas de transporte e atribuição
CONCEITUAR programação de projetos
CAPACITAR o aluno na solução de redes de programação de projetos
CONCEITUAR modelos de filas
CAPACITAR o aluno na solução de modelos analíticos de filas
CONCEITUAR o uso de simulação na solução de problemas

Conteúdo programático

1. Introdução
 - Histórico
 - Escopo da Pesquisa Operacional
2. Formulação e Solução de Problemas de PL
 - Principais tipos de formulação
 - Solução gráfica
3. Solução Algébrica de Problemas de PL
 - Relação geometria-álgebra
 - O método SIMPLEX
 - Problemas especiais de formulação ,solução e interpretação
 - Prática : uso de programas para solução de problemas de programação linear
4. Problemas de Transporte
 - Método do Transporte
 - Método da Designação
5. Introdução à programação de projetos
 - Conceitos fundamentais
 - Montagem de redes / Análise do caminho crítico
 - Durações probabilísticas
 - Prática : uso de softwares de gerenciamento de projetos
6. Filas
 - Introdução
 - Modelos analíticos: um canal, vários canais, população infinita, população finita

Plano de ensino

7. Simulação
 - Introdução
 - Tipos de Simulação
 - Distribuições de probabilidade e números aleatórios
 - Simulação de problemas de filas

Metodologia

1. A disciplina será ministrada através de aula expositivas da teoria, fazendo um paralelo com situações reais através de discussões em sala, bem como práticas laboratório de informática, usando sistemas específicos para solução de problemas de Pesquisa Operacional. Até 14 horas-aula (20% da carga horária da disciplina) poderão ser desenvolvidas na modalidade à distância.

Sistema de avaliação

1. Do desempenho do aluno:

O desempenho do aluno será avaliado com base no desenvolvimento das seguintes atividades e com os seguintes critérios:

- a) participação ativa nas aulas
- b) avaliações individuais (provas).

Prova 1: Prog. Linear e SIMPLEX

Prova 2: Transportes e Programação de projetos

Prova 3: Filas e Simulação

$$\text{Média} = 0,34 \cdot P1 + 0,33 \cdot P2 + 0,33 \cdot P3$$

Do desempenho da disciplina e do professor:

Os estudantes terão, igualmente, a oportunidade de fazer, durante o andamento da disciplina, uma avaliação do desempenho do professor e do andamento da disciplina. Esta atividade será conduzida oportunamente pelo coordenador do curso.

Bibliografia básica

1. EHRLICH, Pierre Jacques. Pesquisa operacional: curso introdutório. 7. ed. São Paulo: Atlas, c1991. 322 p. : ISBN 8522407096 (broch.)
- SILVA, Ermes Medeiros da. Pesquisa operacional: programação linear, simulação. 3. ed. São Paulo: Atlas, 1998. 185 p. ISBN 8522419310 (broch.).
- ANDRADE, Eduardo Leopoldino de. Introdução à pesquisa operacional: métodos e modelos para análise de decisões. 4. ed. Rio de Janeiro: LTC, 2009 204 p. ISBN 9788521616658 (broch.).

Bibliografia complementar

1. ACKOFF, Russell Lincoln; SASIENI, Maurice W. Pesquisa operacional. Rio de Janeiro: Livros Técnicos e Científicos, 1974. 523p.-
- SHAMBLIN, James E; STEVENS, G. T. Pesquisa operacional: uma abordagem básica. São Paulo: Atlas, c1979. 426 p. ISBN (Broch.)
- TAHA, Hamdy A. Pesquisa operacional. 8. ed. São Paulo: Prentice Hall, 2008. 359 p. : ISBN 9788576051503 (broch.)

Plano de ensino

Curso: CCI-BAC - Bacharelado em Ciência da Computação

Turma: CCI122-05U - CIENCIA DA COMPUTAÇÃO FASE 05U

Disciplina: SDI0001 - SISTEMAS DISTRIBUIDOS

Período letivo: 2017/1

Carga horária: 72

Professor: 3746810 - MAURICIO ARONNE PILLON

Ementa

1. Conceitos básicos de sistemas distribuídos (coordenação e sincronização de processos, exclusão mútua, difusão de mensagens); Paradigmas de linguagens de programação distribuída; Técnicas de descrição de sistemas; Tolerância a falhas; Sistemas operacionais distribuídos; Ambientes de suporte ao desenvolvimento de sistemas distribuídos; Estudo de casos.

Objetivo geral

1. Capacitar o aluno a compreender os conceitos de projeto de sistemas distribuídos, bem como os paradigmas envolvidos no desenvolvimento de tais sistemas.

Objetivo específico

1. Compreender os Modelos Arquitetural e Fundamental de Sistemas Distribuídos;
Compreender os conceitos de Comunicação entre processos;
Aplicar ao desenvolvimento conceitos do Modelo Fundamental (falhas, segurança, etc...);
Desenvolver protótipos de sistemas distribuídos envolvendo Middleware;

Conteúdo programático

1. 01- Apresentação da disciplina
2. 02- Caracterização de sistemas distribuídos
3. 03- Modelos de sistemas
4. 04- Comunicação entre processos
5. 05- Chamada de procedimento remoto (RPC)
6. 06- Objetos distribuídos e invocação remota (RMI)
7. 07- Web Services e REST API
8. 08- Comunicação indireta
9. 09- Sistemas Operacionais: de rede e distribuído
10. 10- Sistemas de arquivos distribuídos
11. 11- Sistemas P2P
12. 12- Tempo e estado global
13. 13- Coordenação e acordos
14. 14- Replicação
15. 15- Sistema de multimídia distribuído
16. 16- Estudo de caso: google
17. Tx: Avaliação escrita do tipo Trabalho (x = 1 à 10)
18. Ex: Avaliação escrita do tipo Exercício (x = 1 à 10)
19. Ax: Aula especializada - revisão, palestra, minicurso, eventos, etc - (x = 1 à 5)
20. Px: Avaliação escrita do tipo Prova (x= 1 à 3)

Metodologia

1. Aulas Expositivo -Dialogadas; Uso de Laboratório; Desenvolvimento de atividades individuais e em grupo.

Sistema de avaliação

Plano de ensino

1. Participação em Classe;
Artigo individual apresentando solução que envolva uma técnica de construção de sistemas distribuídos
Provas (2 provas previstas - 2 individuais e s/ consulta);
Trabalhos em grupos de 2 ou mais alunos, com o desenvolvimento de soluções para problemas sugeridos;

Nota Final = $Pr1 * 0,30 + Pr2 * 0,40 + AC * 0,30$;

Pr1 - Avaliação escrita referente a primeira parte da disciplina;

Pr2 - Avaliação escrita referente a segunda parte da disciplina;

AC - Trabalhos Complementares (Exercícios, Trabalhos, entre outros)

Bibliografia básica

1. COULOURIS, G., DOLLIMORE, J., KINDBERG, T. Distributed Systems: Concepts and Design. 3a. Edição, London - UK. Editora Addison - Wesley e Pearson Education, 2001. ISBN:0201-61918-0
2. TANENBAUM, A. S., STEEN, van M. Distributed Systems: Principles and Paradigms. 1a. Edição, Prentice Hall, 2002. ISBN: 0-13-088893-1.
3. LIU, M. L. Distributed Computing: Principles and Applications. 1a. Edição, California - USA, Addison-Wesley, 2004. ISBN: 0-201-79644-9

Bibliografia complementar

4. HORSTMANN, C. S., CORNELL, G. Core Java2 - Volume II - Recursos Avançados. ISBN 853461253-6. Makron Books - São Paulo, 2001.
5. WU, J. Distributed Systems Design. Florida - USA, CRC Press LLC, 1999. ISBN: 0849331781
6. LAGES, N. A. de C., NOGUEIRA, J. M. S. Introdução aos Sistemas Distribuídos. Campinas - SP - BR, 1986. ISBN:000896195
7. OAKS, S., WONG, H. JINI in a Nutshell. 1a. Edição, O'Reilly, 2000. ISBN: 1565927591
8. BOWMAN, H., DERRICK, J. Formal Methods for Distributed Processing: A Survey of Object Oriented Approach, Edited by Howard Bowman e John Derrick. Cambridge CB2 2RU - UK, Cambridge University Press, 2001. ISBN:0521771846
9. ECKEL, B. Thinking in Java. 3a Ed. Prentice Hall (disponível em formato eletrônico), 1998.
10. RITCHEY, T. Programando com Java. Ed. Campus, 1996.
11. JOSEPH, Joshy, FELLENSTEIN, Craig. Grid Computing Prentice Hall PTR, 2003. ISBN 0131456601

Plano de ensino

Curso: CCI-BAC - Bacharelado em Ciência da Computação

Turma: CCI122-05U - CIENCIA DA COMPUTAÇÃO FASE 05U

Disciplina: TEC0001 - TEORIA DA COMPUTAÇÃO

Período letivo: 2017/1

Carga horária: 72

Professor: 1033139444 - KARINA GIRARDI ROGIA

Ementa

1. Histórico e contextualização da Computação. Máquinas de Turing. Formalização do conceito de algoritmo. Problema da Parada. A Tese de Church-Turing. Indecidibilidade. Noções de Redutibilidade. Algoritmo/Máquina de Post. Algoritmo/Máquina de Markov. Máquina de Registradores. Lambda Calculus. Teoria das funções recursivas. Relações entre os modelos de computabilidade e suas equivalências.

Objetivo geral

1. Capacitar o aluno para a aplicação sistematizada e formalizada de conceitos e resultados relativos à computabilidade e à decidibilidade de linguagens e problemas, e à complexidade de tempo de algoritmos.

Objetivo específico

1. Conceituar as Máquinas de Turing e estruturas de poder computacional equivalente;
Apresentar equivalências de modelos computacionais;
Apresentar a Tese de Church-Turing.;
Conceituar a decidibilidade de linguagens e problemas;
Capacitar o aluno na prova por redução de linguagens e problemas;
Introduzir os conceitos de complexidade de tempo.

Conteúdo programático

1. Histórico e contextualização da computação
2. Máquinas de Turing - definição e exemplos
3. Reconhecedores e decidores
4. Máquinas de Turing multifitas
5. Máquinas de Turing não determinísticas
6. Enumeradores
7. Decidibilidade de linguagens
8. Diagonalização
9. Problema da parada
10. Problemas indecidíveis em linguagens
11. Históricos de computação
12. Problema da correspondência de Post
13. Redução por mapeamento
14. Notação assintótica e análise de algoritmos
15. Classe P
16. Classe NP
17. P versus NP
18. NP-Completeness
19. Teorema de Cook-Levin

Metodologia

1. A disciplina será desenvolvida através de aulas expositivo-dialogadas com a participação ativa dos alunos. A cada semana serão passadas referências bibliográficas com o conteúdo a ser abordado na semana seguinte. Com o auxílio da professora, as aulas expositivas serão a partir de questionamentos diretos aos alunos e a avaliação será contínua levando em conta o

Plano de ensino

trabalho de pesquisa e estudos prévios do aluno. Serão desenvolvidos também exercícios práticos orientados em sala de aula e exercícios a serem passados para resolução extra-classe. Também será desenvolvido trabalho final escrito a ser apresentado em sala de aula. O conteúdo da disciplina poderá ser ministrado na modalidade de ensino a distância em até 20% do total de sua Carga Horária (MEC PORTARIA Nº 4.059, DE 10 DE DEZEMBRO DE 2004 publicado no DOU de 13/12/2004, Seção 1, p. 34).

Sistema de avaliação

1. Do desempenho do aluno:

A qualidade do desempenho do aluno será avaliada com base no desenvolvimento das seguintes atividades e com os seguintes critérios:

- a) Avaliação contínua englobando participação em aula (com conteúdo assimilado), resolução de exercícios e assiduidade (AC);
- b) Provas individuais (P1 e P2);
- c) Trabalho final escrito (T).

A Média Semestral (MS) será calculada pela fórmula

$$MS = 0,25 * AC + 0,3 * (P1 + P2) + 0,15 * T$$

Do desempenho da disciplina e do professor:

Os estudantes terão, igualmente, a oportunidade de fazer uma avaliação do desempenho do professor e da disciplina. As informações sobre esta atividade serão fornecidas pelo coordenador do curso.

Das regras para revisão das avaliações:

Depois da publicação das notas pelo professor, os alunos têm 07 dias corridos para solicitar a revisão com o professor. Esta revisão será feita na sala do professor, preferencialmente em horário de atendimento aos alunos, ou em um horário do qual o professor possa atendê-los.

Bibliografia básica

1. SIPSER, Michael. Introdução à Teoria da Computação. 2a ed. São Paulo: Cengage Learning, 2012. ISBN 8522104994

HOPCROFT, John E.; ULLMAN, Jeffrey D.; MOTWANI, Rajeev. Introdução à Teoria de Autômatos, Linguagens e Computação. 2a ed. Rio de Janeiro: Elsevier, 2002. ISBN 9788535210792

FORTNOW, Lance. The status of the P versus NP problem. Commun. ACM 52, 9 (September 2009), 78-86. 2009.
DOI=<http://dx.doi.org/10.1145/1562164.1562186>

Bibliografia complementar

1. DIVERIO, Tiaraju A.; MENEZES, Paulo B. Teoria da computação: máquinas universais e computabilidade. 3. ed. Porto Alegre: Bookman, 2011. ISBN 9788577808243

LEWIS, Harry R.; PAPADIMITRIOU, Christos H. Elementos de Teoria da Computação. 2a ed. Porto Alegre: Bookman, 2000. ISBN 0132624788

Plano de ensino

Curso: CCI-BAC - Bacharelado em Ciência da Computação

Turma: CCI122-06U - CIENCIA DA COMPUTAÇÃO FASE 06U

Disciplina: ACT0001 - AUTOMAÇÃO E CONTROLE

Período letivo: 2017/1

Carga horária: 72

Professor: 3088693 - ROBERTO SILVIO UBERTINO ROSSO JUNIOR

Ementa

1. Sistemas de Manufatura. Introdução à Automação da Manufatura. Equipamentos Industriais. Sistemas de Software. Integração e Controle.

Objetivo geral

1. Os objetivos desta disciplina são: cobrir de maneira introdutória os conteúdos da área de Automação e Controle da forma que é especificada no currículo de referência da Sociedade Brasileira de Computação (SBC), fornecer subsídios para que o estudante possa compreender os problemas específicos associados aos problemas de automação das empresas, em a automação e controle nas indústrias, produzindo soluções adequadas a esse perfil, bem como fornecer um diferencial para o mercado de trabalho nacional.

Objetivo específico

1. OBJETIVOS ESPECÍFICOS/DISCIPLINA :
 - . CONCEITUAR Engenharia de Produto e Engenharia de Processos.
 - . INTRODUZIR o processo de projeto, análise, planejamento, operação e controle automático da manufatura de produtos.
 - . APRESENTAR os sistemas computacionais envolvidos nestes processos bem como alguns equipamentos.
 - . CARACTERIZAR os tipos de dispositivos empregados na automação da manufatura e seus usos.
 - . APRESENTAR a integração e automação da manufatura a partir dos sistemas computacionais e equipamentos apresentados e caracterizados na disciplina.

Conteúdo programático

1. 1 INTRODUÇÃO
Sistemas de Manufatura.
 - O que é manufatura
 - Processos de FabricaçãoModelos de Produção (celular, por produto, por processo, sequencial, tecnologia de grupo, ...)
2. 2 ENGENHARIA DE PRODUTO
 - 2.1 Introdução a Engenharia de Produto e Processos
 - 2.2 O Processo Convencional de Projeto (Design)
 - 2.3 Descrição das Atividades no Design e na Manufatura
 - 2.4 Definição e Justificação da Engenharia Simultânea
 - 2.5 Projeto para Montagem e Manufatura
3. 3 SISTEMAS DE CAD
 - 3.1 Introdução, situar sistemas de CAD no design
 - 3.2 Evolução do CAD (2D, 2,5 D, Wireframe, Sólidos, Superfícies)
 - 3.3 Sólidos (CSG, B-rep, DSG)
4. 4 OUTROS SISTEMAS CLÁSSICOS DO CIM
 - 4.1 Noções sobre CAE (Computer-Aided Engineering)
 - 4.2 Noções sobre CAPP (Computer Aided Process Planning)
 - 4.3 Noções sobre CAM (Computer Aided Manufacturing)
 - 4.4 Sistemas baseados em Features (Definições, DbF, FeR, Validação)
 - 4.5 Interfaces para integração de sistemas
5. 5 INTRODUÇÃO AUTOMAÇÃO DA MANUFATURA
 - 5.1 Equipamentos Industriais. CLP, CNC, Atuadores, Sensores, Robôs
6. 6 SISTEMAS DE CONTROLE
 - Em malha aberta
 - Em malha fechada
7. 7 Integração e Controle.
 - Arquiteturas
 - Integração da Manufatura (CIM, EMI, ...)
 - Supervisórios

Metodologia

Plano de ensino

1. A disciplina será ministrada através de aula expositivas da teoria, discussão de conteúdos, palestras oportunas, visitas a laboratórios, desenvolvimento e apresentação de trabalhos. Até 14 horas/aula (menos que 20% da carga horária da disciplina) poderão ser desenvolvidas na modalidade à distância.

Sistema de avaliação

1. Do desempenho do aluno:
O desempenho do aluno será avaliado com base no desenvolvimento das seguintes atividades e com os seguintes critérios:
a) participação ativa nas aulas e nos trabalhos;
b) elaboração e apresentação dos trabalhos, e;
c) avaliações individuais, podendo ser uma ou mais provas ou trabalhos;
d) as avaliações poderão ter mais de um componente.

Os pesos das avaliações estão assim distribuídos:
Avaliação 01(25%), Avaliação 02(35%) e Avaliação 03 (40%)

Do desempenho da disciplina e do professor:
Os estudantes terão, igualmente, a oportunidade de fazer, durante o andamento da disciplina, uma avaliação do desempenho do professor e do andamento da disciplina. Esta atividade ocorre semestralmente no CCT.

Bibliografia básica

1. GROOVER, Mikell P. Automação Industrial e Sistemas de Manufatura. São Paulo: Pearson Prentice Hall. 3 ed., 2011
LORINI, Flavio José. Tecnologia de grupo e organização da manufatura. Florianópolis: UFSC, 105p. 1993.
PAZOS, Fernando. Automação de sistemas e robótica. Rio de Janeiro : Axcel Books, 377 p. 2002. ISBN 8573231718.

Bibliografia complementar

1. CHANG, T.-C.; WYSK, R. A.; WANG, Hsu-Pin .Computer-Aided Manufacturing. Englewood Cliffs: Prentice Hall. 1998. ISBN: 013754524x
CRAIG, John J., Robótica. São Paulo: Pearson Education do Brasil, 3. ed.. 2012.
FERREIRA, A.C. ,Comando Numérico ,UFSC/EMC/GRUCON,1994
FERREIRA, J.C.E., Planejamento do Processo Assistido por Computador - CAPP, Apostila, 2a Edição, Universidade Federal de Santa Catarina, Departamento de Engenharia Mecânica, Florianópolis, 2002.
FOLEY, J. D; van DAM, A; FEINER, S. K. and Hughes, J. F. Computer Graphics: Principles and Practice. 2 nd Edition in C. Addison-Wesley. 1996.
GROOVER, Mikell P. et alii Robótica: Tecnologia e Programação. McGraw-Hill. 1989.
IEEE transactions on robotics. (Disponível on-line na UDESC)
IEEE Transactions on Industrial Informatics. (Disponível on-line na UDESC)
KIEF, H.B.; WATERS, T.F. ,Computer Numerical Control - A CNC Reference Guide, Glencoe: Macmillan/McGraw-Hill ,1992
McMAHON C and Browne, J. CAD/CAM: From Principles to Practice. Addison-Wesley. 1993.
MOELLER, Klocner. Controlador Programável - LPC 40. Indumatic. 1993.
NAGY, F. N.; SIENGLER, A.. Engineering Foundations of Robotics. Prentice-Hall Internacional, 1987.
NOVASKI, Olívio. Introdução a engenharia de fabricação mecânica. São Paulo: E. Blucher, 2008. 119 p. ISBN 9788521201625 (broch.).
OGATA, K. , Engenharia de Controle Moderno, Prentice Hall do Brasil, 2003. ISBN 8587918230
REMBOLD, U; Nnaji, B. O. and Storr, A. Computer Integrated Manufacturing and Engineering. Addison-Wesley. 1993.
SHAH, J. J. and Mantyla; M. Parametric and Feature-based CAD/CAM. John Wiley and Sons. 1995.
SYAN, C. S. and MENON, U. Concurrent Engineering: Concepts, Implementation and Practice. London: Chapman & Hall, 1994. 234 p. ISBN 0412581302 (enc.)
SIEMENS. Manual de CLP, s.l., s.d
SCHILLING, Robert J. Fundamentals of robotics: analysis and control. New Jersey: Prentice Hall, c1990. 425 p. ISBN 0133444333 (enc.)
ZEID, I. CAD/CAM: Theory and Practice. McGraw-Hill International Editions (Computer Science Series).1991.

Plano de ensino

Curso: CCI-BAC - Bacharelado em Ciência da Computação

Turma: CCI122-06U - CIENCIA DA COMPUTAÇÃO FASE 06U

Disciplina: EMI0001 - EMPREENDEDORISMO EM INFORMÁTICA

Período letivo: 2017/1

Carga horária: 36

Professor: 0398817 - Célia Regina Beiro da Silveira

Ementa

1. A informática como área de negócios: análise dos diversos setores de mercado, suas características e tendências; O empreendimento e o empreendedor; Técnicas de negociação; Desenvolvimento organizacional; Qualidade Total; Política Nacional de Informática; Planejamento de Empreendimentos em informática.

Objetivo geral

1. Entender os fundamentos de administração de empresas de tecnologia e os aspectos relacionados as características empreendedoras ressaltando a importância destas para o indivíduo empreendedor ou intraempreendedor.

Objetivo específico

1. Conhecer os principais conceitos ligados à Administração de Empresas e Empreendedorismo.
Identificar características empreendedoras.
Sensibilizar os alunos quanto a importância do empreendedorismo para a área de tecnologia.
Conhecer e analisar o contexto histórico e atual da administração empreendedora.
Desenvolver técnicas de negociação
Elaborar um plano de Negócios.

Conteúdo programático

1. Apresentação da Disciplina. Plano de Ensino e critérios de avaliação.
2. Dinâmica de Grupo.
3. Estudo de Caso.
4. Empreendedorismo no Brasil
5. Características Empreendedoras. Busca de Oportunidades e Ideias
6. Modelo Canvas
7. Análise da Concorrência
8. Matriz FOFA
9. Missão- Visão e Valores
10. Análise do mercado e público Alvo
11. Fornecedores
12. Plano de Marketing
13. Plano Operacional
14. Parte Financeira - Necessidade de Pessoal - Investimento Inicial - Faturamento - Impostos - Depreciação - CMD e CMV - Custos Fixos - Pré-Operacional - Demonstrativo de Resultados - Cenários
15. Pesquisa de Campo
16. Apresentação Plano de Negócios

Metodologia

1. A disciplina será trabalhada através da temática teórico-prática, com ênfase na discussão ao debate em grupo, assim como uma forte carga de leitura, aulas expositivas, filmes, estudos de caso e dinâmicas que complementarão o trabalho. Todas as aulas serão seguidas de atividades práticas e vivenciais dando ênfase ao referencial teórico administrado. As leituras dos estudos de caso, interpretação de filmes e dinâmicas e os trabalhos em grupos serão significativos para elaboração do Plano de negócio solicitado para o término da disciplina
Para a elaboração do plano de negócios, será necessária a utilização do laboratório e esclarecimento de dúvidas, bem como no decorrer do semestre acontecerá visita a incubadora tecnológica.

Plano de ensino

Sistema de avaliação

1. 1) Busca de Informações e ideias: 5%
- 2) Parte teórica do negócio: 15%
- 3) Parte Financeira do Negócio: 10%
- 4) Plano de Negócios: 30%
- 5) Apresentação do Plano de negócios: 40%

Bibliografia básica

1. ALMEIDA, F. Como ser empreendedor de sucesso. Belo Horizonte: Leitura Empresarial, 2001.
- AYAN, J. As 10 maneiras de libertar seu espírito criativo e encontrar grandes idéias. São Paulo: Negócio Editora, 2001.
- DORNELAS, J.C.A. Empreendedorismo: Transformando idéias em negócio. Rio de Janeiro: Campus, 2001.

Bibliografia complementar

1. GERBER, Michael E; GERBER, Michael E. O mito do empreendedor : como fazer de seu empreendimento um negocio bem-sucedido. 3 ed. São Paulo: Saraiva, 1992. 1992 p. ISBN 8502007270
- FILION, Louis Jacques; DOLABELA, Fernando. Boa Ideia! E agora?: plano de negócio, o caminho seguro para criar e gerenciar sua empresa. São Paulo: Cultura, c2000. 344 p. ISBN 8529300580 (broch.).

Plano de ensino

Curso: CCI-BAC - Bacharelado em Ciência da Computação

Turma: CCI122-06U - CIENCIA DA COMPUTAÇÃO FASE 06U

Disciplina: IAR0001 - INTELIGÊNCIA ARTIFICIAL

Período letivo: 2017/1

Carga horária: 72

Professor: 3605655 - RAFAEL STUBS PARPINELLI

Ementa

1. Histórico. Conceitos e motivações. Jogos e problemas de IA. Métodos informados e não-informados de busca. Heurísticas. Tipos de raciocínio. Representação do conhecimento. Uso da lógica em processos de raciocínio. Cálculo de incertezas. Aplicações. Noções de Paradigmas Bio-inspirados.

Objetivo geral

1. Mostrar técnicas básicas e modernas de Inteligência Artificial bem como suas implementações práticas.

Objetivo específico

1. CONCEITUAR os princípios básicos da Inteligência Artificial
2. INTRODUZIR a prática nesse domínio
3. PROPORCIONAR relações com outros conhecimentos obtidos no curso

Conteúdo programático

1. Plano de Aula
Apresentação do Plano de Aula
2. Introdução
Conceitualização da disciplina;
Escopo da disciplina;
Terminologia;
3. Agentes Inteligentes
Conceitualização de agentes
Definição do ambiente
Estudos de caso
Comportamento Emergente
4. Sistemas de Produção
Definições; Exemplos; Algoritmos de busca cega.
5. Busca Cega
Definição de espaço de busca;
Caracterização de problemas;
Busca em largura e profundidade;
Busca de custo uniforme;
6. Busca Heurística
Definição de Heurística e informação;
Algoritmo A*;
Recozimento Simulado.
7. Avaliação
Avaliação referente ao conteúdo dado em sala.
8. Fixação de conteúdo
Atividades para fixação de conteúdo.
9. Semana da Computação
Atividades relacionadas à Semana da Computação.
10. Laboratório
Implementação e desenvolvimento de algoritmos relacionados em sala.
11. ACO
Otimização por Colônia de Formigas;
Aplicação a problemas combinatoriais;
TSP.
12. Algoritmos Genéticos
Fundamentos;
Teoria;
Aplicações;
Ferramentas.

Plano de ensino

13. Temas
Algoritmos Meta-heurísticos em GPU;
Programação por Expressão Gênica;
Redes Neurais Artificiais;
Lógica Fuzzy;
Ecossistema Computacional para Otimização.

Metodologia

1. A disciplina será desenvolvida através de aula expositivas da teoria, com exercícios práticos em sala de aula, exercícios práticos em laboratórios e trabalhos práticos de laboratórios. Até 20% da carga horária poderá ser desenvolvida à distância.

Sistema de avaliação

1. A qualidade do desempenho do aluno será avaliada com base no desenvolvimento das seguintes atividades e com os seguintes critérios:
- a) participação ativa nas aulas e na solução dos exercícios práticos em sala de aula e em laboratório.
 - b) elaboração e apresentação de trabalho prático de laboratório.
 - c) avaliações individuais (provas)
 - d) avaliações individuais (seminários)

A nota final será composta com o seguinte critério:

PROVA#1*0,30 + PROVA#2*0,30 + TRABALHOS*0,25 + SEMINÁRIO*0,15

Bibliografia básica

1. RUSSEL, Stuart, NORVIG Peter. Inteligência Artificial. 2004.
WINSTON, Patrick H. Artificial Intelligence. (3rd. edition) Addison-Wesley Publishing, 1992.
REZENDE, Solange Oliveira. Sistemas inteligentes: fundamentos e aplicações. São Paulo: Manole, c2005. 525 p.
BITTENCOURT, G. Inteligência Artificial - Ferramentas e Teorias. Editora da UFSC. 3ª ed. Florianópolis, 2001.
RICH, E., KNIGHT, K. Inteligência Artificial. Makron Books. 2ªed. São Paulo, 1994.

Bibliografia complementar

1. NILSSON, N.J. Principles of Artificial Intelligence. Springer-Verlag, 1982.
ROWE, N.C. Artificial Intelligence Through Prolog. Prentice Hall, 1988.
BARR, A. & FEIGERNBAUM, E.A. The handbook of Artificial Intelligence. Los Altos: William Kaufmann, 1981.
BITTENCOURT, Guilherme. Inteligência Artificial: ferramentas e teoria. Florianópolis: Editora da UFSC, 1998.
CHARNIAK, E & McDERMOTT, D. Introduction to Artificial Intelligence. Addison-Wesley, 1985.
DREYFUS, H. What Computers Still Can't Do: A Critique of Artificial Reason. MIT Press, 1992.
GENESSERETH, M.R. & NILSSON, N. Logical Foundations of Artificial Intelligence. Palo Alto: Morgan Kaufmann, 1988.
MITCHEL, Melanie. An Introduction to Genetic Algorithms. MIT Press, 1996.
NILSSON, N.J. Problem Solving Method in Artificial Intelligence. New York: McGraw-Hill, 1971.
RICH, E. & KNIGHT, K. Artificial Intelligence. 2nd edition. McGraw-Hill, 1991.
SCHALKOFF, R.J. Artificial Intelligence: An Engineering Approach. McGraw-Hill, 1990.
WITTEN, I. H; FRANK, Eibe. Data mining: practical machine learning tools and techniques. 2nd ed. Califórnia: Morgan Kaufmann Publishers, 2007. 525 p.
MITCHELL, Tom. Machine Learning, McGraw Hill, 1997.
HAYKIN, S. Redes Neurais: Princípios e Prática, 2nd. Edition, Bookman, 2001.

Plano de ensino

Curso: CCI-BAC - Bacharelado em Ciência da Computação

Turma: CCI122-06U - CIENCIA DA COMPUTAÇÃO FASE 06U

Disciplina: IHC0001 - INTERAÇÃO HOMEM COMPUTADOR

Período letivo: 2017/1

Carga horária: 72

Professor: 3630013 - ISABELA GASPARINI

Ementa

1. Conceitos básicos de Interação Humano-Computador (IHC): Interfaces, Sistemas Interativos, Usuários, Usabilidade, Componentes de Software e Hardware. Fundamentos teóricos. Projeto e Prototipação de Interfaces. Avaliação de Interfaces: tipos e técnicas de avaliação. Interfaces Web. Interfaces Avançadas e Novas Tendências.

Objetivo geral

1. Aplicar os fundamentos de Interfaces para o projeto e a construção de interfaces.

Objetivo específico

1. Capacitar o aluno a:
 1. Conhecer os fundamentos de interfaces e investigar a usabilidade em interfaces humano-computador.
 2. Projetar uma interface com Usabilidade.
 3. Aplicar o conceito de avaliação de interfaces.
 4. Identificar as diferenças entre os diversos estilos e paradigmas de interação.
 5. Desenvolver espírito crítico e consciência dos pressupostos éticos que regulamentam as pesquisas envolvendo seres humanos.

Conteúdo programático

1. Apresentação da Disciplina e do Plano de Ensino
Apresentar a Disciplina
Apresentação do Plano de Ensino
Diagnóstico da Turma
2. Conceitos iniciais;
Conceitos Básicos de Interação Humano- Computador (IHC):
 - Motivação, Histórico, Mapeamento da Área
 - Design do dia a dia; TIC;
 - Conceitos básicos sobre Interação e Interface, affordance e Design da Interação
 - Princípios de Sistemas Interativos
 - Arquitetura, Princípios de design.
3. Conceitos de IHC:
 - Qualidade de Uso: Usabilidade, Experiência do Usuário, Comunicabilidade, Acessibilidade;
 - Aspectos Envolvidos: Usuários, Tarefas, Tecnologias e Contexto
 - Qualidade na Interação
 - Interfaces de qualidade: características principais, conceito de usabilidadeParadigmas da Comunicação Humano-Computador
Estudo de aspectos teórico-práticos do desenvolvimento da interação humano-computador e o conceito de usabilidade
 - Estilos de interação
4. Aspectos Éticos e Sociais em IHC
 - Conduta profissional
 - Legislação em pesquisa envolvendo seres humanos
 - Aspectos éticos e sociais de pesquisas envolvendo pessoas
5. Ergonomia de Interfaces Humano-Computador
Fundamentos da Ergonomia de IHC, Qualidades ergonômicas para IHC, As técnicas da Ergonomia
Ergonomia de Interfaces Humano-Computador para desenvolvimento de interfaces Web
 - Fundamentos da Ergonomia de IHC
 - Qualidades ergonômicas para IHC
6. Concepção de Interfaces
Introdução a concepção de interfaces
Princípios básicos
domínio do conhecimento
ciclo de desenvolvimento

Plano de ensino

tipos de concepção usabilidade como requisito do sistema questões de concepção Análise contextual, Projeto (diálogo e apresentação)- design conceitual e físico - prototipação, construção (por exemplo mapa conceitual, storyboarding, navegação na web, prototipação de alta e baixa fidelidade), ferramentas, Avaliação
7. Interfaces WIMP x Web Introdução Diferenças e similaridade
8. Processo de Webdesign Introdução Processos de desenvolvimento Etapas: Definição e planejamento, arquitetura, projeto dos componentes, desenvolvimento Elementos do conteúdo Princípios de diretrizes, padrões e guias (por exemplo as regras de ouro, usabilidade na web do Nielsen, guias de usabilidade e acessibilidade) Fundamentos para o projeto de Interfaces Web - Fundamentos de Web Design - Processo de desenvolvimento do sistema - Usabilidade e navegabilidade em Interfaces Web - Uso de guidelines em Webdesign - Dicas e Erros mais comuns em Web Design - Design da Interface
9. Avaliação de Interface - Visão geral - Conceito - Problemas de usabilidade - Avaliação - Técnicas de Avaliação: abordagem em relação a etapa do ciclo de vida (formativa versus somativa); abordagem em relação ao usuário (sem versus com usuário) - Comparações e classificações
10. Acessibilidade Introdução, conceitos, diretrizes e checklist, avaliação automática sistemas
11. Tópicos de inovação e tendências Tópicos de Inovações e Tendências na área de IHC, tais como: User experience Sistemas adaptativos, Modelo do Usuário e Adaptabilidade/personalização Sistemas cientes/sensíveis ao contexto (context-aware systems) Sistemas/interfaces móveis e/ou ubíquos, Responsive webdesign Sistemas colaborativos interação humano-robô Questões culturais, Dimensões culturais - internacionalização e localização de interfaces Sistemas de Recomendação Visualização da informação Engenharia baseada em cenários Web responsiva Funology, gamification, jogos Acessibilidade Realidade virtual e aumentada, 3D, Interação 3D Ambientes EAD e IHC Interação natural, tangível, vestíveis, modal e multitouch Web semântica e ontologia Emoção em IHC Experimentos -com métricas estatística em IHC Qualidade Novas tendências *Tópicos a serem escolhidos conforme tendências e projetos inovadores na área.
12. Prova da disciplina
13. Apresentação Trabalho em sala
14. Palestra Palestras Oportunas com profissionais da área
15. Desenvolvimento do trabalho Aplicação do Aluno

Plano de ensino

- Aplicação prática para o aluno
- Utilização dos conceitos de IHC vistos em sala
- Desenvolvimento
- Avaliação do sistema proposto

16. Fechamento disciplina e divulgação notas e médias

Metodologia

1. A disciplina será ministrada através de aulas expositivas da teoria, discussões do conteúdo, palestras oportunas e apresentação de trabalhos. Até 20% da carga horária será desenvolvida à distância.

Sistema de avaliação

1. SISTEMA DE AVALIAÇÃO:

Do desempenho do aluno:

O desempenho do aluno será avaliado com base no desenvolvimento das seguintes atividades e com os seguintes critérios:

- a) avaliações individuais (provas e testes) - P (30%)
- b) elaboração e apresentação de trabalhos (escritos e orais) - T (60%)
- c) exercícios práticos e participação ativa nas aulas e nos trabalhos- uma nota que reflete o nível e a qualidade da participação do aluno durante os encontros, de forma a ajudar a manter a disciplina, envolvimento e motivação da turma sobre os temas em quantidade e objetividade adequada de forma a facilitar alcançar os objetivos do curso. - E (10%)

Do desempenho da disciplina e do professor:

A coordenação do curso fará a avaliação durante o semestre.

Das regras para revisão das avaliações:

Depois da publicação das notas pelo professor, os alunos têm 07 dias corridos para solicitar a revisão com o professor. Esta revisão será feita na sala do professor, preferencialmente em horário de atendimento ao aluno, ou em um horário do qual o professor possa atendê-los.

Bibliografia básica

1. ROGERS, Yvonne; SHARP, Helen; PREECE, Jennifer. Design de interação: além da interação humano-computador, Bookman, 3ª. edição, 2013.
PREECE, Jennifer; ROGERS, Yvonne; SHARP, Helen. Design de interação: além da interação homem-computador, Bookman, 2005.
BARBOSA, S.D.J.; SILVA, B.S. Interação Humano-Computador. Editora Campus-Elsevier, 2010.
ROCHA, Heloísa V. da; BARANAUSKAS, Maria C. C. Design e Avaliação de interfaces humano-computador, NIED/UNICAMP, 2003.
NIELSEN, Jakob. Usability Engineering, Academic Press, 1993.
CYBIS, Walter Otto; BETIOL, Adriana Holtz; FAUST, Richard. Ergonomia e usabilidade: conhecimentos, métodos e aplicações, 2. ed. rev. e ampl. São Paulo:Novatec, 2010.
CYBIS, W.; BETIOL, A.; FAUST, R. Ergonomia e Usabilidade: Conhecimentos, Métodos e Aplicações, Novatec, 2007.
NIELSEN, J. Projetando websites, Campus, 2000.

Bibliografia complementar

1. SHNEIDERMAN, Ben; PLAISANT, Catherine. Designing the User Interface: Strategies for Effective Human-Computer Interaction, 5th edition. Addison-Wesley, 2009.

DIX, Alan; FINLAY, Janet, ABOUW, Gregory; BEALE, Russell. Human-Computer Interaction. 3rd Edition. Prentice Hall, 2004.

RUBIN, Jeffrey. Handbook of Usability Testing: How to Plan, Design, and Conduct Effective Tests. New York: Wiley, 1994.

LYNCH, Patrick J; HORTON, Sarah. Web Style Guide, Yale University. Disponível em: <http://www.webstyleguide.com>

NIELSEN, Jakob; Loranger, Hoa. Prioritizing Web Usability, New Riders, 2006.

HORTON, Sarah. Access by Design: A Guide to Universal Usability for Web Designers, Voices, 2006.

BOWMAN, D.; KRUIJFF, E.; LAVIOLA, J. J. Jr; POUPYREV, I. 3D User Interfaces: Theory and Practice, Addison-Wesley, 2004.

PREECE, J., ROGERS, Y., SHARP, H., BENYON, D., Holland, S. & CAREY, T. Human-Computer Interaction. Wokingham, UK: Addison-Wesley 2002.

Plano de ensino

BARNUM, Carol M. Usability testing essentials: ready, set-- test. Amsterdam: Morgan Kaufmann Publishers, c2011. 382 p. (Human-Computer Interaction / Web Design.). ISBN 9780123750921 (broch.).

DUMAS, Joseph S.; LORING, Beth A. Moderating usability tests: principles and practice for interacting . Amsterdam: Elsevier, 2008. 185 p. ISBN 978-0-12-373933-9 (broch)

CAIRNS, Paul; COX, Anna L. (Ed.). Research methods for human-computer interaction. New York, NY: Cambridge University Press, 2008. 242 p. ISBN 9780521690317 (broch.).

LAZAR, Jonathan; FENG, Jinjuan Heidi; HOCHHEISER, Harry. Research methods in human-computer interaction. United Kingdom: Wiley, 24 cm. 426 p. ISBN 9780470723371 (broch.).

SHNEIDERMAN, Ben; PLAISANT, Catherine. Designing the user interface: strategies for effective human-computer interaction. 5th ed. New York, NY: Addison Wesley, c2010. 606 p. (Software engineering / User interface.). ISBN 9780321537355 (enc.).

MAYHEW, Deborah J. The Usability engineering lifecycle: a practitioner's handbook for user interface design . Califórnia: Morgan Kaufmann, c1999. 542 p. (The Morgan Kaufmann series in interactive technologies.). ISBN 9781558605619 (broch.).

RUBIN, Jeffrey; CHISNELL, Dana. Handbook of usability testing : how to plan, design, and conduct effective tests. 2. ed. Indianapolis, IN: Wiley, c2008. 348 p. ISBN 9780470185483 (broch.).

Plano de ensino

Curso: CCI-BAC - Bacharelado em Ciência da Computação

Turma: CCI122-06U - CIENCIA DA COMPUTAÇÃO FASE 06U

Disciplina: MFO0001 - MÉTODOS FORMAIS

Período letivo: 2017/1

Carga horária: 72

Professor: 2864983 - CINARA TEREZINHA MENEGAZZO

Ementa

1. Estudo de técnicas formais. Classificação de modelos formais. Concepção de sistemas: especificação, verificação e validação. Apresentação e aplicação de métodos e linguagens de especificação formal.

Objetivo geral

1. Compreender o que é desenvolvimento formal de software (motivação, ciclo de desenvolvimento, verificação X validação, grau de formalismo, classificação de métodos baseados em modelos e orientados a propriedades). Conhecimento dos principais métodos formais e semi-formais empregados.

Objetivo específico

1. 1) Conceituar especificação formal
2) Estudar as categorias de especificações semi-formais
3) Estudar as categorias de especificações formais
4) Comparar os métodos formais existentes
5) Estudar os principais métodos formais utilizados atualmente

Conteúdo programático

1. 1. Apresentação Disciplina e Métodos de Avaliação
2. Visão Geral
1.1. Programação em Pequena e Larga escala
1.2. Ferramentas automatizadas
1.3. Arquitetura de ambientes de desenvolvimento de sistemas
1.4. Especificação e ciclo de vida de desenvolvimento de software
1.5. Reutilização de Especificações
2. 1. Especificações Semi-Formais
1.1 Categorias de linguagens semi-formais
1.2 Avaliação de métodos semi-formais
1.3 Ferramentas automatizadas
3. 2. Especificação Algébrica
2.1 Linguagem
2.2 Processo de Construção
2.3 Dificuldades de uso das especificações algébricas
4. 3. Especificação Baseada em Modelos
3.1 Contextualização
3.2 Linguagens de especificação
3.3 Exemplos
5. 4. Especificação de Sistemas Concorrentes Usando o Modelo Operacional
4.1 Sintaxe
4.2 Semântica Operacional
6. 5. Estudo de Métodos Formais de Especificação
5.1 Especificação de programas (Operacional, Modelo Abstrato, etc.)
5.2 Especificação de Sistemas (VDM, FDM, LOTTOS, etc.)
5.3 Sistemas com Concorrência (Estelle, Z, etc.)

Metodologia

1. Aulas expositivas. Aula dialogada. Leitura de textos referentes aos temas abordados para discussão, estudos dirigidos e seminários. Trabalhos individuais e em grupo.
O conteúdo da disciplina poderá ser ministrado na modalidade de ensino a distância em até 20% do total de sua Carga Horária (MEC PORTARIA No 4.059, DE 10 DE DEZEMBRO DE 2004 publicado no

Plano de ensino

DOU de 13/12/2004, Seção 1, p. 34).

Sistema de avaliação

1. Provas escritas (P): P1, P2, P3;
Trabalho Final (TF);
Listas de Exercícios;
Apresentação de trabalhos (individuais ou em grupos);
Participação efetiva nas aulas (presença, pontualidade, atenção e principalmente contribuição significativa nos estudos realizados).

$$MS = 0.25P1 + 0.25P2 + 0.25P3 + 0.2TF + 0.05LE$$

Bibliografia básica

1. MENDES, Sueli. Métodos para Especificação de Sistemas. Editora Edgard Blücher Ltda. 1989.
TURNER, Kenneth. Formal Description Techniques. North Holland. 1989.

Bibliografia complementar

1. WOODCOCK, J. & LOOMES, M. Software Engineering Mathematics. Pitman, 1988.
PETERSON, J.L. Petri Net Theory and The Modelling of Systems. Prentice-Hall International, 1981.
BJORNER, D. & JONES, C.B. Formal Specification and Software Development. Prentice Hall International, Englewood Cliffs, NJ, 1982.
MOURA, A.V. Especificações em Z - Uma Introdução. Editora UNICAMP, ISBN 85-268-0575-4, 2002.
BAKKER, Dik. Formal and computational aspects of functional grammar and language typology, 1994.
MILNE, George J. Formal specification and verification of digital systems, 1994.
SHEPPARD, Deri. An introduction to formal specification with Z and VDM, 1995.
FLOYD, Robert W. The language of machines: an introduction to computability and formal language, 1994.
SUDKAMP, Thomas A. Languages and machines: an introduction to the theory of computer science, 1988.
KAIN, Richard Y. Automata theory : machines and languages, 1972.
Norris, James R. (1998). Markov chains Cambridge University Press [S.I.]
Artigos científicos internacionais indexados.

Plano de ensino

Curso: CCI-BAC - Bacharelado em Ciência da Computação

Turma: CCI122-07U - CIENCIA DA COMPUTAÇÃO FASE 07U

Disciplina: PIM0001 - PROCESSAMENTO DE IMAGENS

Período letivo: 2017/1

Carga horária: 72

Professor: 3321045 - GILMARIO BARBOSA DOS SANTOS

Ementa

1. Fundamentos. Operações globais e de vizinhança. Transformadas. Teorema da convolução. Realce. Restauração. Segmentação. Morfologia. Reconhecimento. Compressão. Aplicações.

Objetivo geral

1. Apresentar os principais conceitos envolvidos na aquisição, processamento e análise de imagens digitais. Introduzir os conceitos fundamentais das técnicas de processamento e codificação de imagem. Preparar os alunos para o uso de desenvolvimento de sistemas de processamento e análise de imagens.

Objetivo específico

1. - Compreender os principais métodos de forma esquemática (em alto nível) e matemática para imagens digitais binárias e em níveis de cinza.
- Implementar as técnicas mais simples e protótipos completos de aplicações em uma determinada área da ciência, selecionando informações importantes registradas em imagens digitais de forma semiautomática ou totalmente independente de interferência humana.
- Analisar diversos problemas de visão computacional e propor soluções aparentemente complexas em tempo mínimo, usando as ferramentas apresentadas.
- Capacitar os alunos com embasamento teórico para trabalhos de pesquisa na área de processamento de imagens (grupo LAPIS).

Conteúdo programático

1. Apresentação da Disciplina:
- Contexto na grade do BCC
- Metodologia e avaliação
- Ambientes de programação a serem utilizados
- Bibliografia
2. Introdução ao Python: bibliotecas de PIM
3. Introdução ao conteúdo:
- Histórico
- Aplicações
- Percepção Visual e Formação de Imagens
- Amostragem e Quantização
4. Introdução ao conteúdo:
- Histórico
- Aplicações
- Percepção Visual e Formação de Imagens
- Amostragem e Quantização
5. Fundamentos de imagens digitais:
- Conceitos relacionados com imagem digital
- Fundamentos matemáticos
- Relacionamentos básicos entre pixels
-- Vizinhança
-- Conectividade
-- Adjacência
-- Caminho
-- Medidas de Distância
-- Componentes Conexos
6. Realce
- Transformações do Histograma
- Filtragem no Domínio Espaço
-- Convolução
-- Supressão de Ruído
-- Realce de Detalhes
- Filtragem no Domínio Frequência
-- Filtros Passa-Baixas

Plano de ensino

-- Filtro Passa-Altas
7. Restauração: - Técnicas de Restauração e Degradação -- Filtro de Wiener -- Interpolação de Níveis de Cinza
8. Segmentação: -- Detecção de Descontinuidades -- Detecção de Bordas -- Limiarização --- Global --- Local - Segmentação Orientada a Regiões
9. Representação e Descrição: - Esquemas de Representação -- Código da Cadeia -- Aproximações Poligonais -- Assinaturas -- Esqueletização - Descritores -- Descritores de Fronteira -- Descritores de Fourier -- Descritores Regionais
10. Reconhecimento e Interpretação: - Padrões e Classes - Reconhecimento de Padrões em Imagens Compressão de Imagens.
11. Reconhecimento e Interpretação: - Padrões e Classes - Reconhecimento de Padrões em Imagens Compressão de Imagens.
12. Reconhecimento e Interpretação: - Padrões e Classes - Reconhecimento de Padrões em Imagens Compressão de Imagens.
13. Prova1
14. Prova2
15. Prova3
16. Exercícios
17. Orientação de trabalhos práticos
18. Avaliação de trabalhos práticos
19. Especificação de trabalhos práticos
20. Seminários
21. Resultados finais
22. Revisão para exame
23. Estudo de artigo técnico
24. Solução de exercícios
25. Outra atividade, especificada no complemento

Metodologia

1. Aulas expositivas acompanhadas de trabalhos práticos relacionados aos conteúdos apresentados em sala e de um projeto final. Aulas em laboratório também são desenvolvidas objetivando a implementação de algumas técnicas de processamento de imagens. Ao final da disciplina, os discentes terão estudado e implementado um subconjunto razoável de procedimentos fundamentais em Processamento de Imagens.

Vinte por cento da carga horária da disciplina poderá ser desenvolvido com auxílio do sistema Moodle (moodle2.joinville.udesc.br).

Plano de ensino

Sistema de avaliação

- O desempenho do aluno será avaliado com base no desenvolvimento das seguintes atividades e com os seguintes critérios:
 - participação ativa nas aulas e nos trabalhos;
 - realização de avaliações: apresentação de trabalhos individuais ou em grupo; avaliações individuais (provas).
 - a média semestral é calculada de acordo com a equação abaixo, onde PR corresponde à soma das notas de pelo menos duas provas (avaliações individuais escritas) e MT corresponde à média dos trabalhos (avaliações em grupo), os quais poderão ser ponderados de acordo com o grau de dificuldade dos mesmos).
Não está descartada a atribuição de nota (extra) como reconhecimento à participação destacada e de qualidade do(a) aluno(a) durante as atividades do curso:
$$MS = 0,6 * PR + 0,4 * MT$$
 - não é permitido o uso de aparelhos de comunicação (celulares e similares) durante as aulas.

Bibliografia básica

- Bibliografia Básica:
GONZALEZ, Rafael C.; WOODS, Richard E. Processamento de Imagens Digitais. São Paulo, Edgard Blücher, 2000. ISBN 8521202644.
PEDRINI, Hélio; SCHWARTZ, William R. Análise de Imagens Digitais - Princípios, Algoritmos e Aplicações. São Paulo, Thomson, 2008. ISBN 9788522105953.
O'ORMAN, Lawrence; SAMMON, Michael J.; SEUL, Michael. Practical Algorithms for Image Analysis: Description, Examples, Programs, and Projects. 2nd ed. New York: Cambridge University Press, 2009. ISBN 9780521884112.

Bibliografia complementar

- Bibliografia Complementar:
BAXES, Gregory A. Digital Image Processing: Principles and Applications. 1st ed. John Wiley & Sons, 1994. ISBN 0471009490.
BOVIK, A. C. (editor). Handbook of Image and Video Processing. 1st ed. Academic Press, 2000. ISBN 0121197905.
DOUGHERTY, E. R.; LOTUFO, R. A. Hands-on Morphological Image Processing, SPIE Press, 2003. ISBN 081944720X.
GOMES, Jonas; VELHO, Luiz. Computação Gráfica: Imagem. 2a ed. IMPA/SBM, 2002. ISBN 8524400889.
GONZALEZ, Rafael C.; WOODS, Richard E. Processamento Digital de Imagens. 3ª ed. Pearson, 2010. ISBN 9788576054016.
GONZALEZ, Rafael C.; WOODS, Richard E. Digital Image Processing. 3rd ed. Prentice Hall, 2007. ISBN 013168728X.
KIUSALAAS, Jaan. Numerical Methods in Engineering with Python. Cambridge University Press, 2005. ISBN 0521852870.
JÄHNE, B. Digital Image Processing. 5th ed. Springer, 2002. ISBN: 3540677542.
JAIN, Anil K. Fundamentals of Digital Image Processing. Prentice-Hall, Englewood Cliffs, US. Ed ed. 1988. ISBN 0133361659.
PARKER, J. R. Algorithms for Image Processing and Computer Vision. Bk&CD-Rom ed. John Wiley & Sons, 1996. ISBN 0471140562.
PRATT, W. K. Digital Image Processing: PIKS Inside. 3rd ed. John Wiley & Sons, 2001. ISBN: 0471374075.
RUSS, John C. The Image Processing Handbook, 5th ed. CRC, 2006. ISBN 0849372542.
Bibliografia Eletrônica:
BANON, G. J. F.; BARRERA, J. Bases da Morfologia Matemática para Análise de Imagens. 2a ed. 1998.
<http://www.vision.ime.usp.br/~jb/books/conteudo.pdf>
FALCÃO, A. Fundamentos de Processamento de Imagem Digital.
<http://www.dcc.unicamp.br/~cpg/materialdidatico/mo815/9802/curso/curso.html>
FISHER, R. B. CVonline: The Evolving, Distributed, Non-Proprietary, On-Line Compendium of Computer Vision.
<http://homepages.inf.ed.ac.uk/rbf/CVonline/>

Plano de ensino

JORDÁN, R.; LOTUFO, R. A. Digital Image Processing (DIP) with Khoros 2. Version 1.0.2, 1997.

<http://www.dca.fee.unicamp.br/dipcourse>

PRICE, K. Annotated Computer Vision Bibliography: Table of Contents. 2006.

<http://iris.usc.edu/VisionNotes/bibliography/contents.html>

LOTUFO, R. A.; MACHADO, R. C. pymorph Morphology Toolbox - Morphological image processing tools for Python. Version 0.8, 2003. <http://www.mmorph.com/pymorph>

SILVA, A. G.; LOTUFO, R. A. Image Processing Toolbox (ia636) with Python. Version 0.5, 2002.

<http://www.dca.fee.unicamp.br/ia636>

YOUNG, Ian T.; GERBRANDS, Jan J.; VAN VLIET, Lucas J. Fundamentals of Image Processing. Version 2.2, 1998. ISBN 9075691017.

<http://www.ph.tn.tudelft.nl/Courses/FIP/frames/fip.html> ; <http://www.ph.tn.tudelft.nl/DIPlib/docs/FIP.pdf>

Plano de ensino

Curso: CCI-BAC - Bacharelado em Ciência da Computação
Turma: CCI122-07U - CIENCIA DA COMPUTAÇÃO FASE 07U
Disciplina: TCC1003 - TRABALHO DE CONCLUSÃO DE CURSO I
Período letivo: 2017/1
Carga horária: 36
Professor: 3321045 - GILMARIO BARBOSA DOS SANTOS

Ementa
1. Desenvolvimento de Planejamento da Pesquisa do trabalho de conclusão de curso; definição de tema, escopo, objetivos, metodologia e levantamento bibliográfico.

Objetivo geral
1. Desenvolvimento do TCC-I de acordo com as orientações do professor orientador do acadêmico.

Objetivo específico
1. Desenvolvimento do TCC-I

Conteúdo programático
1. Apresentação das normas do TCC
2. Atendimento/Controle de dados
3. Atendimento/Controle de dados
4. Atendimento/Controle de dados
5. Atendimento/Controle de dados
6. Atendimento/Controle de dados
7. Atendimento/Controle de dados
8. Atendimento/Controle de dados
9. Atendimento/Controle de dados
10. Atendimento/Controle de dados
11. Atendimento/Controle de dados
12. Atendimento/Controle de dados
13. Atendimento/Controle de dados
14. Atendimento/Controle de dados
15. Atendimento/Controle de dados
16. Atendimento/Controle de dados
17. Atendimento/Controle de dados
18. Atendimento/Controle de dados
19. Atendimento/Controle de dados
20. Atendimento/Controle de dados

Plano de ensino

Metodologia
1. A metodologia de desenvolvimento é determinada pelo orientador do TCC.
Sistema de avaliação
1. Nota baseada na apresentação do TCC-I realizada pelo aluno diante de uma banca qualificada.
Bibliografia básica
1. Determinada pelo prof. Orientador do TCC em relação ao tema do trabalho.
Bibliografia complementar
1. Determinada pelo prof. Orientador do TCC em relação ao tema do trabalho.

Plano de ensino

Curso: CCI-BAC - Bacharelado em Ciência da Computação

Turma: CCI122-08U - CIENCIA DA COMPUTAÇÃO FASE 08U

Disciplina: ETI0001 - ÉTICA EM INFORMÁTICA

Período letivo: 2017/1

Carga horária: 36

Professor: 30175596883 - IVO FERNANDO DA COSTA

Ementa

1. Fundamentos da ética; O profissional de computação; A abrangência da ética em computação; A importância do raciocínio na tomada de decisões éticas; Problemas e pontos a ponderar; Códigos de ética profissionais; Ética profissional; Ética e regulamentação da profissão; Códigos de ética profissionais na área de computação.

Objetivo geral

1. Apresentar aos alunos discussões acerca da ética levando-os a refletirem sobre noções e princípios que fundamentam a vida moral e sobre a conduta profissional respeitando os princípios éticos em todas as esferas da vida em sociedade diante dos dilemas provocados pelo uso da tecnologia da informação.

Objetivo específico

1. a) Compreender o que é a ética e seu objeto de estudo.
b) Entender a diferença entre ética e moral.
c) Assimilar a relação entre a ética e os demais ramos do saber.
d) Adquirir uma visão geral da história da ética e suas principais correntes.
e) Reconhecer a necessidade de uma fundamentação da ética para determinar o reto atuar.
f) Conceituar valor como guia do reto atuar.
g) Entender o processo de tomada de decisão.
h) Reconhecer o papel da ética para a vida em sociedade em geral.
i) Entender a importância da ética para as empresas e organizações como sociedades particulares.
j) Reconhecer o campo da computação como uma profissão.
k) Fundamentar a origem dos códigos de ética profissional nos princípios da ética geral e social.
l) Resgatar um breve histórico dos computadores e da internet.
m) Perceber o computador como um agente humanitário.
n) Identificar os principais crimes virtuais em suas dimensões práticas e legais.

Conteúdo programático

1. Apresentação, planejamento semestral, cronograma pedagógico, grupos de estudo e sistema de avaliação.
2. 1. Objeto de estudo: o que é ética?
1.1. Significado.
1.2. Problemas morais e problemas éticos.
1.3. Distinção entre direito e ética.
1.4. O campo da ética.
1.5. Ética e filosofia.
1.6. Ética e outras ciências.
3. 2. Inventário histórico.
2.1. Ética e história.
2.2. Antiguidade greco-romana: ética da polis.
2.3. Idade Média: ética cristã.
2.4. Idade moderna: ética antropocêntrica.
2.5. Idade contemporânea: ética do homem concreto.
4. 2. Inventário histórico:
2.6. Ética da convicção e ética da responsabilidade.
2.7. Em busca de uma teoria ética unificadora.
5. 3. Inventário teórico: fundamentação ética
3.1. Fenomenologia da experiência moral: a essência da moral
3.2. Elementos da eticidade dos atos humanos
3.2.1. Consciência
3.2.2. Liberdade (ou determinismo?)
3.2.3. Responsabilidade
6. 3. Inventário teórico: fundamentação ética
3.3. Valores

Plano de ensino

3.4. Avaliação moral
7. 4. Realização da moral: A) Ética em sua vertente individual. 4.1. Princípios básicos. 4.2. Virtudes. 4.3. Critérios de tomada de decisão. 4.4. Processo de tomada de decisão.
8. 5. Realização da moral: B) Ética social. 5.1. Conceito, natureza e origem da sociedade. 5.2. Fundamentos da sociedade. 5.3. Princípios da ética social.
9. 6. Realização da moral: C) Ética profissional e o profissional de TI na organização 6.1. O profissional como membro de uma sociedade: empresa ou categoria 6.2. A empresa como instância concreta de sociedade 6.3. Dimensão ética das empresas/organizações 6.4. Empresa e demais organizações/empresas da sociedade
10. 6. Realização da moral: C) Ética profissional e o profissional de TI na organização 6.5. Finalidade da empresa 6.6. Legitimidade dos benefícios 6.7. O campo da computação como uma profissão 6.8. Que oportunidades tem a ética num mundo globalizado de consumidores?
11. 7. Ética aplicada ao mundo da informática. 7.1. Uma breve história dos computadores e da internet. 7.2. Ética na computação, um tipo singular de ética? 7.3. Computador como agente humanitário. 7.4. Objeto da ética em informática. 7.5. Campo da computação como uma profissão. 7.6. Códigos de ética relacionados à computação. 7.7. Robótica e IA. 7.8. Propriedade intelectual. 7.9. Crimes virtuais. 7.10. Ética, informática e legalidade. 7.11. Ética na Internet. 7.12. Pragas virtuais.

Metodologia

1. Aulas expositivo-dialogadas, leituras e análises de textos, revistas e livros, trabalhos em grupo e individuais, apresentações de seminários e discussões sobre os temas estudados.

Sistema de avaliação

1. Os alunos serão avaliados mediante a atribuição de notas individuais oriundas de: Apresentações de seminários, atividades extraclasse, prova escrita.
Os critérios de avaliação das atividades são:
 - Assiduidade e pontualidade;
 - participação em sala de aula e nos trabalhos de grupos;
 - leitura e discussão dos textos propostos;
 - produção escrita.

As atividades de ensino realizadas em sala de aula não suprem a leitura dos textos indicados para o correto desenvolvimento do processo de ensino-aprendizagem.

Os trabalhos e atividades deverão ser entregues pontualmente na data assignada. Trabalhos entregues com atraso terão a nota descontada em 0,5 ponto por data de atraso até a semana subsequente em que já não serão mais aceitos. Faltas em datas de realização ou entrega de avaliações devem ser justificadas na secretaria acadêmica e solicitada a segunda chamada de avaliação.

Na avaliação da apresentação de seminário serão levados em conta os seguintes itens:

- Apresentação em Power Point ou similar.
- Pertinência do conteúdo apresentado.
- Capacidade de análise e síntese.
- Apresentação de imagens e vídeos relacionados ao tema.
- Dinâmicas de grupo.
- Questões para debate.
- Resumo do conteúdo a ser distribuído para a turma.
- Quem faltar sem justificativa no dia da apresentação de outra equipe terá 0,5 pontos descontados da nota de seu seminário.

Plano de ensino

PESOS DAS AVALIAÇÕES:

[(P1+P2) (40%)] + [(Atividades Extraclasse) (15%)] + [(Debates Coordenados) (45%)]

EXAME: 05/07

Bibliografia básica

1. BARGER, Robert N. Ética na computação: uma abordagem baseada em casos. Rio de Janeiro: LTC, 2011.
BOWYER, K. W. Ethics and Computing: living responsibly in a computerized world. IEEE Computer Society Press 1996.
COTRIM, GILBERTO. Fundamentos de filosofia. São Paulo: Saraiva, 2010.
DE CICCIO, Claudio; GONZAGA, Alvaro de Azevedo. Teoria geral do Estado e ciência política. 2.ed. rev. atual. e ampl. São Paulo: Revista dos Tribunais, 2009.
EDGAR, Stacey L. Morality and machines: perspectives on computer ethics. 2nd ed. Massachusetts: Jones and Bartlett, c2003.
FORESTER, T. e MORRISON, P. Computer Ethics. The MIT Press, 1993.
MARCONDES, Danilo. Textos básicos de ética: de Platão à Foucault. Rio de Janeiro: J. Zahar, 2007.
MASIERO, Paulo Cesar. Ética em computação. Edusp, 2000.
SÁNCHEZ VÁZQUEZ, Adolfo. Ética. Rio de Janeiro: Civilização Brasileira, 2008.

Bibliografia complementar

1. ARRUDA, Maria Cecília Coutinho de; WHITAKER, Maria do Carmo; RAMOS, José Maria Rodriguez. Fundamentos de ética empresarial e econômica. 3. ed. São Paulo: Atlas, 2005.
FORESTER, Tom; MORRISON, Perry. Computer ethics: cautionary tales and ethical dilemmas in computing. 2nd. ed. Massachusetts: MIT Press, 2001.
SROUR, Robert Henry. Casos de ética empresarial: chaves para entender e decidir. Rio de Janeiro: Campus, c2011.
SROUR, Robert Henry. Ética empresarial: o ciclo virtuoso dos negócios. 3. ed. rev. e atual. Rio de Janeiro: Elsevier, 2008.
VIDAL, Marciano. Ética teológica: conceitos fundamentais. Petrópolis: Vozes, 1999.

Plano de ensino

Curso: CCI-BAC - Bacharelado em Ciência da Computação

Turma: CCI122-08U - CIENCIA DA COMPUTAÇÃO FASE 08U

Disciplina: OACO001 - ANIMAÇÃO POR COMPUTADOR

Período letivo: 2017/1

Carga horária: 72

Professor: 3569829 - ROGERIO EDUARDO DA SILVA

Ementa

1. Animação Convencional. Etapas da animação computacional. Formas de animação. Animação hierárquica. Corpos rígidos e flexíveis. Controle de grupos (partículas, bandos e comportamento autônomo). Fenômenos naturais. Animações interativas (jogos computacionais).

Objetivo geral

1. Introduzir conceitos de animação tradicional vs. computacional. Apresentar as principais técnicas de animação computacional. Permitir ao aluno projetar e construir sistemas animados com e sem interação do usuário

Objetivo específico

1. Especificamente, após o curso o estudante deverá ser capaz de:
 - . CARACTERIZAR o conceito de animação computacional e seus principais tipos
 - . INTRODUIZIR os conceitos pertinentes ao projeto de animações computacionais
 - . APRESENTAR os princípios da animação
 - . DISCUTIR as principais técnicas computacionais de animação auxiliada por computador
 - . APLICAR os conceitos aprendidos no projeto e implementação de um sistema de animação.

Conteúdo programático

1. Apresentação da Disciplina
Apresentação do Plano de Aulas, Conteúdo Programático, Critérios de Avaliação
2. Conceitos Introdutórios
Definições básicas: quadro, frequência, persistência visual
O que é animação?
O que é animação computacional?
3. Revisão C++
Conceitos de Orientação a Objetos
Classes e Objetos
Herança
Polimorfismo de Método e de Operador
Ponteiros
Documentação Automática com a Ferramenta Doxygen
UML
4. Revisão de OpenGL
Conceitos introdutórios
Câmera Virtual, Projeções, Viewport
Rendering, Cores, Texturas
Iluminação
Depth buffer, Stencil buffer, Transparências, Reflexos
Modelagem e Primitivas Gráficas
Importação de objetos do Blender3D
5. Revisão de Conceitos Matemáticos
Vetores e Matrizes
Interpolações
Geometria analítica: plano, esfera, cubo, cilindro, elipse, circunferência
6. Ferramentas de Autoria
Introdução a ferramentas de autoria em animação: Blender3D
7. Princípios da Animação
12 Princípios da Animação
Pose-to-Pose/Straight Ahead, Ease-in Ease-out, Staging, Solid Drawings, Arcs,
Follow Through/Overlapping Action, Secondary Action, Anticipation, Exaggeration
Timing/Spacing, Squash/Stretch
8. Processo de Animação
Pré-produção: concepção, roteiro, arte, storyboard, modelagem, rigging, cenários
Produção: áudio/trilha sonora, geração de frames
Pós-produção: composição, legendas, vinhetas

Plano de ensino

9. Keyframing Processo de Animação por Quadros-chave Pose-to-Pose Animation Implementação
10. Modelos Hierárquicos Coordenadas Globais vs Locais Relação Pai-Filho entre Objetos Modelos Articulados Tipos de Articulações: hinge, pivot, ball-and-socket Cinemática Direta vs Inversa
11. Modelos Baseados em Física Simulação de Forças Modelos de Corpos Rígidos Modelos de Corpos Flexíveis
12. Sistema de Partículas Animação de grupos Objetos Amorfos e Fenômenos Naturais
13. Steering Behaviors Conceito de Boids Animação de Comportamento de Indivíduos em um Grupo Flee, Seek, Evade, Pursue, e outros Animação de Comportamento de Grupos Coesão, Acoplamento e Alinhamento
14. Projeto Final Implementação Prática do Projeto Final Storyboard e Estória Combinação de Múltiplas Técnicas de Animação Criação de um Filme de Animação completo
15. Animação Straight-ahead Fundamentos da animação stop-motion
16. Técnicas de Modelagem 3D B-Rep Operações e Modificadores Poligonais CSG
17. Rendering Blender Render vs Blender Cycles Modelos de Iluminação Câmera Virtual
18. Atores Digitais Postura Corporal Animação Facial - FACS Sincronização Labial
19. Prova Escrita Avaliação teórica sobre os conceitos abordados durante o semestre
20. Atividades Extra-Classe Leitura de material didático de apoio Vídeo-aulas (tutoriais) com técnicas específicas
21. EPI Apresentação de Exercício Prático Individual
22. Semana Acadêmica Conjunto de palestras com temas relacionados à disciplina

Metodologia

1. Conteúdos expostos pelo professor com participação colaborativa dos estudantes, acompanhados pela proposição de exercícios práticos associados. Implementação de problemas clássicos da área.
--

Sistema de avaliação

1. A média semestral será calculada de acordo com os seguintes critérios: - 6 Exercícios Práticos Individuais (EPI): 10% cada - Prova Escrita: 15% - Projeto Final Semestral: 25%

Plano de ensino

Exame

Caso o discente não obtenha média M igual ou superior a 7.0, um exame escrito será aplicado objetivando aferir o conhecimento teórico do estudante.

Do desempenho da disciplina e do professor:

Os estudantes terão, igualmente, a oportunidade de efetuar, ao andamento da disciplina uma avaliação mais completa do desempenho do professor e da disciplina. O formulário para esta atividade será preparado para a ocasião ou fornecido pelo coordenador do curso.

Bibliografia básica

1. PARENT, Rick, "Computer Animation - Algorithms and Techniques", Morgan Kaufmann, 2002.

BYL, P. B. "Programming Believable Characters for Computer Games", Charles River Media, 2004.

ROBERTS, S. "Character Animation. 2D Skills for Better 3D", Focal Press, 2007

Bibliografia complementar

1. WHITAKER, H. & HALAS, J., "Timing for Animation", Focal Press, 2009.

AZEVEDO, E. (Coord.), "Desenvolvimento de Jogos 3D e Aplicações em Realidade Virtual", Editora Campus, 2005.

DELOURA, M., "Game Programming Gems (1 a 6)", Charles River Media, 2000 a 2006

Plano de ensino

Curso: CCI-BAC - Bacharelado em Ciência da Computação

Turma: CCI122-08U - CIENCIA DA COMPUTAÇÃO FASE 08U

Disciplina: OCEV001 - COMPUTAÇÃO EVOLUCIONÁRIA

Período letivo: 2017/1

Carga horária: 72

Professor: 3605655 - RAFAEL STUBS PARPINELLI

Ementa

1. Introdução à Computação Evolucionária (CE): comparação de paradigmas, histórico e métodos de otimização. Fundamentos teóricos e tópicos avançados de Algoritmos Genéticos. Estudo de aplicações de Algoritmos Genéticos. Tópicos avançados em Computação Evolucionária: Otimização por Colônias de Formigas (Ant Colony Optimization) e Otimização por Enxame de Partículas (Particle Swarm Optimization).

Objetivo geral

1. Fornecer ao aluno um conhecimento teórico e operacional, das técnicas de computação evolucionária mais comuns. Espera-se que, após a disciplina, o aluno esteja capacitado para a leitura e entendimento de aplicações relacionadas à computação evolucionária e que seja, ele mesmo, capaz de fazer aplicações das técnicas aprendidas. As aplicações apresentadas terão como objetivo ilustrar as técnicas, conceitos e aspectos importantes para a prática.

Objetivo específico

1. SITUAR a Computação Evolucionária dentro do universo da Computação Natural;

CONCEITUAR Computação Evolucionária e seus principais paradigmas;

APRESENTAR as equivalências e diferenças entre os paradigmas apresentados;

MODELAR problemas de otimização para APLICAR as heurísticas citadas.

Conteúdo programático

1. Introdução e Conceitos Básicos:
 - . Computação Evolucionária dentro da Computação Natural;
 - . Ciclo básico de um Algoritmo Evolucionário;
 - . Vantagens e desvantagens da CE.
2. Princípios de Otimização:
 - . Tipos de problemas;
 - . Métodos de solução de problemas;
 - . Quando usar CE para otimização;
 - . Diferenças entre Algoritmos Genéticos (AG); e outros métodos de otimização;
 - . Características de problemas 'difíceis'.
3. Fundamentos Teóricos:
 - . Terminologia;
 - . AG canônico;
 - . Definição formal;
 - . Critérios de término;
 - . Genótipo x Fenótipo;
 - . Princípios de codificação de problemas;
 - . Função objetivo e função de fitness;
 - . Tratamento de restrições e aplicação de penalidades;
 - . Métodos de seleção
 - . Operadores genéticos: recombinação, mutação e inversão;
 - . Operadores especiais para problemas combinatoriais;
 - . Critérios de convergência;
 - . Exploração do espaço de busca e diversidade genética;
 - . Diversidade genética e pressão seletiva;
 - . Convergência prematura e chegada lenta;
 - . Escalonamento de fitness;
 - . Definições: Elitismo, Generation Gap, Epistasia, problemas enganadores, nichos e espécies, fator de crowding.
4. . Programação Genética:
 - . Fundamentos teóricos;
 - . Estudo de aplicações.
5. . Tópicos avançados em Computação Evolucionária.

Plano de ensino

Metodologia

1. Aulas expositivas dialogadas, com resolução exercícios e apresentação de trabalhos. Até 20% da carga horária poderá ser desenvolvida à distância.

Sistema de avaliação

1. A qualidade do desempenho do aluno será avaliada com base no desenvolvimento das seguintes atividades e com os seguintes critérios:

- a) participação ativa nas aulas e na solução dos exercícios práticos em sala de aula e em laboratório.
- b) elaboração e apresentação de trabalho prático de laboratório.
- c) avaliações individuais (provas)
- d) avaliações individuais (seminários)

A nota final será composta com o seguinte critério:

$PROVA1 \times 0,30 + PROVA2 \times 0,30 + TRABALHOS \times 0,25 + SEMINÁRIO \times 0,15$

Bibliografia básica

1. Goldberg, D.E., Genetic Algorithms in Search, Optimization and Machine Learning. Reading: Addison-Wesley, 1989.
2. Mitchell, M., An Introduction to Genetic Algorithms. Cambridge: MIT Press, 1996.
3. Koza, J.R., Genetic Programming: on the programming of computers by means of natural selection. Cambridge: MIT Press, 1992.

Bibliografia complementar

1. Bonabeau, E., Dorigo, M., Theraulaz, G., Swarm Intelligence: From Natural to Artificial Systems. Santa Fe Institute Studies on the Sciences of Complexity, Oxford University Press, 1999.
2. Kennedy, J., Eberhart, R. C., Swarm Intelligence. San Francisco, Morgan Kaufmann Publishers, 2001.
3. Outros artigos científicos.

Plano de ensino

Curso: CCI-BAC - Bacharelado em Ciência da Computação

Turma: CCI122-08U - CIENCIA DA COMPUTAÇÃO FASE 08U

Disciplina: ODAW001 - DESENVOLVIMENTO DE APLICAÇÕES NA WEB

Período letivo: 2017/1

Carga horária: 72

Professor: 3629953 - DEBORA CABRAL NAZARIO

Ementa

1. Desenvolvimento de aplicações orientado às necessidades do usuário. Estudo e utilização de tecnologias para Web: XML, XSL, XHTML, CSS, JavaScript, Java para Web (Servlets, JSP), PHP.

Objetivo geral

1. Capacitar o aluno no conhecimento das tecnologias e desenvolvimento de uma aplicação/ambiente protótipo para Web.

Objetivo específico

1. - Conceituar os elementos básicos do ambiente web;
- Conhecer tecnologias para desenvolvimento na Web;
- Discutir os aspectos de design/usabilidade de um Web Site;
- Discutir aspectos de segurança em aplicações na web;
- Desenvolver um ambiente/aplicação web protótipo.

Conteúdo programático

1. 1. Introdução: Conceitos básicos relacionados com a Internet.
2. 2. Estudo de tecnologias para Web: HTML, XML, XSL, XHTML, CSS, JavaScript, JSP, ASP, PHP.
3. 3. Programação para Internet (estático)
Criação de sites com HTML
Criação de formulários
Scripts para validação de campos (Javascript)
Folhas de estilo (CSS)
4. 4. Programação para Internet (dinâmico)
Como publicar seu site na Internet.
Desenvolvendo sites dinâmicos
Acesso a banco de dados
Seções e cookies
5. 5. Metodologia de Projeto para Web, aspectos de acessibilidade, usabilidade, Design Web.
6. 6. Segurança na Web
7. 7. Desenvolvimento de uma aplicação para Web.
Elaboração de um projeto
Desenvolvimento do protótipo

Metodologia

1. Aulas expositivas, trabalhos individuais ou em grupo, atividades práticas em laboratório, prova.
O conteúdo da disciplina poderá ser ministrado na modalidade de ensino a distância em até 20% do total de sua Carga Horária (MEC PORTARIA Nº 4.059, DE 10 DE DEZEMBRO DE 2004 publicado no DOU de 13/12/2004, Seção 1, p. 34).

Sistema de avaliação

1. Exercícios em Laboratório;
Prova escrita/prática;
Apresentação de trabalhos, individuais ou em grupos;
Projeto e implementação de uma aplicação web;
Participação efetiva nas aulas.

Média Semestral = $(P*2 + T1*2 + T2*3 + E*1)/8$, onde:

P = Prova

T1 = Trabalho 1: escrito + apresentação

T2 = Trabalho 2: projeto + aplicação

E = Exercícios resolvidos em laboratório, só poderão entregar os alunos presentes em aula, sem direito à recuperação da nota. A data não será marcada com antecedência. Ou ainda atividades encaminhadas e entregues através do Moodle.

Plano de ensino

Bibliografia básica

1. DEITEL, H.M.; Deitel, P.J.; Nieto, T.R. Internet & World Wide Web Como Programar, 2a Edição. Porto Alegre: Bookman, 2003.
ANSELMO, Fernando. PHP 4 e MySQL: maior, melhor e totalmente sem cortes. Florianópolis: Visual Books, 2002.
MARCON, Antonio Marcos; NEVES, Denise. Aplicações e banco de dados para internet. 2.ed. São Paulo: Livros Erica, 2000.

Bibliografia complementar

1. GOODMAN, Danny. JavaScript: a bíblia. Rio de Janeiro: Campus, 2001.
FLANAGAN, David. JavaScript: o guia definitivo. 4. ed. Porto Alegre: Bookman, 2004.
CARVALHO, A. Desenvolvendo Sites Profissionais com HTML 4.0 & CSS 2, Editora Book Express.
Budi Kurniawan. Java para a Web com Servlets, JSP e EJB. Rio de Janeiro: Ciência Moderna, c2002.

Plano de ensino

Curso: CCI-BAC - Bacharelado em Ciência da Computação

Turma: CCI122-08U - CIENCIA DA COMPUTAÇÃO FASE 08U

Disciplina: OMIC001 - MICROPROCESSADORES

Período letivo: 2017/1

Carga horária: 72

Professor: 9672753 - LEANDRO ISRAEL PINTO

Ementa

1. Análise funcional e operacional de microprocessadores. Interligação de memórias. Programação assembly. Interfaces de comunicação. Circuitos periféricos.

Objetivo geral

1. Explorar a arquitetura de programação de microprocessadores e microcontroladores, com ênfase em conceitos tecnologicamente independentes.

Objetivo específico

1. Introduzir os principais conceitos de sistemas digitais;
Identificar as diferenças entre microprocessador e microcontrolador;
Explorar diferentes modelos de microcontroladores, seus diferentes recursos disponíveis e aplicações;
Explorar o conjunto de instruções de um microcontrolador;
Desenvolver aplicações utilizando o conjunto de instruções estudado;
Desenvolver aplicações utilizando compiladores de linguagens de alto nível;
Explorar técnicas de modelagem para o desenvolvimento de sistemas embarcados.

Conteúdo programático

1. 1. Introdução aos Conceitos Básicos
1.1 Evolução dos Computadores
1.2 Sistemas Digitais
1.3 Microprocessador e Microcontrolador
1.4 Microcontroladores e Periféricos
1.5 Arquiteturas dos microcontroladores PIC (Microchip) e AVR (Atmel)
2. 2. Programação Assembly
2.1 Conjunto de Instruções
2.2 O Montador Assembly e o Formato Intel HEX
2.3 Gravando o Programa no Microcontrolador
2.4 Acessando Periféricos em Assembly
3. 3. Programação em Linguagem C
3.1 Diferenças entre Compiladores
3.2 Passos da Compilação
3.3 Acessando Periféricos em Linguagem C
4. 4. Modelagem para Sistemas Embarcados

Metodologia

1. A disciplina será trabalhada através da temática Teórico-prática, com exercícios e aulas expositivas dialogadas. As aulas serão seguidas de atividades práticas em laboratório. Os trabalhos serão apresentados durante o transcorrer da disciplina.

Sistema de avaliação

1. Do desempenho do aluno:
A qualidade do desempenho do aluno será avaliada com base no desenvolvimento das seguintes atividades e com os seguintes critérios:
a) Trabalhos práticos (30%+20%=50 %)
b) Exercícios (15 %)
c) Trabalho Final (35 %)

Do desempenho da disciplina e do professor:
Os estudantes terão, igualmente, a oportunidade de fazer uma avaliação mais completa do desempenho do professor e da disciplina através do sistema de avaliação.

Plano de ensino

Bibliografia básica

1. SOUZA, David Jose de. Desbravando o PIC: ampliado e atualizado para PIC16F628A. 12. ed. São Paulo: Livros Erica, c2003. 268 p. ISBN 9788571948679

PEREIRA, Fábio. Microcontroladores PIC: programação em C. 7. ed. São Paulo: Érica, c2003. 358 p. ISBN 9788571949355 (broch.).

PEREIRA, Fabio. Microcontrolador PIC18 detalhado: hardware e software. São Paulo: Érica, 2010. 304 p. ISBN 9788536502717 (broch.).

Bibliografia complementar

1. PEREIRA, Fábio. Microcontroladores PIC: técnicas avançadas. 2. ed. São Paulo: Érica, 2002. 358 p. ISBN 8571947279 (broch.).

Plano de ensino

Curso: CCI-BAC - Bacharelado em Ciência da Computação

Turma: CCI122-08U - CIENCIA DA COMPUTAÇÃO FASE 08U

Disciplina: OPRP001 - PROGRAMAÇÃO PARALELA

Período letivo: 2017/1

Carga horária: 72

Professor: 3746810 - MAURICIO ARONNE PILLON

Ementa

1. Modelos de computação paralela. Expressão e extração do paralelismo. Sincronização e comunicação: métodos e primitivas. Programação concorrente e distribuída: linguagens e algoritmos. Problemas clássicos de programação paralela. Princípios de implementação.

Objetivo geral

1. Capacitar o aluno a compreender os conceitos sobre programação paralela bem como os paradigmas e princípios de implementação envolvidos no desenvolvimento de tais sistemas.

Objetivo específico

1. Compreender os conceitos introdutórios de programação paralela;
Compreender as aplicações práticas vinculadas à programação paralela;
Aplicar algoritmos e estratégias de particionamento e divisão para conquistar em problemas clássicos;
Desenvolver protótipos de soluções para problemas clássicos da área;

Conteúdo programático

1. 1. Apresentação da disciplina
2. 2. Conceitos básicos de Programação Paralela
3. 3. Arquiteturas paralelas
4. 4. Sincronização e comunicação
5. 5. Programação concorrente e distribuída
6. 6. Problemas clássicos de Programação Paralela
7. 7. Princípios de Programação Paralela
8. 8. Projeto de algoritmos paralelos
9. 9. Introdução a avaliação de desempenho
10. 10. Implementação de algoritmos paralelos
11. Tx: Avaliação escrita do tipo Trabalho (x = 1 à 10)
12. Ex: Avaliação escrita do tipo Exercício (x = 1 à 10)
13. Ax: Aula especializada - revisão, palestra, minicurso, eventos, etc - (x = 1 à 5)
14. Px: Avaliação escrita do tipo Prova (x= 1 à 3)

Metodologia

1. Aulas Expositivo-Dialogadas; Uso de Laboratório; Desenvolvimento de atividades individuais e em grupo.

Sistema de avaliação

1. - Provas teóricas e/ou práticas;
- Participação em aula;
- Implementações e relatórios apresentando uma solução envolvendo técnicas de construção de sistemas paralelos;
- Trabalhos em grupos objetivando o desenvolvimento de soluções para problemas sugeridos.

Nota Final = $A1 * 0,30 + A2 * 0,35 + A3 * 0,35$

A1: Avaliação 01 - Prova escrita.

A2: Avaliação 02 - Implementação e demonstração de aplicativos elaborados ao longo da disciplina

A3: Avaliação 03 - Implementação e demonstração de aplicativos elaborados ao longo da disciplina.

Plano de ensino

Cada avaliação poderá ser decomposta em um subconjunto de avaliações com pesos ponderados de acordo com sua complexidade. Complementarmente, as avaliações podem solicitar a elaboração de relatórios técnicos e apresentações.

Bibliografia básica

1. DE ROSE, César A. F.; NAVAUX, Philippe O. A. Arquiteturas Paralelas. Porto Alegre: Sagra-Luzzato, 2003.
2. FOSTER, Ian. Designing and Building Parallel Programs: Concepts and Tools for Parallel Software Engineering. Editora Addison-Wesley, 1995
3. STALLINGS, William. Operating Systems - Internals and Design Principles. Prentice-Hall. 1997. 3a. Edição.

Bibliografia complementar

1. 4. PITANGA, Marcos. Construindo Supercomputadores com Linux. Brasport, 2002.
5. ROOSTA, Seyed H. Parallel Processing and Parallel Algorithms: Theory and Computation. New York: Springer-Verlag, 2000.
6. TOSCANI, Simão Sirineu. et al. Sistemas Operacionais e Programação Concorrente. Porto Alegre: Sagra-Luzzato, 2003.
7. QUINN, Michael J. Parallel Programming in C with MPI and OpenMP. McGRAW HILL, 2003.
8. WESLEY, Petersen. Introduction to Parallel Computing - A practical guide with examples in C. Oxford University Press, 2004.
9. WILKINSON, Barry and Allen, Michael. Parallel Programming. Techniques and Applications Using Networked Workstations and Parallel Computers. Prentice Hall. 1999. 1a. Edição.

Plano de ensino

Curso: CCI-BAC - Bacharelado em Ciência da Computação

Turma: CCI122-08U - CIENCIA DA COMPUTAÇÃO FASE 08U

Disciplina: OSIM001 - SISTEMAS MULTIAGENTES

Período letivo: 2017/1

Carga horária: 72

Professor: 3092798 - CLAUDIO CESAR DE SA

Ementa

1. Motivação do paradigma. Agentes reativos e cognitivos. Teoria e arquitetura de agentes. Sistema multi-agentes (SMA) reativo e cognitivo. Linguagens e protocolos de comunicação. Coordenação e negociação. Metodologias para desenvolvimento de SMAs. Ambientes de desenvolvimento.

Objetivo geral

1. Apresentar o conceito de inteligência artificial distribuída: desenvolvimento de agentes e abordagens para coordenação de sistemas multiagentes, permitindo ao aluno ser capaz de modelar problemas de forma a modularizar sua solução de forma distribuída.

Objetivo específico

1. Motivação do paradigma. Agentes reativos e cognitivos. Teoria e arquitetura de agentes. Sistema multi-agentes (SMA) reativo e cognitivo. Linguagens e protocolos de comunicação. Coordenação e negociação. Metodologias para desenvolvimento de SMAs. Ambientes de desenvolvimento.

Conteúdo programático

1. Aula Laboratório -- Implementações -- listas de exercícios
2. 1. Conceitos básicos de IA, Agentes e Motivação
 - 1.2. O que são agentes?
 - 1.3. Para que servem sistemas multiagentes ?
 - 1.4. Arquiteturas de sistemas multiagentes
 - 1.5. Motivação
 - 1.6. Aplicações
3. 2 Introdução a Agentes Racionais
 - 2.1 Tipos de agentes: reativos e cognitivos
 - 2.2 Arquiteturas de agentes reativos
 - 2.2.1 Agentes Reativos
 - 2.2.2 Agentes baseado em disparos
 - 2.2.3 Agentes baseado em objetivos
 - 2.2.4 Agentes baseado em utilidade
 - 2.2.5 Agentes instantaneos (Anytime)
 - 2.3 Arquiteturas de agentes cognitivos
 - 2.3.1 Agentes BDI
 - 2.3.2 Agentes Afetivos EBDI
4. 3 Sistemas multiagentes
 - 3.1 Arquiteturas de sistemas multiagentes: distribuído e centralizado
 - 3.1.2 Quadro-negro (blackboard)
 - 3.1.3 Negociação
 - 3.1.4 Leilão
 - 3.1.5 Votação
 - 3.1.6 Abordagens inspiradas na natureza
5. 4 Introdução a Programação de Agentes
 - 4.1 JADE, JADEX
 - 4.2 NetLogo
 - 4.3 Sistemas dedutivos com PICAT
6. Seminários de SMA
7. Avaliação escrita (prova) do conteúdo corrente
8. Exame Final
9. Dúvidas do alunos
10. Defesa e apresentação do projeto final dos alunos

Plano de ensino

11. Teoria de jogos aplicada a SMAs
12. Coordenação de SMAs
13. Abordagens para Planejamento com SMAs
14. Listas de exercício em laboratório
15. Vídeo-aulas de NETLOGO + Laboratório
16. Apresentação da disciplina Dinâmica de aulas Avaliações da disciplina para aprovação do aluno nesta. Datas dos eventos
17. Revisão de conteúdo de IA ou de SMA (aula anterior)

Metodologia

1. Conteúdos expostos pelo professor com participação colaborativa dos estudantes, acompanhados pela proposição de exercícios práticos associados. Implementação de problemas clássicos da área. O conteúdo da disciplina poderá ser ministrado na modalidade de ensino a distância em até 20% do total de sua Carga Horária (MEC PORTARIA Nº 4.059, DE 10 DE DEZEMBRO DE 2004 publicado no DOU de 13/12/2004, Seção 1, p. 34)..
--

Sistema de avaliação

1. Ao longo do semestre, serão realizadas as seguintes avaliações: - 3 implementações práticas (IP1, IP2 e IP3) sobre diferentes arquiteturas de agentes e sistemas multiagentes vistos em aula. Estas problemas são debatidos e modelados em aula; - Apresentação de um seminário (S) sobre um artigo de SMA: S - 02 avaliações escritas: P1 e P2 - Projeto final em formato de artigo e implementação: Pf A média semestral, MS, portanto, será obtida por meio da seguinte fórmula: $MS = (IP1 + IP2 + IP3) \times 0.2 + S \times 0.2 + (P1 + P2) \times 0.3 + 0.3 \times Pf$ Caso o discente não obtenha média M igual ou superior a 7,0, um exame escrito será aplicado objetivando aferir o conhecimento teórico do estudante.

Bibliografia básica

1. ALVARES, L. O., SICHMAN, J. Introdução aos Sistemas Multiagentes, Anais do EINE - Escola de Informática do Nordeste, Sociedade Brasileira de Computação - SBC, Brasil, 1997. FERBER, J. Multi-Agent Systems: An Introduction to Distributed Artificial Intelligence. Harlow, England, Addison-Wesley, 1999. WOOLDRIDGE, M.. An introduction to MultiAgent Systems, John Wiley, 2001
--

Bibliografia complementar

1. Yoav Shoham, Kevin Leyton-Brown. Multiagent Systems - Algorithmic, Game-Theoretic, and Logical Foundations. Cambridge University Press, 2009. O'HARE, G.; JENNINGS, N. (Editors) Foundations of distributed artificial intelligence, New York, NY: John Wiley, 1996. BORDINI, R.; DASTANI, M.; DIX, J; SEGHTROUCHNI, A. (Editors). Multi-Agent Programming : Languages, Platforms and Applications (Multiagent Systems, Artificial Societies, and Simulated Organizations), Hardcover/Springer Verlag, 2005. COELHO, H.. IA em 25 Lições, Lisboa, Fundação Calouste Gulbenkian - Serviço Educação, 1995. RUSSEL, S.; NORVIG, P. Inteligência Artificial. Tradução da 2a. Edição. Elsevier, Rio de Janeiro. 2004. WEISS, G. Multiagent Systems: A Modern Approach to Distributed Artificial Intelligence. London, MIT Press, 2001.

Plano de ensino

Curso: CCI-BAC - Bacharelado em Ciência da Computação

Turma: CCI122-08U - CIENCIA DA COMPUTAÇÃO FASE 08U

Disciplina: OSRC001 - SEGURANÇA EM REDES DE COMPUTADORES

Período letivo: 2017/1

Carga horária: 72

Professor: 3375552 - CHARLES CHRISTIAN MIERS

Ementa

1. Estudo dos desafios referentes à segurança em ambientes computacionais. Estudo de soluções para segurança em software, sistema operacional e rede de computadores, assim como estudos dos mecanismos de proteção, políticas e cultura de segurança, ações necessárias frente a ataques. Auditoria de Sistemas. Aspectos especiais: vírus, fraudes, criptografia, acesso não autorizado.

Objetivo geral

1. Apresentar as noções fundamentais das principais metodologias de segurança da informação e do desenvolvimento de sistemas com maior confiabilidade, com abordagem teórica prática, visando ensinar para o aluno as fases de análise, projeto e implementação de sistemas de segurança, assim como a identificação e análise de ameaças e vulnerabilidades.

Objetivo específico

1. - Capacitar o aluno a entender o que é segurança da informação e sua composição;
- Diferenciar aspectos de Segurança da Informação técnicos e não técnicos;
- Caracterizar e utilizar os mecanismos e ferramentas básicos de segurança em redes de computadores;
- Compreender a aplicação e organização de segurança da informação em empresas e redes de computadores; e
- Conhecer e diferenciar as principais normas de segurança da informação.

Conteúdo programático

1. 0. Plano de ensino e método de avaliação
Explicação da organização e conteúdo da disciplina
Explicação do método de avaliação da disciplina
Explicação sobre os objetivos, organização e avaliação da AP1, AP2 e TE1
2. 1. Fundamentos
1.1 Introdução à segurança da informação
1.2 Riscos envolvendo informações
1.3 Principais ameaças e vulnerabilidades
3. 1. Fundamentos
Filme / documentário Takedown
Dinâmica sobre engenharia social
4. 2. Níveis de Segurança
2.1 Classificação da segurança
5. 2. Níveis de Segurança
2.2 Normas de segurança da informação
6. 3. Análise de Riscos
3.1 Método para análise de risco segundo a BS7799
7. 3. Análise de Riscos
3.2 Pré-análise de riscos
8. 4. Criptografia
4.1 Conceitos básicos e história
9. 4. Criptografia
4.2 Criptografia simétrica SDES / DES
10. 4. Criptografia
4.3 Criptografia simétrica SAES / AES
11. 4. Criptografia
4.4 Criptografia assimétrica RSA e curvas elípticas
12. 4. Criptografia
4.5 Resumos, assinatura digital e autenticação
13. 4. Criptografia
4.6 Infraestrutura de chaves públicas
14. 5. Segurança em redes de computadores
5.1 Firewalls e proxies

Plano de ensino

15. 5. Segurança em redes de computadores (firewall) 5.1 Firewall e VPN
16. 5. Segurança em redes de computadores (Cod Mal) 5.2 Sistemas contra códigos maliciosos
17. 5. Segurança em redes de computadores (IDS/IPS/HN) 5.3 IDS, IPS e Honeyd/Honeynets
18. 5. Segurança em redes de computadores (Wireless) 5.4 Segurança em redes sem fio
19. Apresentação dos temas TE1 Explicação das regras do TE1 Explicação dos temas disponíveis para o TE1 e processo de escolha Definição da data para os alunos informarem as equipes e escolha dos temas
20. Escolha TE1: Temas x Equipes Execução do processo de escolha do tema x equipes Orientações sobre elaboração do projeto do TE1 Explicação da organização da planilha de avaliação do TE1
21. Orientação de escrita do TE1 Orientação da escrita científica Definição de plágio e consequências Explicação do processo de revisão
22. Execução do TE1 em sala Acompanhamento do estado de evolução do TE1 em sala
23. Apresentação do Capítulo TE1 Apresentação do capítulo de fundamentação do TE1
24. Apresentação Final do TE1 Apresentação das equipes do trabalho final (TE1)
25. Avaliação progressiva - questão Avaliação progressiva realizada no final da aula.
26. Avaliação progressiva - resumo Avaliação progressiva feita através de resumo do conteúdo já lecionado Elaboração de resumo referenciado pela equipes
27. Avaliação progressiva - apresentação Avaliação progressiva feita através de resumo do conteúdo já lecionado Apresentação do resumo elaborado na aula anterior
28. Feriado ou ponto facultativo - Feriado ou ponto facultativo. Oficial da UDESC.
29. Segunda chamada da avaliação progressiva Segunda chamada para os alunos que faltaram a alguma das avaliações progressivas
30. Execução TE1 - EAD - Elaboração do TE1 e correção remota via e-mail

Metodologia

1. O método empregado será de aulas expositivas mesclado com trabalhos, que devem ser apresentados em sala de forma a dar o conhecimento necessário ao aluno e induzir o mesmo a pesquisar/aprofundar nos assuntos através de trabalhos. As aulas poderão ser ministradas em modo presencial ou semipresencial (até 20% da carga total no modo semipresencial).

Sistema de avaliação

1. Provas escritas (Avaliações Progressivas, AP1 e AP2); Apresentação de trabalhos em grupo (TE1); Participação efetiva nas aulas (presença, pontualidade, atenção e principalmente contribuição significativa nos estudos realizados). $MS = (TE1*6+AP1*2+AP2*2)/10$ Os estudantes terão, igualmente, a oportunidade de efetuar, ao andamento da disciplina uma avaliação mais completa do desempenho do professor e da disciplina. O formulário para esta atividade será preparado para a ocasião ou fornecido pelo coordenador do curso.

Bibliografia básica

1. - Stallings, William - Cryptography and Network Security: Principles and Practice. 4ª Edição. Prentice-Hall. 2007 - Bishop, Matt. Computer Security: Art and Science. Boston, MA: Addison-Wesley, 2002.

Plano de ensino

Bibliografia complementar

1. - Dias, Claudia - Segurança e Auditoria da Tecnologia da Informação. São Paulo. Axcel Books. 2000
- Schneier, Bruce - Applied Cryptography - Protocols, Algorithms, and Source Code in C. John Wiley & Sons, Inc.. Addison Wesley. 2000
- Schneier, B. - Segurança.com. 1ª Edição. Rio de Janeiro. Campus. 2001
- Garfinkel, Simson; Spafford, Gene - Practical Unix & Internet Security. O'Reilly. 2003
- Sêmola, Marcos - Gestão da Segurança da Informação: Uma Visão Executiva. Rio de Janeiro. Campus. 2003
- Terada, Ruto - Segurança de Dados Criptografia em Redes de Computador. São Paulo. Edgard Blücher. 2000
- Toxen, Bob. - Real World Linux Security. Prentice-Hall. 2001
- Zwicky, Elizabeth D. - Construindo Firewalls para a Internet. 2ª Edição. Rio de Janeiro. Campus. 2000
- Artigos científicos e especializados.

Plano de ensino

Curso: CCI-BAC - Bacharelado em Ciência da Computação

Turma: CCI122-08U - CIENCIA DA COMPUTAÇÃO FASE 08U

Disciplina: TCC2003 - TRABALHO DE CONCLUSÃO DE CURSO II

Período letivo: 2017/1

Carga horária: 36

Professor: 3321045 - GILMARIO BARBOSA DOS SANTOS

Ementa

1. Desenvolvimento do projeto em Ciência da Computação. Execução da pesquisa, fundamentação e elaboração da solução. Desenvolvimento da implementação/modelagem, teste, análise de resultados do projeto em Ciência da Computação.

Objetivo geral

1. Desenvolvimento do TCC-II de acordo com as orientações do professor orientador do acadêmico.

Objetivo específico

1. Desenvolvimento do TCC-II de acordo com as orientações do professor orientador do acadêmico.

Conteúdo programático

1. Apresentação das normas de TCC-II
2. Atendimento/Controle de dados/
3. Atendimento/Controle de dados
4. Atendimento/Controle de dados
5. Atendimento/Controle de dados
6. Atendimento/Controle de dados
7. Atendimento/Controle de dados
8. Atendimento/Controle de dados
9. Atendimento/Controle de dados
10. Atendimento/Controle de dados
11. Atendimento/Controle de dados
12. Atendimento/Controle de dados
13. Atendimento/Controle de dados
14. Atendimento/Controle de dados
15. Atendimento/Controle de dados
16. Atendimento/Controle de dados
17. Atendimento/Controle de dados
18. Atendimento/Controle de dados
19. Atendimento/Controle de dados

Plano de ensino

20. Atendimento/Controle de dados
21. Atendimento/Controle de dados
Metodologia
1. A específica metodologia de desenvolvimento do TCC-II é determinada pelo orientador do trabalho de conclusão de curso.
Sistema de avaliação
1. Nota baseada na apresentação do TCC-II realizada pelo aluno diante de uma banca qualificada.
Bibliografia básica
1. Determinada pelo prof. Orientador do TCC em relação ao tema do trabalho.
Bibliografia complementar
1. Determinada pelo prof. Orientador do TCC em relação ao tema do trabalho.