

Plano de ensino

Curso: CCI-BAC - Bacharelado em Ciência da Computação

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

Disciplina: AGT0001 - ALGORITMOS

Período letivo: 2018/2

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. 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

Exercícios práticos de laboratório

Exercícios de fixação

Prova 1
(escrita)

2. 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
Exercícios de fixação

Prova 1

3. 3. Desvios e Laços
3.1. Estruturas de seleção

Plano de ensino

3.1.1. Seleção simples: (SE...ENTÃO) 3.1.2. Seleção composta: (SE...ENTÃO...SENÃO) 3.1.3. Múltiplas escolhas: (CASO) Exercícios de fixação Prova 1
4. 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) Exercícios práticos de laboratório Exercícios de fixação Prova 2 (escrita)
5. 5. Vetores e Matrizes 5.1. Vetores unidimensionais 5.2. Manipulação de strings (vetores de caracteres) 5.3. Vetores bidimensionais (matrizes) Exercícios práticos de laboratório Exercícios de fixação Prova 3 (escrita)
6. 6. Procedimentos e Funções 6.1. Passagem de parâmetro (por valor e por referência) 6.2. Recursividade Exercícios práticos de laboratório Exercícios de fixação Prova 3 (escrita)
7. 7. Experimentação em linguagem de alto nível 7.1. Introdução à linguagem C Exercícios práticos de laboratório Trabalho Final (prático)

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:
--

Plano de ensino

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 - CIÊNCIA DA COMPUTAÇÃO FASE 01A

Disciplina: LMA0001 - LÓGICA MATEMÁTICA

Período letivo: 2018/2

Carga horária: 72

Professor: 211020122 - JEFERSON LUIZ RODRIGUES SOUZA

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

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

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: CCI122-01B - CIÊNCIA DA COMPUTAÇÃO FASE 01B

Disciplina: AGT0001 - ALGORITMOS

Período letivo: 2018/2

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. 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

Exercícios práticos de laboratório

Exercícios de fixação

Prova 1
(escrita)

2. 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. 3. Desvios e Laços
3.1. Estruturas de seleção
3.1.1. Seleção simples: (SE...ENTÃO)
3.1.2. Seleção composta: (SE...ENTÃO...SENÃO)

Plano de ensino

3.1.3. Múltiplas escolhas: (CASO)
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)
5. Vetores e Matrizes 5.1. Vetores unidimensionais 5.2. Manipulação de strings (vetores de caracteres) 5.3. Vetores bidimensionais (matrizes)
6. Procedimentos e Funções 6.1. Passagem de parâmetro (por valor e por referência) 6.2. Recursividade
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%

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 - CIÊNCIA DA COMPUTAÇÃO FASE 01B

Disciplina: LMA0001 - LÓGICA MATEMÁTICA

Período letivo: 2018/2

Carga horária: 72

Professor: 211020122 - JEFERSON LUIZ RODRIGUES SOUZA

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

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

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 - CIÊNCIA DA COMPUTAÇÃO FASE 01C

Disciplina: AGT0001 - ALGORITMOS

Período letivo: 2018/2

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. 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. 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)
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)
5. Vetores e Matrizes
5.1. Vetores unidimensionais
5.2. Manipulação de strings (vetores de caracteres)
5.3. Vetores bidimensionais (matrizes)
6. Procedimentos e Funções
6.1. Passagem de parâmetro (por valor e por referência)
6.2. Recursividade
7. Experimentação em linguagem de alto nível

Plano de ensino

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) = 30%
Prova 3 (escrita) = 30%
Exercícios avaliativos (escritos e práticos) e 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. COMPLEMENTAR:
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 - CIÊNCIA DA COMPUTAÇÃO FASE 01U

Disciplina: ALG1002 - ÁLGEBRA LINEAR E GEOMETRIA ANALÍTICA I

Período letivo: 2018/2

Carga horária: 72

Professor: 2490099 - JONES CORSO

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. Apresentação do Plano de Ensino da disciplina
2. 1. VETORES NO R3
1.1. Vetores no R3 - representação
3. 1. VETORES NO R3
1.2. Alguns tipos de vetores
4. 1. VETORES NO R3
1.3. Operações com vetores
5. 1. VETORES NO R3
1.4. Adição e propriedades
6. 1. VETORES NO R3
1.5. Subtração
7. 1. VETORES NO R3
1.6. Produto por escalar e propriedades
8. 1. VETORES NO R3
1.7. Base ortonormal
9. 1. VETORES NO R3
1.8. Norma de um vetor
10. 1. VETORES NO R3
1.9. Cálculo de versor
11. 1. VETORES NO R3
1.10. Condição de paralelismo entre dois vetores
12. 1. VETORES NO R3
1.11. Exercícios e aplicações
13. 1. VETORES NO R3
1.12. Produto escalar - Definição
14. 1. VETORES NO R3
1.13. Produto escalar - Propriedades
15. 1. VETORES NO R3
1.14. Produto escalar - Expressão cartesiana
16. 1. VETORES NO R3
1.15. Produto escalar - Interpretação geométrica
17. 1. VETORES NO R3
1.16. Produto escalar - Cossenos diretores de um vetor
18. 1. VETORES NO R3

Plano de ensino

1.17. Produto escalar - Exercícios e aplicações
19. 1. VETORES NO R3 1.18. Produto vetorial - Definição
20. 1. VETORES NO R3 1.19. Produto vetorial - Propriedades
21. 1. VETORES NO R3 1.20. Produto vetorial - Expressão cartesiana
22. 1. VETORES NO R3 1.21. Produto vetorial - Interpretação geométrica
23. 1. VETORES NO R3 1.22. Produto vetorial - Duplo produto vetorial
24. 1. VETORES NO R3 1.23. Produto vetorial - Interpretação geométrica
25. 1. VETORES NO R3 1.24. Produto vetorial - Exercícios e aplicações
26. 1. VETORES NO R3 1.25. Produto misto - Definição
27. 1. VETORES NO R3 1.26. Produto misto - Expressão cartesiana
28. 1. VETORES NO R3 1.27. Produto misto - Propriedades
29. 1. VETORES NO R3 1.28. Produto misto - Interpretação geométrica
30. 2. RETAS NO R3 2.1. Equação vetorial da reta
31. 2. RETAS NO R3 2.2. Equação paramétrica da reta
32. 2. RETAS NO R3 2.3. Equação simétrica da reta e equações reduzidas
33. 2. RETAS NO R3 2.4. Equação da reta determinada por dois pontos distintos
34. 2. RETAS NO R3 2.5. Cossenos diretores de uma reta
35. 2. RETAS NO R3 2.6. Posição relativa entre as retas
36. 2. RETAS NO R3 2.7. Paralelismo e perpendicularismo de retas
37. 2. RETAS NO R3 2.8. Ângulo entre duas retas
38. 2. RETAS NO R3 2.9. Interseção de retas
39. 3. PLANOS NO R3 3.1. Definição
40. 3. PLANOS NO R3 3.2. Equação vetorial do plano
41. 3. PLANOS NO R3 3.3. Equação paramétrica do plano
42. 3. PLANOS NO R3 3.4. Equação geral do plano
43. 3. PLANOS NO R3 3.5. Vetor normal do plano
44. 3. PLANOS NO R3 3.6. Posições relativas de dois planos
45. 3. PLANOS NO R3 3.7. Posições relativas de um plano a uma reta
46. 3. PLANOS NO R3 3.8. Ângulo entre planos
47. 4. DISTÂNCIAS 4.1. Distância entre dois pontos

Plano de ensino

48. 4. DISTÂNCIAS 4.2. Distância de um ponto a uma reta
49. 4. DISTÂNCIAS 4.3. Distância entre duas retas
50. 4. DISTÂNCIAS 4.4. Distância de um ponto a um plano
51. 4. DISTÂNCIAS 4.5. Distância entre dois planos
52. 4. DISTÂNCIAS 4.6. Distância de uma reta a um plano
53. 5. ESTUDO DAS CÔNICAS 5.1. Parábola - Definição
54. 5. ESTUDO DAS CÔNICAS 5.2. Parábola - Elementos da parábola
55. 5. ESTUDO DAS CÔNICAS 5.3. Parábola - Equações da parábola
56. 5. ESTUDO DAS CÔNICAS 5.4. Elipse - Definição
57. 5. ESTUDO DAS CÔNICAS 5.5. Elipse - Elementos da elipse
58. 5. ESTUDO DAS CÔNICAS 5.6. Elipse - Equações da elipse
59. 5. ESTUDO DAS CÔNICAS 5.7. Hipérbole - Definição
60. 5. ESTUDO DAS CÔNICAS 5.8. Hipérbole - Elementos da hipérbole
61. 5. ESTUDO DAS CÔNICAS 5.9. Hipérbole - Equações da hipérbole
62. 5. ESTUDO DAS CÔNICAS 5.10. Hipérboles equiláteras
63. 6. SISTEMAS DE COORDENADAS EM R ² 6.1. Coordenadas polares
64. 6. SISTEMAS DE COORDENADAS EM R ² 6.2. Transformação de coordenadas
65. 6. SISTEMAS DE COORDENADAS EM R ² 6.3. Curvas polares
66. 6. SISTEMAS DE COORDENADAS EM R ² 6.4. Parametrização de curvas
67. 7. ESTUDO DAS QUÁDRICAS 7.1. Superfícies Centradas 7.1.1. Elipsoide
68. 7. ESTUDO DAS QUÁDRICAS 7.1. Superfícies Centradas 7.1.2. Hiperboloide de uma folha
69. 7. ESTUDO DAS QUÁDRICAS 7.1. Superfícies Centradas 7.1.3. Hiperboloide de duas folhas
70. 7. ESTUDO DAS QUÁDRICAS 7.2. Superfícies não Centradas 7.2.1. Paraboloide Eliptico
71. 7. ESTUDO DAS QUÁDRICAS 7.2. Superfícies não Centradas 7.2.2. Paraboloide Hiperbólico
72. 7. ESTUDO DAS QUÁDRICAS 7.3. Superfície cônica
73. 7. ESTUDO DAS QUÁDRICAS 7.4. Superfície Cilíndrica
74. 8. SISTEMAS DE COORDENADAS EM R ³ 8.1. Coordenadas Cilíndricas
75. 8. SISTEMAS DE COORDENADAS EM R ³

Plano de ensino

8.2. Coordenadas Esféricas
76. 8. SISTEMAS DE COORDENADAS EM R3 8.3. Mudança de Coordenadas
77. 8. SISTEMAS DE COORDENADAS EM R3 8.4. Parametrização de superfícies
78. Resolução de exercícios dirigidos

Metodologia

1. As aulas serão realizadas de forma expositiva e dialogada com a utilização do quadro negro, sendo que em alguns momentos serão utilizados alguns sólidos geométricos e ferramentas tecnológicas como apoio didático. Algumas aulas poderão ser reservadas para a resolução de exercícios, além disso, os estudantes terão direito a atendimento individualizado no horário pré-estabelecido pelo professor.

Sistema de avaliação

1. A avaliação do desempenho do estudante na disciplina acontecerá por meio de quatro avaliações escritas realizadas individualmente, todas elas com o mesmo peso, portanto, a Média Semestral dos estudantes será calculada através da média aritmética das notas obtidas nestas quatro avaliações escritas.

EXAME: Será realizado em 11/12/2018, em horário de aula, conforme resolução em vigor da UDESC. Será uma prova dissertativa individual referente ao conteúdo programático da disciplina.

INFORMAÇÕES IMPORTANTES A RESPEITO DA APROVAÇÃO:

1. Se o número de presenças for inferior a 75% do número total de aulas da disciplina, o estudante está automaticamente reprovado por falta, independentemente de ter alcançado alguma nota na disciplina.

2. Se o número de presenças for igual ou superior a 75% do número total de aulas da disciplina e:

2.1. A Média Semestral for maior ou igual a 7,0 (sete), o estudante obtém aprovação na disciplina.

2.2. Se a Média Semestral for maior ou igual a 1,7 (um vírgula sete) e menor que 7,0 (sete), o estudante terá direito a realizar um Exame e será calculada uma Média Final da seguinte forma:

Média Final = $[(6 \cdot (\text{Média Semestral}) + 4 \cdot (\text{Nota do Exame})) / 10]$.

Se esta Média Final for maior ou igual a 5,0 (cinco) o estudante obtém aprovação na disciplina. E se a Média Final for menor que 5,0 (cinco) o estudante não obtém aprovação e está reprovado por nota.

DIVULGAÇÃO DE NOTAS E FREQUÊNCIA: no Sistema de Gestão Acadêmico (SIGA) disponível em: [/siga.udesc.br/](http://siga.udesc.br/).

Bibliografia básica

1. BOULOS, P.; CAMARGO, I. Geometria Analítica. Um Tratamento Vetorial. Makron Books Editora, 1990.
STEINBRUCH, A.; WINTERLE, P. Geometria Analítica. 2ª ed. Makron Books Editora, 1987.
VENTURI, J. J. Cônicas e Quádricas. 5ª ed. Curitiba (PR): Artes Gráficas e Editora Unificado, 2003. Disponível em [/geometriaa.dominiotemporario.com/livros/cq.pdf](http://geometriaa.dominiotemporario.com/livros/cq.pdf) sob licença do autor. Acesso em 09/02/2018.

Bibliografia complementar

1. BOULOS, P.; CAMARGO, I. Introdução à Geometria Analítica no Espaço. Makron Books Editora, 1997.
LEHMANN, C. H. Geometria Analítica. Editora Globo, 1982.
LEITHOLD, L. O Cálculo com Geometria Analítica. Vol 1. 3ª ed. Editora Harbra, 1994.
VENTURI, J. J. Álgebra Vetorial e Geometria Analítica. 10ª ed. Curitiba (PR): Livrarias Curitiba, 2015. 242 p. Disponível em [/geometriaa.dominiotemporario.com/livros/av.pdf](http://geometriaa.dominiotemporario.com/livros/av.pdf) sob licença do autor. Acesso em: 09/02/2018.

Plano de ensino

Curso: CCI-BAC - Bacharelado em Ciência da Computação

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

Disciplina: CDI1001 - CÁLCULO DIFERENCIAL E INTEGRAL I

Período letivo: 2018/2

Carga horária: 108

Professor: 211811206 - RAIANE LEMKE

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 de limites, derivadas e integrais de funções reais de uma variável real.

Objetivo específico

1. - 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 por meio de técnicas, dos limites notáveis e da regra de L'Hôpital.
- Estudar a continuidade de uma função.
- Derivar funções.
- Interpretar geometricamente/fisicamente derivadas e diferenciais.
- Resolver problemas com diferenciais.
- Analisar a variação das funções e construir seus gráficos.
- Resolver problemas de otimização por meio da teoria de derivadas.
- Determinar primitivas de funções através de técnicas de integração.

Conteúdo programático

1. Números, variáveis e funções:
 - 1.1. Conjuntos numéricos.
 - 1.2. Desigualdades.
 - 1.3. Valor absoluto.
 - 1.4. Funções.
 - 1.4.1. Definição.
 - 1.4.2. Formas de representação.
 - 1.4.3. Operações com funções.
 - 1.4.4. Função par, ímpar e periódica.
 - 1.4.5. Função composta.
 - 1.4.6. Translações.
 - 1.4.7. Função polinomial.
 - 1.4.8. Função racional.
 - 1.4.9. Função potencial.
 - 1.4.10. Função modular.
 - 1.4.11. Função inversa.
 - 1.4.12. Função exponencial.
 - 1.4.13. Função logarítmica.
 - 1.4.14. Funções hiperbólicas.
 - 1.4.15. Funções trigonométricas.
 - 1.4.16. Função inversa de trigonometria.
2. Limite e continuidade de uma função:
 - 2.1 Definição formal de limite.
 - 2.2 (Não) Existência de limites.
 - 2.3 Propriedades de limites.
 - 2.4 Cálculo de limites.
 - 2.5 Limites notáveis.
 - 2.6 Estudo da continuidade.
3. Derivada e Diferencial:
 - 3.1. Interpretação cinemática da derivada de uma função.
 - 3.2. Interpretação geométrica da derivada de uma função.
 - 3.3. Definição de derivada de uma função.
 - 3.4. Regras de derivação.
 - 3.4.1. Derivada da função constante.

Plano de ensino

3.4.2. Regra da potência. 3.4.3. Derivada da soma. 3.4.4. Regra do produto. 3.4.5. Regra do quociente. 3.5. Derivada de função composta. 3.6. Derivada de função exponencial. 3.7. Derivada de função logaritmo. 3.8. Derivada de funções trigonométrica. 3.9. Derivada de funções hiperbólica 3.10. Derivação implícita. 3.11. Derivada da função inversa. 3.12. Derivadas de ordem superior. 3.13. Taxas relacionadas; 3.14 Diferenciais e aproximação linear local. 3.15 Regra de L'Hopital
4. Análise da variação das funções: 4.1 Ponto crítico. 4.2 Ponto extremo. 4.3 Teoremas sobre funções deriváveis. 4.4 Critérios para determinação dos extremos de uma função. 4.5 Concavidade. 4.6 Pontos de inflexão. 4.7 Assíntotas. 4.8 Construção de gráfico de funções usando a teoria de derivadas. 4.9 Otimização.
5. Integral indefinida: 5.1 Definição de primitiva. 5.2 Definição de integral indefinida. 5.3 Propriedades de integral indefinida; 5.4 Integrais imediatas; 5.5 Integração por substituição; 5.6 Integração por partes; 5.7 Integração de funções trigonométricas; 5.8 Integrais por substituição trigonométrica; 5.9 Integrais elementares que contém um trinômio quadrado; 5.10 Integração de funções racionais por decomposição em frações parciais.

Metodologia

1. As aulas serão expositivas e dialogadas, com resolução de exercícios orientados.

Sistema de avaliação

1. Cinco avaliações escritas individuais durante o semestre letivo com nota de 0 a 10,0 cada. Poderão ser realizados trabalhos individuais e/ou em grupos. Quando houver algum tipo trabalho avaliativo, a nota final da respectiva prova será obtida pela soma da nota da prova escrita com a nota do(s) trabalho(s) realizado(s).

Bibliografia básica

1. ANTON, H. Cálculo, um novo horizonte. Porto Alegre: Bookman, vol. 1, 6ª ed., 2000. 2. FLEMING, D. M.; GONÇALVES, M. B. Cálculo A. São Paulo: Pearson Prentice Hall, 6ª ed. rev. e ampl., 2006. 3. STEWART, J. Cálculo. São Paulo. Cengage Learning, vol. 1, 6ª ed, 2009.

Bibliografia complementar

1. KÜHLKAMP, N. Cálculo 1. Florianópolis. Editora UFSC, 3ª ed. rev. e ampl. 2006. 2. LEITHOLD, L. O Cálculo com Geometria Analítica. São Paulo. Editora HARBRA Ltda, 3ª ed., 1994. 3. PISKOUNOV, N. Cálculo Diferencial e Integral. Moscou, Editorial Mir, 4ª ed., 1977. 4. SWOKOWSKI, Earl William. Cálculo com geometria analítica. 2. ed. São Paulo: Makron Books, v.1, 1995. 5. WEIR, Maurice D; HASS, Joel; GIORDANO, Frank R; THOMAS, George Brinton; ASANO, Claudio Hirofume. Cálculo: George B. Thomas. 11. ed. São Paulo: Pearson/Addison Wesley, v.1, 2009. 6. Apostila de Cálculo Diferencial e Integral I. Disponível em http://www.joinville.udesc.br/portal/professores/eliane/materiais/ApostilaCDI_2016_2.pdf Departamento de Matemática, CCT/Udesc, Joinville.

Plano de ensino

Curso: CCI-BAC - Bacharelado em Ciência da Computação

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

Disciplina: MCI0001 - METODOLOGIA CIENTÍFICA

Período letivo: 2018/2

Carga horária: 36

Professor: 3198057 - CARLA DIACUI MEDEIROS BERKENBROCK

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. Desenvolver habilidades, capacidades e competências relacionadas com a metodologia científica, de tal forma a auxiliar na produção e na divulgação do conhecimento na área da ciência da computação.

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;
- Abordar a necessidade e exercício da ética nas pesquisas e nas demais atividades acadêmicas;
- Ler, analisar e interpretar textos para desenvolver autoria crítica;
- Conhecer os tipos de comunicação técnico-científicas: sumarização e resumo, resenha, artigos, periódicos, ensaios, trabalho de conclusão de curso, relatório, dissertação, tese;
- Estabelecer as diferenças, particularidades e similaridades entre os diferentes trabalhos científicos;
- Identificar e analisar 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 autoplágio;
- Desenvolver e aperfeiçoar a comunicação na área da ciência da computação;
- Incentivar a elaboração e publicação de trabalhos científicos, bem como a participação de trabalhos em eventos da área da ciência da computação.

Conteúdo programático

1. Aula Inaugural (Apresentação da Professora e alunos/ Apresentação do Plano de Ensino/Organização e detalhamento das atividades avaliativas)
2. Pesquisa tecnológica, Contexto universitário, A ética na pesquisa
3. Diretrizes para a leitura, análise e interpretação de textos
4. Comunicação técnico-científica (sumarização e resumo/ resenha/artigo, paper e ensaio, relatório e trabalho de conclusão de curso - TCC)
5. Normatização do documento científico
6. Comunicação oral

Metodologia

1. A disciplina será trabalhada por meio da temática teórico-prática, com ênfase a discussão ao debate em grupo, assim como uma forte carga de leitura e aulas expositivas complementaram o trabalho.

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) Seminários (SEM)
e) Elaboração e apresentação de um artigo (ART)

Média = SEM*0,40 + ART*0,60

Plano de ensino

Bibliografia básica

1. ANDRADE, Maria Margarida de. Introdução à metodologia do trabalho científico: elaboração de trabalhos na graduação. 5. ed. São Paulo: Atlas, 2001. 174 p.
2. BASTOS, Cleverson Leite; KELLER, Vicente. Aprendendo a aprender: introdução a metodologia científica. 22. ed. Rio de Janeiro: Vozes, 2008. 111 p.
3. MARCONI, Marina de Andrade; LAKATOS, Eva Maria. Fundamentos de metodologia científica. 7. ed. São Paulo: Atlas, 2010. 297 p.

Bibliografia complementar

1. BARROS, Aidil de Jesus Paes de; LEHFELD, Neide Aparecida de Souza. Fundamentos de metodologia científica: um guia para a iniciação científica. 2. ed. ampl. São Paulo: Pearson Education, 2000. 122 p.
2. FACHIN, Odília. Fundamentos de metodologia. 3. ed. São Paulo: Saraiva, 2001. 200 p.
3. LUZ, A. C. da et al. Manual para elaboração de trabalhos acadêmicos da UDESC: tese, dissertação, trabalho de conclusão de curso e relatório de estágio. Universidade do Estado de Santa Catarina, 4ª ed. Florianópolis: UDESC, 2013. Disponível em: http://www.udesc.br/arquivos/id_submenu/6/manual_a4___abnt.pdf. Acesso em: 20/07/2016.
4. PINHEIRO, J. M. S. Da iniciação científica ao TCC. Uma abordagem para os cursos de tecnologia. Rio de Janeiro: Ed. Ciência Moderna Ltda., 2010.
5. SILVA, E. da; TAFNER, E. P.; FISCHER, J.; MALCON, A. T. Metodologia do trabalho acadêmico. 3. ed. rev. e atual. Curitiba: Juruá Ed., 2010. 131 p.

Plano de ensino

Curso: CCI-BAC - Bacharelado em Ciência da Computação

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

Disciplina: TGS0001 - TEORIA GERAL DE SISTEMAS

Período letivo: 2018/2

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. 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).
2. Filosofia da Ciência
Visão geral, proposição e limites da ciência.
3. Histórico da TGS
Origem, propósito, significado e proposta da TGS dentro da filosofia da ciência.
4. 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.
5. Características dos Sistemas
Retroação, input/output de energia, entropia X entropia negativa, equifinalidade, endocausalidade, retroação, homeostase e estabilidade, diferenciação, autopoiesis, auto-referência, modelo de informação isomórfico ao da entropia negativa.
6. 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
7. Cibernética
Insurgência das causas sobre seus efeitos, o pensamento artificial, retroinformação negativa, revitalização da teleologia, tectologia.
8. 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 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

Plano de ensino

complexidade de Morin (sinergia, totalidade, organização), teoria do Caos, teoria dos jogos.

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.

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. 3ª Edição. Petrópolis. Vozes, 2008.
KATZ & KAHN, D., R. Psicologia Social das Organizações. São Paulo. Atlas, 1974.
VASCONCELLOS, M.J.E. Pensamento sistêmico - o novo paradigma da ciência. 10ª Edição. Campinas. Papirus Editora, 2016.

Bibliografia complementar

1. ALVES, Rubem. Filosofia da Ciência. 12ª edição. São Paulo. Loyola, 2000.
MATURANA, Humberto; VARELA, Francisco. A árvore do conhecimento. Campinas. Editorial Psy II, 1995..
MORIN, Edgar. O Método - 4 - as idéias. Porto Alegre. Editora Sulina, 1998.
MORIN, Edgar. O Método - 3 - o conhecimento do conhecimento. Porto Alegre. Editora Sulina, 1999.
SENGE, P. A quinta disciplina: teoria e prática da organização de aprendizagem. São Paulo. Nova Cultural, 1990.

Plano de ensino

Curso: CCI-BAC - Bacharelado em Ciência da Computação

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

Disciplina: LPG0001 - LINGUAGEM DE PROGRAMAÇÃO

Período letivo: 2018/2

Carga horária: 72

Professor: 2646943 - OMIR CORREIA ALVES 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. Apresentação do plano de ensino
2. avaliação
3. referências bibliográficas
4. revisão de lógica
5. revisão de algoritmos e linguagem C
5.1. Características
5.2. Tipos, Constantes e Variáveis
5.3. Operadores, Expressões e Funções
6. Funções de Entrada (scanf) e de Saída (printf)
7. exercícios
2. 1. Estruturas Sequenciais/seleção
1.1. switch ... case
1.2. if
1.3. if ... else
2. Estruturas de iteração
2.1. for
2.2. do ... while
2.3. while
3. exercícios
3. 1. tipos de dados em C
2. Funções
2.1. Parâmetros passados por valor
2.2. Parâmetros passados por referência
3. exercícios
4. 1. Recursividade
2. exercícios
5. 1. Arranjos Vetores e Matrizes
1.1. iniciação de vetores e matrizes
1.2. passagem de parâmetros
2. exercícios
6. 1. Ponteiros
1.1. Conceito e funcionamento
1.2. Declaração e Utilização
1.3. Aritmética de ponteiros
1.4. Iniciação de ponteiros
1.5. Endereçamento de elementos de vetores
1.6. Ponteiros e strings
1.7. Ponteiros para funções
1.8. Ponteiros para ponteiros
2. exercícios

Plano de ensino

7.	1. Alocação dinâmica de memória 1.1. Conceitos 1.2. Funções de alocação de memória 1.3. Realocação 1.4. Liberação 1.5. Alocação dinâmica de vetores e matrizes 2. exercícios
8.	1. Cadeia de caracteres 1.1 biblioteca de tratamento de caracteres 1.2 funções de entrada e de saída 2. exercícios
9.	1. Tipos de dados definidos pelo usuário 1.1 Estruturas 1.2 Criando e usando uma estrutura 1.3 Atribuições entre estruturas 1.4 Estruturas aninhadas 1.5 Passagem para funções 1.6 Ponteiros para estruturas 1.7 Vetor de estruturas 1.8 Alocação dinâmica de estruturas 2. exercícios
10.	1. Union 2. Enumerações 3. typedef 4. Diretivas de compilação 5. exercícios
11.	1. Noções de arquivos 1.1 Introdução 1.2 Abrindo e fechando 1.3 Modo texto e binário 1.4 Entrada e saída formatada 2. exercícios 1.5 Leitura e gravação 1.6 Lendo e gravando registros 1.7. Acesso aleatório
12.	1. manipulação de arquivos 1.1 Lendo e gravando registros em um arquivo 1.2 formatação de E/S no modo texto 1.3. Acesso randômico à arquivos 2. exercícios
13.	1. Construindo módulos em C 2. exercícios
14.	Primeira avaliação de LPG0001
15.	Segunda avaliação de LPG0001
16.	primeiro teste prático de LPG0001
17.	segundo teste prático de LPG0001
18.	resolução de exercícios
19.	Projeto da disciplina LPG0001
20.	Apresentação dos Projetos da disciplina LPG0001
21.	correção da primeira avaliação de LPG0001
22.	correção da segunda avaliação de LPG0001

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.
----	---

Plano de ensino

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
Artigos/materiais fornecidos pelo professor.
Linguagem C: Completa e Descomplicada. Backes, Andre, Editora Campus Elsevier, 2012.
Curso de Linguagem em C, Adriano Joaquim de Oliveira Cruz, UFRJ, 2016

Plano de ensino

Curso: CCI-BAC - Bacharelado em Ciência da Computação

Turma: CCI122-02B - CIÊNCIA DA COMPUTAÇÃO FASE 02B

Disciplina: LPG0001 - LINGUAGEM DE PROGRAMAÇÃO

Período letivo: 2018/2

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 - CIÊNCIA DA COMPUTAÇÃO FASE 02U

Disciplina: ALG2002 - ÁLGEBRA LINEAR E GEOMETRIA ANALÍTICA II

Período letivo: 2018/2

Carga horária: 72

Professor: 6577849 - DEBORA DE FARIA FERREIRA GOMES

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. 1. Identificar os vários tipos de matrizes, calcular determinantes, classificar sistemas lineares e resolver problemas diversos utilizando sistemas de equações lineares.
2. Definir espaço vetorial com suas operações, propriedades e teoremas e resolver problemas envolvendo esses conceitos.
3. Compreender o conceito de transformação linear, suas propriedades, operações, sua representação matricial.
4. Reconhecer transformações lineares como uma importante ferramenta na resolução de problemas em diversas áreas do conhecimento.
5. 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.
6. Identificar o produto interno como uma ferramenta potencial na resolução de problemas.

Conteúdo programático

1. Matrizes e sistemas:
 - 1.1 Tipos especiais de matrizes
2. 1.2 Operações com matrizes
3. 1.3 Determinante de uma matriz
4. 1.4 Matriz linha reduzida e matriz escalonada
5. 1.5 Matriz inversa
6. 1.6 Sistemas de equações lineares
7. 1.7 Matriz ampliada de um sistema
8. 1.8 Classificação de um sistema de equações $m \times n$
9. 1.9 Resolução de um sistema linear
 - 1.9.1 Método de escalonamento de Gauss
10. 1.9.2 Método da inversa
11. Avaliação 1
12. 2. Espaço vetorial:
 - 2.1 Definições de espaço vetorial e subespaço vetorial
13. 2.2 Dependência e independência linear
14. 2.3 Interseção e soma de subespaços vetoriais
15. 2.4 Subespaço gerado por um conjunto de vetores
16. 2.5 Base e dimensão de um espaço vetorial
17. 2.6 Matriz mudança de base e sua inversa
18. Avaliação 2
19. 3. Transformações lineares:
 - 3.1 Definição de transformação linear

Plano de ensino

20. 3.2 Propriedades das transformações lineares
21. 3.3 Núcleo e imagem de uma transformação linear
22. 3.4 Transformações lineares injetora e sobrejetora
23. 3.5 Transformações induzidas por uma matriz
24. 3.6 Composição de transformações lineares
25. 3.7 Matriz de uma transformação linear
26. 3.8 Isomorfismo e inversa de uma transformação linear
27. 4. Operadores lineares
4.1 Transformações especiais no plano
28. 4.2 Transformações especiais no espaço
29. 4.3 Operadores auto-adjuntos e ortogonais
30. 4.4 Operadores inversíveis
31. Avaliação 3
32. 5. Autovalores e autovetores:
5.1 Definição e exemplos
33. 5.2 Autovalores e autovetores de uma matriz
34. 5.3 Polinômio característico
35. 5.4 Cálculo de autovalores e autovetores
36. 5.5 Matrizes semelhantes
37. 5.6 Diagonalização de operadores
38. 6. Produto interno
6.1 Definição de produto interno
39. 6.2 Ortogonalidade em espaços com produto interno
40. 6.3 Complementos e projeções ortogonais
41. 6.4 Bases ortonormais; processo de Gram-schmidt
42. 6.5 Fatoração QR.
43. Avaliação 4
44. Exame Final

Metodologia

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

Sistema de avaliação

1. Serão feitas quatro avaliações escritas individuais. A média semestral será a média aritmética das provas. Datas prováveis das provas: Prova 1: 23/08/2018. Prova 2: 25/09/2018. Prova 3: 25/10/2018. Prova 4: 29/11/2018. Exame: Conforme Resolução. Data provável para o exame: 11/12/2018.
--

Bibliografia básica

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

Plano de ensino

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.

Plano de ensino

Curso: CCI-BAC - Bacharelado em Ciência da Computação

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

Disciplina: CDI2001 - CÁLCULO DIFERENCIAL E INTEGRAL II

Período letivo: 2018/2

Carga horária: 72

Professor: 211811125 - LUIS GUSTAVO LONGEN

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. 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.

Metodologia

Plano de ensino

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.

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 - CIÊNCIA DA COMPUTAÇÃO FASE 02U
Disciplina: EST0007 - PROBABILIDADE E ESTATÍSTICA
Período letivo: 2018/2
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 - CIÊNCIA DA COMPUTAÇÃO FASE 02U

Disciplina: FCC0002 - FÍSICA PARA CIÊNCIA DA COMPUTAÇÃO

Período letivo: 2018/2

Carga horária: 72

Professor: 2888874 - ANDRE LUIZ DE OLIVEIRA

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 Eletricidade e do Magnetismo.
 - 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. Revisão
35. Prova 4ª Prova semestral.

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. - Quatro prova de mesmo peso.
- A média semestral será determinada pelo cálculo da média aritmética das notas obtidas.

Bibliografia básica

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.

Plano de ensino

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 - CIÊNCIA DA COMPUTAÇÃO FASE 02U

Disciplina: MDI0001 - MATEMÁTICA DISCRETA

Período letivo: 2018/2

Carga horária: 72

Professor: 9672753 - LEANDRO ISRAEL PINTO

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. Estruturas Algébricas: Operações Binárias
19. Estruturas Algébricas: Propriedades das Operações Binárias
20. Estruturas Algébricas: Semigrupos, Monóides e Grupos
21. Homomorfismos
22. Reticulados: Limitantes de Conjuntos Parcialmente Ordenados

Plano de ensino

23. Reticulados: Reticulados como Relação de Ordem e como Álgebra
24. Reticulados: Tipos de Reticulados
25. Álgebra de Boole
26. Contagem: Princípios da Multiplicação e da Adição
27. Contagem: Princípios de Inclusão e Exclusão
28. Contagem: Princípio das Casas de Pombo

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 e P2).
A Média Semestral (MS) será calculada pela fórmula
$$MS = 0.45 P1 + 0.4 P2 + 0.15 AC$$

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 - CIÊNCIA DA COMPUTAÇÃO FASE 02U

Disciplina: SNA0001 - SISTEMAS DE NUMERAÇÃO E ÁLGEBRA DE BOOLE

Período letivo: 2018/2

Carga horária: 36

Professor: 2877546 - CLEVERSON PONTELLI SANTOS

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;
- Introduzir Funções booleanas e formas canônicas;
- Introduzir Mapas de Karnaugh.

Conteúdo programático

1. Aula Inaugural - Plano de Ensino e Métodos de Avaliação.

2. Capítulo I - Sistemas de Numeração:

- 1.1 Objetivo;
- 1.2 Tipos;
- 1.3 Base;
- 1.4 Potência;
- 1.5 Principais sistemas.

3. 1.6 Conversão entre sistemas de numeração.

- 1.7 Aula Prática - Resolução de Exercícios

4. Capítulo II - Representação Binária da Informação:

- 2.1 Informações não numéricas;
- 2.2 Informações numéricas.

5. 2.3 Representação de Inteiros com e sem sinal;

- 2.4 Representação de reais - Ponto flutuante.
- 2.5 Aula Prática - Resolução de Exercícios

6. Capítulo III - Funções de Variáveis Lógicas:

- 3.1 Relação funcional;
- 3.2 Tabela verdade;
- 3.3 Funções AND, OR e NOT.
- 3.4 Aula Prática - Resolução de Exercícios

7. Semana da Computação.

8. AVALIAÇÃO - P1

9. Capítulo IV - Álgebra de Boole

- 4.1 Postulados e Teoremas fundamentais
- 4.2 Funções Booleanas

10. 4.3 Aplicações da Álgebra Booleana

- 4.4 Formas Canônicas: Minitermos e Maxitermos

Plano de ensino

4.5 Aula Prática - Resolução de Exercícios.
11. Capítulo V - Mapas de Karnaugh: 5.1 Construção de mapas de até 6 variáveis.
12. 5.2 Simplificação com Mapas Karnaugh; 5.3 Mapeamento de funções fora da forma padrão e incompletamente especificadas. 5.4 Aula Prática - Resolução de Exercícios.
13. AVALIAÇÃO - P2
14. Provas sobre Capítulos Teóricos.(P1 + P2). Peso=70% Elaboração e Apresentação de Trabalhos em Grupo com entrega (TG). Peso=10% Resolução de Exercícios com Entrega (RE). Peso=20%

Metodologia

1. Aulas expositivas e dialogadas com aulas práticas para desenvolvimento de exercícios. A disciplina poderá oferecer 20 % de sua carga horária na modalidade 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) Provas sobre o(s) capítulo(s) teóricos (P1 e P2) - (80 %) b) Elaboração de trabalho em grupo com entrega (TG) - (20 %)
--

Bibliografia básica

1. MELO, M. O. Eletrônica Digital. Editora da UDESC. Florianópolis, 2002. APOSTILA - Sistemas Numéricos e Álgebra de Boole. LOURENÇO, A. C. Sistemas Numéricos e Álgebra Booleana. Editora Érica. BIGNELL, James. DONOVAN, Robert. Eletrônica Digital. Cengage Learning, 2009.

Bibliografia complementar

1. LORIN, H. Introdução à Arquitetura e Organização de Computadores. Ed. Campus. 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 - CIÊNCIA DA COMPUTAÇÃO FASE 03U

Disciplina: ANN0001 - ANÁLISE NUMÉRICA

Período letivo: 2018/2

Carga horária: 72

Professor: 3485641 - LUIZ ANTONIO FERREIRA COELHO

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. Aprender métodos numéricos utilizados para resolver problemas de cunho científico e tecnológico em engenharia. Resolver problemas em computador com os métodos estudados, usando rotinas prontas e também implementando códigos em linguagem de alto nível.

Objetivo específico

1. O aluno deverá ao final do semestre letivo ser capaz de utilizar os conceitos e métodos numéricos para:
 - * resolver sistemas lineares e não lineares
 - * encontrar zeros de funções
 - * efetuar interpolação polinomial
 - * ajustar curvas a dados
 - * calcular integrais definidas
 - * solucionar problemas de equações diferenciais
 - * elaborar e aplicar rotinas dos métodos estudados em linguagem de alto nível

Conteúdo programático

1. Sistemas de ponto flutuante
2. Erros: existência, tipos e propagação
3. Métodos iterativos para se obter zeros de funções
4. Método da bissecção
5. Método da posição falsa
6. Método de iteração de ponto fixo
7. Método de Newton-Raphson
8. Método da secante
9. Raízes de polinômios
10. Resolução de sistemas lineares pelo método direto da eliminação de Gauss
11. Resolução de sistemas lineares pelo método iterativo de Jacobi
12. Resolução de sistemas lineares pelo método iterativo de Gauss-Seidel
13. Resolução de sistemas não-lineares
14. Interpolação numérica e o polinômio interpolador
15. Interpolação por polinômios de Lagrange
16. Interpolação por diferenças divididas de Newton
17. Interpolação por partes: splines
18. Ajuste de curvas a dados por mínimos quadrados
19. Integração numérica pela regra dos trapézios
20. Integração numérica pela regra de Simpson

Plano de ensino

21. Integração numérica por quadratura gaussiana

22. Método de Euler para equações diferenciais

23. Aplicações e implementação em computador

Metodologia

1. Aulas expositivas e dialogadas com resolução de exercícios orientados e atividades com auxílio de computador.

Sistema de avaliação

1. Serão feitas três avaliações escritas individuais, de mesmo peso, e um ou mais trabalhos ao longo do semestre letivo. A média semestral será a média ponderada $M = (3P + T)/4$, onde P é a média das provas e T é a média dos trabalhos. Em outras palavras, a média semestral será gerada pela adição de 75% da média das provas com 25% da média do(s) trabalho(s) realizado(s).

Bibliografia básica

1. * RUGGIERO, M. A. G.; LOPES, V. L. R. Cálculo Numérico: Aspectos Teóricos e Computacionais. 2. ed. São Paulo: McGraw Hill, 1996.
* CLAUDIO, D. M.; MARINS, J. M. Cálculo Numérico Computacional: Teoria e Prática. 3. ed. São Paulo: Atlas, 2000.
* BARROSO, L. C.; BARROSO, M. M.; CAMPOS, F. F.; CARVALHO, M. L. B.; MAIA, M. L. Cálculo Numérico: com aplicações. 2. ed. Harbra: São Paulo, 1987.

Bibliografia complementar

1. * BURDEN, R. L.; FAIRES, J. D.. Análise Numérica. 8. ed. São Paulo: Cengage Learning, 2008.
* CHAPRA, S. C.; CANALE, R. P. Métodos Numéricos para Engenharia. 5. ed. São Paulo: McGraw-Hill, 2008.
* CUNHA, M. C. C. Métodos Numéricos. 2. ed. Campinas: Editora da Unicamp, 2003.
* GILAT, A.; SUBRAMANIAM, V. Métodos Numéricos para Engenheiros e Cientistas. Bookman, 2008.
* RALSTON, F. F. Algoritmos Numéricos. 2. ed. Rio de Janeiro: LTC, 2007.

Plano de ensino

Curso: CCI-BAC - Bacharelado em Ciência da Computação

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

Disciplina: AOC0003 - ARQUITETURA E ORG. DE COMPUTADORES

Período letivo: 2018/2

Carga horária: 72

Professor: 2877546 - CLEVERSON PONTELLI SANTOS

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 discente a compreender os circuitos digitais combinacionais e sequenciais básicos;
- Habilitar o discente a realizar experimentos e montagens de projetos e circuitos digitais básicos;
- Capacitar o aluno a entender o funcionamento geral de computadores digitais;
- Auxiliar o discente a compreender melhor a funcionalidade e utilização dos blocos constituintes de computadores digitais, bem como, dos dispositivos de entrada e saída;
- Apresentar a organização interna de arquiteturas de microprocessadores;
- 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 - Aula Inaugural
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 Implementação de funções booleanas
5. 1.5 Formas Canônicas
6. 1.6 Minimização de Funções
7. 1.7 Mapas de Karnaugh
8. 1.8 Circuitos Aritméticos
9. 1.9 Atividades de Laboratório.
10. Semana da Computação.
11. Capítulo II - Circuitos Lógicos Sequenciais
 - 2.1 Circuitos Flip-Flops
12. 2.2 Circuitos Flip-Flops com clock
13. 2.3 Entradas assíncronas
2.4 Características de Temporização dos Flip-Flops
2.5 Contadores Síncronos e Assíncronos
2.6 Registradores
14. 2.7 Atividades de Laboratório.

Plano de ensino

15. Semana da Computação
16. Capítulo III - O Sistema de Computação
3.1 O conceito de Processamento de Dados
17. 3.2 Representação de informações
18. 3.3 Conceito de arquivos e registros
3.4 Medidas de desempenho de sistemas computacionais
19. 3.5 A arquitetura e Organização do computador
20. 3.6 Os níveis de abstração de um computador
3.7 A influência do Sistema Operacional.
21. 3.8 Atividades de Laboratório.
22. AVALIAÇÃO - P1
23. Capítulo IV - A Unidade Central de Processamento - CPU
4.1 A placa-mãe (Motherboard)
24. 4.2 Barramentos
25. 4.3 Funções Básicas da CPU
4.4 Circuitos de apoio (IRQ's, DMA, etc.)
26. 4.5 A Unidade Lógica e Aritmética
4.6 Os registradores da CPU
27. 4.7 Instruções de máquina
28. 4.8 Formato das instruções
29. 4.9 Arquitetura RISC e CISC
4.10 Pipeline de instruções
30. 4.11 Noções de Linguagem de montagem (Assembly)
31. 4.12 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
5.7 Memória secundária
36. 5.8 Latência
5.9 Tipos e nomenclatura de memórias
37. Capítulo VI - Noções de Processamento Paralelo
6.1 Organização Symmetric Multiprocessor - SMP
38. 6.2 Clusters
6.3 Tipos de Sistemas com Processadores Paralelos

Plano de ensino

39. 6.4 Processadores Multicore.
40. AVALIAÇÃO - P2
41. Capítulo VII - Noções de Microcontroladores 7.1 Conceitos
42. 7.2 Symmetric Multiprocessing - SMP's 7.3 Programmable Intelligent Computer - PCI's 7.4 FPGA 7.5 Arduino
43. 7.4 Projetos com Microcontroladores. 7.5 Arduino - Hardware e Software
44. 7.6 Atividades de Laboratório - Arduino
45. Apresentação de Trabalhos em Grupo
46. Capítulo VIII- Dispositivos de entrada e saída 8.1 Módulos de Entrada e Saída 8.2 Funções de Entrada e Saída 8.3 Formas de Comunicação entre CPU e dispositivos de E/S 8.4 Teclado, Mouse, Joystick, Mesa digitalizadora, Scanner
47. 8.5 Impressoras, Plotter e Monitores de Vídeo 8.6 Fita Magnética, Discos Magnéticos, Óticos, CD, DVD, Cartão de memória, Pen drive, Unidades de Estado Sólido
48. 8.7 Motherboard 8.8 Padrões de barramento
49. 8.9 Microprocessadores PC e dispositivos móveis 8.10 Montagem e Manutenção de Microcomputadores
50. Avaliação do Trabalho em grupo (TG).

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. 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) com Entrega - (20 %)
 - c) Atividades de Laboratório (LAB) - (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 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

Plano de ensino

1. APOSTILA: Arquitetura e Organização de Computadores. FERNANDES, E.M.L BIBLIOGRAFIA ET AL
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 - CIÊNCIA DA COMPUTAÇÃO FASE 03U

Disciplina: EDA0001 - ESTRUTURA DE DADOS

Período letivo: 2018/2

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. Revisão de C e introdução à análise de algoritmos
2. Abstração de Dados
3. Estrutura de Dados
Pilhas (Implementações e aplicações)
4. Estrutura de Dados
Filas (implementações, aplicações)
5. Estrutura de Dados
Listas (implementações e aplicações)
Métodos de busca
6. Árvores e métodos de ordenação
Binária, balanceada e não balanceada
n-ária, balanceada e não balanceada
Métodos de ordenação

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 programas/soluções para os problemas propostos. A apresentação da disciplina deve ser permeada por discussões comparativas entre diferentes formas de implementação das estruturas de dados, comparando seus desempenhos computacionais em relação às aplicações propostas. No decorrer das aulas, eventualmente, poderão ser utilizados recursos de interação via internet, tais como fóruns online e assemelhados.

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).A média semestral é calculada de acordo com a Eq. 1 na qual a Pi indica uma prova escrita e MT indica a média ponderada das notas obtidas nos trabalhos práticos. Haverá pelo menos duas avaliações escritas e ao menos um trabalho prático no desenvolvido na disciplina. 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.
 - 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.

Plano de ensino

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

Plano de ensino

Curso: CCI-BAC - Bacharelado em Ciência da Computação

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

Disciplina: LFA0001 - LINGUAGENS FORMAIS E AUTÔMATOS

Período letivo: 2018/2

Carga horária: 72

Professor: 3092798 - CLAUDIO CESAR DE SA

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. Computação, problemas e formalizações.
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 - GR
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
27. Indução Finita em PALAVRAS. Baseado no comprimento das palavras.
28. Revisão
29. Estudo individual
30. Atendimento de alunos
31. Dúvidas Gerais e Revisão
32. Provão final da disciplina
33. Apresentação da disciplina Objetivos Gerais e Específicos Avaliação e datas de provas
34. Exercícios do tópico em questão
35. Dúvidas de alunos sobre o tópico corrente
36. Resolução de Exercícios
37. Discussão e Resolução de Problemas
38. Contextualização da disciplina: formalização de problemas, complexidade e computabilidade -- motivação ao conteúdo
39. Lista de exercícios
40. Correções de provas e exercícios
41. Diversos: seminário, visitas técnicas, viagens, semana acadêmica, etc -- out of topic
42. Correção de exercício e projeto de minimização
43. Uso da disciplina em modelagem de problemas - sensibilização aos alunos sobre o curso
44. Apresentação da disciplina: contexto, avaliação, conteúdos, PROVAS, datas, sistemáticas, horários, salas, labs, condutas, procedimentos, listas de discussão, repositório das transparências, etc
45. Laboratório
46. Trabalhos práticos
47. Apresentação de exercícios e trabalhos
48. Avaliação escrita (Provas ou PROVÃO)
49. Exame Final
50. Estudo/Ensino a distância prevista em Lei - FERRAMENTAS MOOC

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 e P2).
b) Trabalho prático (T)
c) PROVAO FINAL
d) LISTAS DE EXERCÍCIOS
A Média Semestral (MS) será calculada pela fórmula
$$MS = (P1 + P2) \cdot 0,3 + T \cdot 0,2 + Provas \cdot 0,4 + Listas \cdot 0,1$$
$$MS \geq 7,0$$

Plano de ensino

aprovação

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:

Após a publicação das notas pela professora, os alunos têm 07 dias corridos para solicitar a revisão com a professora. Esta revisão será feita na sala da professora, preferencialmente em horário de atendimento aos alunos, ou em um horário do qual a professora 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 - CIÊNCIA DA COMPUTAÇÃO FASE 03U

Disciplina: MEP0003 - METODOLOGIA DA PESQUISA

Período letivo: 2018/2

Carga horária: 36

Professor: 3198057 - CARLA DIACUI MEDEIROS BERKENBROCK

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. Desenvolver habilidades, capacidades e competências relacionadas com a metodologia da pesquisa, de tal forma a auxiliar na produção e na divulgação do conhecimento na área da ciência da computação.

Objetivo específico

1. - Compreender os conceitos e a importância do conhecimento científico e da pesquisa na área de ciência da computação;
- Identificar a relevância da elaboração de um projeto de pesquisa;
- Reconhecer outras formas de produção e divulgação científica;
- Abordar a necessidade e exercício da ética nas pesquisas e nas demais atividades acadêmicas;
- Assimilar as etapas de um projeto de pesquisa, bem como os seus requisitos e estruturas;
- Aplicar a normalização de trabalhos científicos, bem como as técnicas e procedimentos metodológicos;
- Orientar sobre a redação de um projeto de pesquisa, bem como de outros trabalhos científicos;
- Estabelecer as diferenças, particularidades e similaridades entre os diferentes trabalhos científicos;
- Apresentar as bases de periódicos e trabalhos científicos, incluindo os mecanismos e parametrização de buscas;
- Desenvolver e aperfeiçoar a comunicação na área da ciência da computação;
- Incentivar a elaboração e publicação de trabalhos científicos, bem como a participação de trabalhos em eventos da área da ciência da computação.

Conteúdo programático

1. Aula Inaugural (Apresentação da Professora e alunos/ Apresentação do Plano de Ensino/Organização e detalhamento das atividades avaliativas)
2. Pesquisa tecnológica, Ciência e Tecnologia, A ética na pesquisa
3. A pesquisa (Conceitos/As questões (perguntas) de pesquisa e as hipóteses/O problema de pesquisa/A justificativa e os objetivos)
4. Revisão de literatura e formulação do referencial teórico
5. Normatização do documento científico
6. Comunicação científica

Metodologia

1. A disciplina será trabalhada por meio da temática teórico-prática, com ênfase a discussão ao debate em grupo, assim como uma forte carga de leitura e aulas expositivas complementaram o trabalho.

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) Seminários (SEM)
c) Elaboração e apresentação de um artigo (ART)

Média = SEM*0,40 + ART*0,60

Bibliografia básica

1. 1. FACHIN, Odília. Fundamentos de metodologia. 3. ed. São Paulo: Saraiva, 2001. 200 p.

Plano de ensino

2. GIL, Antonio Carlos. Como elaborar projetos de pesquisa. 5. ed. São Paulo: Atlas, 2010. 184 p.

3. MARCONI, Marina de Andrade; LAKATOS, Eva Maria. Fundamentos de metodologia científica. 7. ed. São Paulo: Atlas, 2010. 297 p.

Bibliografia complementar

1. 1. BARROS, Aidil de Jesus Paes de; LEHFELD, Neide Aparecida de Souza. Fundamentos de metodologia científica: um guia para a iniciação científica. 2. ed. ampl. São Paulo: Pearson Education, 2000. 122 p.
2. BASTOS, Cleverson Leite; KELLER, Vicente. Aprendendo a aprender: introdução a metodologia científica. 22. ed. Rio de Janeiro: Vozes, 2008. 111 p.
3. LUZ, A. C. da et al. Manual para elaboração de trabalhos acadêmicos da UDESC: tese, dissertação, trabalho de conclusão de curso e relatório de estágio. Universidade do Estado de Santa Catarina, 4ª ed. Florianópolis: UDESC, 2013. Disponível em: http://www.udesc.br/arquivos/id_submenu/6/manual_a4____abnt.pdf. Acesso em: 20/07/2016.
4. PINHEIRO, J. M. S. Da iniciação científica ao TCC. Uma abordagem para os cursos de tecnologia. Rio de Janeiro: Ed. Ciência Moderna Ltda., 2010.
5. SILVA, E. da; TAFNER, E. P.; FISCHER, J.; MALCON, A. T. Metodologia do trabalho acadêmico. 3. ed. rev. e atual. Curitiba: Juruá Ed., 2010. 131 p.

Plano de ensino

Curso: CCI-BAC - Bacharelado em Ciência da Computação

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

Disciplina: POO0001 - PROGRAMAÇÃO ORIENTADA A OBJETOS

Período letivo: 2018/2

Carga horária: 72

Professor: 211020727 - DIEGO BUCHINGER

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

Conteúdo programático

1. 1. Introdução
1.1. Introdução ao Paradigma Orientado a Objetos
1.2. Introdução a Linguagem de Programação Java
2. 2. Conceitos Básicos de Orientação a Objetos
2.1. Introdução a IDE Netbeans
2.2. Classes, objetos, propriedade static, encapsulamento e método construtor
3. 3. Introdução à Java Parte II
3.1. Wrappers e conversão entre valores
3.2. String, Math e Random
3.3. Arrays - vetores e matrizes
3.4. Manipulação de Arquivos
4. 4. Conceitos Básicos de Orientação a Objetos II
4.1. Enumeradores
4.2. Associação, Agregação e Composição
4.3. Destrutores e Garbage Collector
5. 5. Conceitos de Orientação a Objetos I
5.1. Herança
5.2. Polimorfismo
6. 6. Conceitos de Orientação a Objetos II
6.1. Interface
6.2. Classe Abstrata
7. 7. Introdução à Java Parte III
7.1. Tratamento de Exceções
8. 8. Introdução à Java Parte IV
8.1. Classe Arrays
8.2. Coleções em Java: Collections
9. 9. Introdução à Java Parte V
9.1. Multithreading - Thread e Runnable
10. 10. Introdução à Java Parte V
10.1. Introdução à Interface Gráfica - Swing
10.2. Componentes de interface gráfica - javax.swing
10.3. Gerenciamento de layouts
10.4. Tratamento de eventos
11. 11. Padrões de Projeto
11.1. Padrões criacionais
11.2. Padrões estruturais
11.3. Padrões comportamentais

Plano de ensino

11.4. Padrão de projeto em Camadas

12. 12. Conceitos de Orientação a Objetos III
12.1 Sobrecarga de Operadores (em C++)

Metodologia

1. Aulas expositivas acompanhadas de práticas em laboratório, que objetivam a implementação dos conceitos apresentados nas aulas teóricas. Listas de exercícios para auxiliar na fixação do conteúdo apresentado. Trabalhos práticos relacionados aos conteúdos apresentados na aula e avaliar a capacidade do uso dos conceitos aprendidos. Provas teóricas para avaliar o conteúdo conceitual aprendido.

Sistema de avaliação

1. Os alunos serão avaliados com base no seu desempenho nas provas, trabalhos e 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) dois trabalhos de implementação (40% da média, 20% cada)
 - b) três provas (60% da média, 20% cada).

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 - CIÊNCIA DA COMPUTAÇÃO FASE 03U

Disciplina: PPR0001 - PROJETO DE PROGRAMAS

Período letivo: 2018/2

Carga horária: 36

Professor: 211020122 - JEFERSON LUIZ RODRIGUES SOUZA

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. Fundamentos da Análise e Projeto de Sistemas
 - 1.1 Visão Geral
 - 1.2 Introdução a Engenharia de Software
2. Fundamentos da Análise e Projeto de Sistemas
 - 1.3 Participantes, Ferramentas e o Processo de Desenvolvimento de Software
 - 1.4 Levantamento de Requisitos
3. Fundamentos da Análise e Projeto de Sistemas
 - 1.5 Qualidade do Projeto de Software
4. Projeto de Dados
 - 2.1 Modelo Entidade-Relacionamento
5. Projeto de Dados
 - 2.2 Projeto Arquitetural Básico - Diagrama de Classe
6. Projeto Comportamental
 - 3.1 Diagrama de Fluxo de Dados
7. Projeto Comportamental
 - 3.2 Diagrama de Eventos
 - 3.3 Diagrama de Estados
8. Padrões de Projeto
 - 4.1 Padrões Estruturais
 - 4.1.1 Padrão Singleton
 - 4.1.2 Padrão Façade
9. Padrões de Projeto
 - 4.2 Padrões Arquiteturais
 - 4.2.1 Padrão Model View Controller
 - 4.2.2 Padrão em Camadas
10. Projeto de Interfaces

Metodologia

1. Aulas expositivas com atividades para fixação do conteúdo ministrado. 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:
 - * Provas I e II [50% - 25% cada]
 - * Trabalho Final [50%]

Plano de ensino

Bibliografia básica

1. BEZERRA, Eduardo. Princípios de análise e projeto de sistemas com UML. 2.ed. Rio de Janeiro: Campus, 2007. 369 p. ISBN 8535216960 (Broch.).
PRESSMAN, Roger S. Engenharia de software. 5. ed. Rio de Janeiro: McGraw-Hill, c2001. 843 p. ISBN 8586804258 (broch.).
SOMMERVILLE, Ian. Engenharia de software. 9. ed. São Paulo: Pearson Education, 2013. 529 p. ISBN 8579361087.

Bibliografia complementar

1. BOOCH, Grady; RUMBAUGH, James; JACOBSON, Ivar. UML: guia do usuário. 2. ed. rev. e atual. Rio de Janeiro: Campus, c2006. 474 p. ISBN 8535217843 (broch.).
DENNIS, Alan; WIXOM, Barbara Haley. Análise e projeto de sistemas. 2. ed. Rio de Janeiro: LTC, 2005. 461 p. ISBN 8521614578 (broch.).
GAMMA, Erich et al. Padrões de projeto: soluções reutilizáveis de software orientado a objetos. Porto Alegre: Bookman, 2000. 364 p. ISBN 8573076100 (broch.).
LARMAN, Craig. Utilizando UML e padrões. Porto Alegre: Bookman, 2000. 492 p. ISBN 8573076518 (broch.).
RUMBAUGH, James. Modelagem e projetos baseados em objetos. Rio de Janeiro: Campus, c1994. 652 p. ISBN 857001841X (broch.).

Plano de ensino

Curso: CCI-BAC - Bacharelado em Ciência da Computação

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

Disciplina: AMS0001 - ANÁLISE E MODELAGEM DE SISTEMAS

Período letivo: 2018/2

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 - CIÊNCIA DA COMPUTAÇÃO FASE 04U

Disciplina: BAN1001 - BANCO DE DADOS I

Período letivo: 2018/2

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 - CIÊNCIA DA COMPUTAÇÃO FASE 04U

Disciplina: COM0002 - COMPILADORES

Período letivo: 2018/2

Carga horária: 72

Professor: 9672753 - LEANDRO ISRAEL PINTO

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 e livres de contexto; 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*Prova + 25*Trabalho + 25*Prova Prática + 10*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 - CIÊNCIA DA COMPUTAÇÃO FASE 04U

Disciplina: PRA0001 - PROJETO DE ARQUIVOS

Período letivo: 2018/2

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 mídias de armazenamento permanente.

Objetivo específico

1. Construir códigos eficientes de armazenamento e busca em mídias permanentes.

Conteúdo programático

1. Revisão ordenação básica
2. Revisão OO
3. Programação com clean code
4. Padrões de projeto
5. CRC
6. Hash
7. Espalhamento
8. Busca em String
9. Unidades de armazenamento
10. SAP Hanna
11. Berkeley DB
12. Índices com BTREE

Metodologia

1. Aulas expositivas, práticas e dinâmica de grupo aplicando coding dojo

Sistema de avaliação

1. 1a. Prova - 27/03
2a. Prova - 8/05
3. Trabalho final 19/06

Bibliografia básica

1. DOS SANTOS, C.S.; AZEREDO, P.A. Tabelas: Organizac_ões e Pesquisa. UFRGS, 2001. HOROWITZ, E., SAHNI, S. Fundamentos de Estruturas de Dados. Editora Campus, 1987. ISBN 8570014228.
ZIVIANI, N. Projeto de Algoritmos com Implementac_ões 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 - CIÊNCIA DA COMPUTAÇÃO FASE 04U
Disciplina: REC0001 - REDES DE COMPUTADORES
Período letivo: 2018/2
Carga horária: 72
Professor: 6675298 - GUILHERME PIEGAS KOSLOVSKI

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. Introdução às redes de computadores - Conceitos básicos: comutação de dados, sistema de comunicação, meios de transmissão, protocolos, tempo de propagação e de transmissão, atrasos, tipos de conexão, vazão.
2. Introdução às redes de computadores - Apresentação da evolução das arquiteturas e sistemas: camadas e protocolos, entidade, serviços, interfaces, endereçamento.
3. Introdução às redes de computadores - Funções de um protocolo: controle de erros, controle de fluxo, segmentação e remontagem, endereçamento, multiplexação, encapsulamento
4. Introdução às redes de computadores - Apresentação das topologias de rede
Meios de transmissão
5. Introdução às redes de computadores - Comutação de circuitos e de pacotes
6. Introdução às redes de computadores - Arquiteturas e padrões
 - i. Modelo de referência OSI da ISO
 - ii. Noções Arquitetura TCP/IP
 - iii. Comparação entre modelo OSI e arquitetura TCP/IP
7. Camada de aplicação - Princípios e Serviços
8. Camada de aplicação - Protocolo HTTP
9. Camada de aplicação - Aplicações
10. Camada de aplicação - Arquitetura P2P e arquitetura cliente-servidor
11. Camada de transporte - Princípios e serviços
12. Camada de transporte - Algoritmos Go-Back-N e Selective Repeat
13. Camada de transporte - Comunicação confiável
14. Camada de transporte - Estabelecimento e encerramento de conexões
15. Camada de transporte - Protocolo UDP
16. Camada de transporte - Protocolo TCP
17. Camada de transporte - Princípios de Controle de Congestionamento
18. Camada de rede - Introdução
19. Camada de rede - Comutação
20. Camada de rede - Endereçamento
21. Camada de rede - Roteamento
22. Camada de rede - Protocolo IP

Plano de ensino

23. Camada de rede - Algoritmos de roteamento
24. Camada de rede - Roteamento na Internet
25. Camada de enlace e redes locais - Introdução
26. Camada de enlace e redes locais - Detecção e correção de erros de transmissão
27. Camada de enlace e redes locais - Tipos de Serviços
28. Camada de enlace e redes locais - Protocolos de Acesso Múltiplos
29. Camada de enlace e redes locais - Endereçamento
30. Camada de enlace e redes locais - Ethernet
31. Camada de enlace e redes locais - Equipamentos de redes: hubs e switches
32. Camada de enlace e redes locais - Redes Virtuais
33. Camada de enlace e redes locais - Noções de redes sem fio

Metodologia

1. Aulas Expositivo-Dialogadas. Uso de Laboratório. Desenvolvimento de atividades individuais e em grupo. Ocasionalmente ocorrerá o desenvolvimento de atividades através de um ambiente de auxílio para aprendizado a distância.

Sistema de avaliação

1. - Provas teóricas e/ou práticas;
- Participação em aula;
- Implementações e relatórios;
- Lista de exercícios;
- Trabalhos em grupos.

$$\text{Nota Final} = A1 * 0,35 + A2 * 0,35 + A3 * 0,3$$

A1: Avaliação 01 - Prova escrita. Poderá ser complementada com outras atividades.

A2: Avaliação 02 - Prova escrita. Poderá ser complementada com outras atividades.

A3: Avaliação 03 - Implementação e demonstração de aplicativos elaborados ao longo da disciplina. Apresentação de seminários. Discussão de artigos. Poderá ser complementada com outras atividades.

Cada avaliação poderá ser decomposta em um subconjunto de avaliações com pesos ponderados de acordo com sua complexidade.

Bibliografia básica

1. KUROSE, J. & ROSS, K. Computer Networking: A Top-Down Approach Featuring the Internet. Addison- Wesley, 2010, quinta edição.
TANENBAUM, Andrew. Redes de Computadores. 5a. 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.

Bibliografia complementar

1. FOROUZAN, Behrouz, MOSHARRAF, Firouz, Redes de Computadores. Uma abordagem Top-Down, McGraw Hill, 2012
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>
Artigos da área.

Plano de ensino

Curso: CCI-BAC - Bacharelado em Ciência da Computação

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

Disciplina: SOFT001 - ENGENHARIA DE SOFTWARE

Período letivo: 2018/2

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 (SWEBOK). 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 - CIÊNCIA DA COMPUTAÇÃO FASE 04U

Disciplina: SOP0001 - SISTEMAS OPERACIONAIS

Período letivo: 2018/2

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 processador (processos e threads) de um SO;
- Compreender os princípios de programação concorrente, incluindo exclusão mútua e sincronização de processos e threads;
- 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. Fundamentos de SO
3. Histórico de SO
4. Tipos de SO
5. 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
6. Princípios de hardware
Revisão de conceitos básicos de hardware do ponto de vista de um SO
7. Fundamentos de programação Assembly
8. Organização de SO
Organização interna de SO
Arquiteturas monolíticas, em camadas, máquinas virtuais, cliente-servidor, etc.
9. Gerência de processador: conceitos básicos
10. Processos
Conceito de processo
Criação e encerramento de processos
Diagrama de estados de processos
Blocos de controle de processo
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

Plano de ensino

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. Gerência de memória: conceitos básicos Hierarquia de memória Memória lógica vs memória física Organização da memória lógica
22. Monoprogramação e endereçamento absoluto Relocação de memória
23. Gerência de memória contígua Gerência de memória com partições fixas Gerência de memória com partições variáveis Swapping Gerência de espaço livre Algoritmos para alocação de memória física
24. Memória virtual Paginação Tabelas de páginas Memória associativa (TLB)
25. Algoritmos de substituição de páginas
26. Paginação: questões de projeto e implementação Paginação por demanda Localidade, conjunto de trabalho, thrashing Políticas de alocação de páginas Compartilhamento de memória Arquivos e partições de troca
27. Segmentação pura Segmentação paginada
28. Gerência de memória no Linux
29. Sistemas de arquivos: conceitos de arquivos e diretórios
30. Implementação de sistemas de arquivos Organização de um sistema de arquivos Implementação de arquivos Implementação de diretórios Gerência de espaço livre
31. Sistemas de arquivos: tópicos adicionais Arquivos compartilhados Consistência de sistemas de arquivos Sistemas de arquivos virtuais Cache de blocos Leitura antecipada de blocos Redução do movimento do braço do disco Desfragmentação
32. Sistemas de arquivos no Linux
33. Princípios de hardware de E/S
34. Princípios de software de E/S Camadas de software de E/S

Plano de ensino

35. Discos magnéticos
Princípios de funcionamento
Tempos de acesso a disco
Algoritmos de escalonamento de disco

36. Gerência de E/S no Linux
Princípios de gerência de E/S no Linux
Escalonamento de disco no Linux

37. Deadlocks
Conceitos de deadlocks
Modelagem de deadlocks
Tratamento de deadlocks

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) e um trabalho (T). Ocasionalmente, exercícios propostos em aula poderão ser avaliados e compor a nota da prova que cobre o assunto correspondente.

Média final = $0,2 \cdot P1 + 0,3 \cdot P2 + 0,3 \cdot P3 + 0,2 \cdot T$

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 - CIÊNCIA DA COMPUTAÇÃO FASE 04U

Disciplina: TEG0001 - TEORIA DOS GRAFOS

Período letivo: 2018/2

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
referências 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

<p>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</p>
<p>5. Aula_05: Representação Computacional de Grafos Representação Computacional de grafos Matrizes adjacentes Trabalho prático em equipe na aula: implementação de grafos utilizando matrizes</p>
<p>6. aula_06: Representação Computacional de grafos representação computacional de grafos Estrutura de Listas Trabalho prático em grupo em sala: Implementar grafos utilizando Listas</p>
<p>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</p>
<p>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</p>
<p>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</p>
<p>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</p>
<p>11. Aula_11: Primeira avaliação TEG0001: primeira avaliação escrita. Duração 01:40 minutos</p>
<p>12. Aula_12: Entrega primeira prova e coloração Entrega e correção da primeira avaliação</p>
<p>13. Aula 13: Árvores Introdução arvores enraizadas árvores disjuntas grau de uma árvore vértices internos árvore cheia</p>

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 Huffmann 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
39. Projeto da disciplina de TEG0001
40. Apresentação dos projetos da disciplina TEG0001

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:

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

Plano de ensino

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

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 alli. Aspectos Teóricos da Computação, Parte C: Teoria dos Grafos, projeto Euclides, 1979.
- SANTOS, J. P. O. et alli. 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.) FEOFILOFF, 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 - CIÊNCIA DA COMPUTAÇÃO FASE 05U

Disciplina: BAN2001 - BANCO DE DADOS II

Período letivo: 2018/2

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. Curso de SQL On-line (a distância)
33. Prova 1
34. Prova 2
35. Apresentação Trabalho Final
36. Revisão para Prova 1
37. Revisão para Prova 2
38. Semana da Computação
39. Apresentação de Seminário
40. Banco de Dados Chave-Valor - Redis
41. Prática - Consultas no Redis
42. Banco de Dados Documento - MongoDB
43. Prática - Consultas no MongoDB
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 do 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 - CIÊNCIA DA COMPUTAÇÃO FASE 05U

Disciplina: CAL0001 - COMPLEXIDADE DE ALGORITMOS

Período letivo: 2018/2

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 (limite superior)
 - Ordens de complexidade
 - Análise de complexidade com uma variável
3. Conceitos Básicos de Complexidade:
 - Análise de complexidade de tempo de algoritmos recursivos
4. Conceitos Básicos de Complexidade:
 - Somatórios
5. Conceitos Básicos de Complexidade:
 - Complexidade de Espaço
 - Notações Assintóticas (limite inferior e limite firme)
 - Teorema Mestre
6. Análise de Algoritmos de Ordenação:
 - Merge Sort, Quick Sort e Heap Sort
7. Análise de Algoritmos de Ordenação:
 - Counting Sort e Bucket Sort
8. Análise de Complexidade envolvendo Estruturas de Dados Elementares
9. Análise de Complexidade com múltiplas variáveis:
 - Strings
 - Hash
10. Análise de Complexidade com múltiplas variáveis:
 - Grafos (algoritmos tradicionais)
11. Análise de Complexidade com múltiplas variáveis:
 - Operações elementares com inteiros grandes
12. Complexidade em Teoria dos Números e Criptografia:
 - números primos
 - aritmética modular
13. Complexidade em Teoria dos Números e Criptografia:
 - algoritmo de criptografia RSA
14. Abordagens para Resolução de Problemas:
 - Indução matemática
 - Divisão e conquista
 - Algoritmos gulosos
 - Algoritmos de tentativa e erro
15. Abordagens para Resolução de Problemas:
 - Programação dinâmica

Plano de ensino

- Algoritmos de aproximação e heurísticas
16. Vínculo entre Complexidade e Teoria da Computação: - Problemas tratáveis e intratáveis; - Classes de problemas: P, NP, NP-Completo e NP-Difícil; - Redução de problemas;
17. Vínculo entre Complexidade e Teoria da Computação: - Problemas NP-Completo: SAT, 3-CNF-SAT, Clique, Cobertura de Vértices
18. Vínculo entre Complexidade e Teoria da Computação: - Problemas NP-Completo: 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

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 nos seguintes critérios: a) Provas: 55% a.1) Prova 1 (20%) a.2) Prova 2 (20%) a.3) Prova 3 (15%) b) Trabalhos: 45% b.1) Comparação entre algoritmos de ordenação (10%) b.2) Trabalho comparativo entre busca sequencial, binária e hash (10%) b.3) Pesquisa e apresentação de complexidade de algoritmo (10%) b.4) Implementação e análise de algoritmos para o problema do caixeiro-viajante (15%)
--

Bibliografia básica

1. CORMEN, Thomas H. Algoritmos: teoria e prática. 3. ed. Rio de Janeiro: Elsevier, 2012. CORMEN, Thomas H. Desmistificando algoritmos. Rio de Janeiro: Elsevier, 2014. DASGUPTA, Sanjoy; PAPADIMITRIOU, Christos H; VAZIRANI, Umesh Virkumar. Algoritmos. São Paulo: McGraw-Hill, 2009.
--

Bibliografia complementar

1. AHO, Alfred V; HOPCROFT, John E.; ULLMAN, Jeffrey D.; Data structures and algorithms. Reading, MA: Addison Wesley, 1987. KNUTH, Donald Ervin. The art of computer programming - Fundamental Algorithms. 3rd ed. Massachusetts: Addison Wesley, 2008. LEVITIN, Anany. Introduction to the design & analysis of algorithms. 2nd ed. Pearson/Addison Wesley, 2007. TOSCANI, Laura Vieira; VELOSO, Paulo A. S. Complexidade de algoritmos. 3.ed. Porto Alegre: Bookman, 2012. ZIVIANI, Nivio. Projeto de algoritmos: com implementações em Pascal e C. 3. ed. rev. e ampl. São Paulo: Pioneira Thomson Learning, 2011.

Plano de ensino

Curso: CCI-BAC - Bacharelado em Ciência da Computação

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

Disciplina: CGR0001 - COMPUTAÇÃO GRÁFICA

Período letivo: 2018/2

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;
$$\text{Nota Final} = P1 * 0.3 + P2 * 0.3 + TC * 0.2 + TF * 0.2$$

onde:
P1 - Prova 1
P2 - Prova 2
TC - Trabalhos complementares [Exercícios, Trabalhos (6 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 - CIÊNCIA DA COMPUTAÇÃO FASE 05U

Disciplina: PAP0002 - PARADIGMAS DE PROGRAMAÇÃO

Período letivo: 2018/2

Carga horária: 72

Professor: 1033182135 - WESLEY GONCALVES SILVA

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. Estudar as características dos principais modelos de programação quanto a estruturação, interação e a relação entre as linguagens de programação. Estudar os tipos de técnicas de implementação utilizados para o desenvolvimento de programas e solução de problemas.

Objetivo específico

1. Estudar os custos de implementação;
Comparar técnicas entre linguagens de programação;
Estudar as estruturas conceituais para resolver problemas;
Entender alguns dos princípios conflitos entre os recursos de linguagens.

Conteúdo programático

1. Introdução à disciplina:
 - * os paradigmas de programação
 - * as diferenças entre eles
 - * características para avaliação de uma linguagemMotivação para estudar.
2. Tipos de dados e seu gerenciamento na memória nos paradigmas imperativo e orientado a objetos:
 - * Análise com a linguagem C/C++
 - * Análise com a linguagem Python
3. Polimorfismo e funções nos paradigmas imperativo:
 - * Análise com a linguagem C
 - * Análise com a linguagem Python
4. Abstração de dados e modularidade
5. Programação orientada a objetos, polimorfismo e funções
 - * Análise com a linguagem C++
 - * Análise com a linguagem Python
6. Python: estudo de implementação de servidor na nuvem
7. Introdução a Haskell:
 - * conceitos básicos da linguagem
 - * tipos de dados e seu gerenciamento na memória
 - * estruturas
 - * polimorfismo e funções
 - * abstração e modularidade

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 práticos de forma a provocar discussões acerca dos conteúdos e produzir implementações (programação) de soluções para os problemas propostos. Propiciando aos discentes: análise, interpretação e construção de novos conhecimentos com apoio das realizações de pesquisas bibliográficas.

Sistema de avaliação

1. Avaliações escritas: P1 (peso 0,2) e P2 (peso 0,3).
Trabalhos individuais e em equipe: T1 (peso 0,2) e T2 (peso 0,2).
Participação em sala de aula e exercícios: EX (0,1).

Plano de ensino

Bibliografia básica

1. MICHELL, JOHH C. Concepts in Programming Languages. Cambridge University Press, 2003.
SEBESTA, R.W. Conceitos de Linguagens de Programação. 5a ed. Editora Bookman. 2003.

Bibliografia complementar

1. <https://docs.python.org/3/>
<https://www.haskell.org/tutorial/>

Plano de ensino

Curso: CCI-BAC - Bacharelado em Ciência da Computação

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

Disciplina: PES0001 - PESQUISA OPERACIONAL

Período letivo: 2018/2

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

$$\text{Média} = (Px + Py) / 2$$

onde Px e Py são as duas maiores de P1, P2 e 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 - CIÊNCIA DA COMPUTAÇÃO FASE 05U

Disciplina: SDI0001 - SISTEMAS DISTRIBUIDOS

Período letivo: 2018/2

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. Apresentação da disciplina
2. Caracterização de sistemas distribuídos
3. Modelos de sistemas
4. Comunicação entre processos
5. Chamada de procedimento remoto (RPC)
6. Objetos distribuídos e invocação remota (RMI)
7. Web Services
8. Comunicação indireta
9. Sistemas Operacionais: de rede e distribuído
10. Sistemas de arquivos distribuídos
11. Sistemas P2P
12. Tempo e estado global
13. Coordenação e acordos
14. Replicação
15. Sistema de multimídia distribuído
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 - CIÊNCIA DA COMPUTAÇÃO FASE 05U

Disciplina: TEC0001 - TEORIA DA COMPUTAÇÃO

Período letivo: 2018/2

Carga horária: 72

Professor: 3949795 - CRISTIANO DAMIANI VASCONCELLOS

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 decisores.
4. Máquinas de Turing com múltiplas fitas.
5. Máquinas de Turing não determinísticas.
6. Enumeração.
7. Decidibilidade de linguagens.
8. Redução de Problemas.
9. Problemas indecidíveis.
10. Notação assintótica e análise de algoritmos.
11. Classes de Problemas (P, NP e NP-Completo).
Teorema de Cook-Levin.

Metodologia

1. A disciplina será desenvolvida através de aulas expositivo-dialogadas. 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) Provas individuais (P1 e P2);
b) Trabalho final escrito (T);
c) Exercícios (E).
A Média Final (MF) será calculada pela fórmula
$$MF = 0,25 * P1 + 0,25 * P2 + 0,2 * T + 0,3 * E$$

Plano de ensino

Bibliografia básica

1. 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

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

VIEIRA, Newton José. Introdução aos fundamentos da computação: linguagem e máquinas. São Paulo: Thomson, 2006. 319 p. ISBN 8522105081

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

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>

Plano de ensino

Curso: CCI-BAC - Bacharelado em Ciência da Computação

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

Disciplina: ACT0001 - AUTOMAÇÃO E CONTROLE

Período letivo: 2018/2

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, Máquinas para Manufatura Aditiva.
6. 6 SISTEMAS DE CONTROLE
 - Em malha aberta
 - Em malha fechada
 - Supervisórios
7. 7 Integração e Controle.
 - Arquiteturas
 - Integração da Manufatura
 - Tópicos sobre integração e a "Quarta Revolução Industrial"

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 - CIÊNCIA DA COMPUTAÇÃO FASE 06U

Disciplina: EMI0001 - EMPREENDEDORISMO EM INFORMÁTICA

Período letivo: 2018/2

Carga horária: 36

Professor: 1033140358 - GLAUCO VINICIUS SCHEFFEL

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. Gerar a compreensão do que é empreender. E a importância de praticar as habilidades e atitudes de identificar necessidades e supri-las.

Objetivo específico

1. Compreender os comportamentos necessários para empreender;
2. Identificar e suprir necessidades de mercado;
3. Provar a validade de uma ideia;
4. Entender e usar as estratégias gerenciais do empreendedorismo tecnológico;

Conteúdo programático

1. O que são Startups
2. Processos e autores (lean startup)
3. Business model canvas
4. Geração de ideias
5. Prova de conceito
6. Avanço da ideia
7. Tipos de investimento
8. Contratos
9. Aceleradoras e incubadoras

Metodologia

1. Aulas práticas e dialogadas com o desenvolvimento de negócios

Sistema de avaliação

1. Uma prova (30/03) e dois trabalhos (11/05 e 22/06) todas notas com mesmo peso.

Bibliografia básica

1. Blank, Steve. The four steps to the epiphany: successful strategies for products that win. BookBaby, 2013.
Ries, Eric. Lean Startup: Schnell, risikolos und erfolgreich Unternehmen gründen. Redline Wirtschaft, 2014.
Eisenmann, Thomas R., Eric Ries, and Sarah Dillard. "Hypothesis-driven entrepreneurship: The lean startup." (2012).
Pesce, Bel. A menina do vale. Leya, 2012.
Croll, Alistair, and Benjamin Yoskovitz. Lean analytics: Use data to build a better startup faster. " O'Reilly Media, Inc.", 2013.

Bibliografia complementar

1. Osterwalder, Alexander, and Yves Pigneur. Business model generation: a handbook for visionaries, game changers, and challengers. John Wiley & Sons, 2010.

Plano de ensino

Curso: CCI-BAC - Bacharelado em Ciência da Computação

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

Disciplina: IAR0001 - INTELIGÊNCIA ARTIFICIAL

Período letivo: 2018/2

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:
 $PROVA\#1 \times 0,30 + PROVA\#2 \times 0,30 + TRABALHOS \times 0,25 + SEMINÁRIO \times 0,15$

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 - CIÊNCIA DA COMPUTAÇÃO FASE 06U

Disciplina: IHC0001 - INTERAÇÃO HOMEM COMPUTADOR

Período letivo: 2018/2

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

<p>tipos de concepção</p> <p>usabilidade como requisito do sistema</p> <p>questões de concepção</p> <p>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</p>
<p>7. Interfaces WIMP x Web</p> <p>Introdução</p> <p>Diferenças e similaridade</p>
<p>8. Processo de Webdesign</p> <p>Introdução</p> <p>Processos de desenvolvimento</p> <p>Etapas: Definição e planejamento, arquitetura, projeto dos componentes, desenvolvimento</p> <p>Elementos do conteúdo</p> <p>Princípios de diretrizes, padrões e guias (por exemplo as regras de ouro, usabilidade na web do Nielsen, guias de usabilidade e acessibilidade)</p> <p>Fundamentos para o projeto de Interfaces Web</p> <ul style="list-style-type: none"> - 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
<p>9. Avaliação de Interface</p> <ul style="list-style-type: none"> - 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
<p>10. Acessibilidade</p> <p>Introdução, conceitos, diretrizes e checklist, avaliação automática sistemas</p>
<p>11. Tópicos de inovação e tendências</p> <p>Tópicos de Inovações e Tendências na área de IHC, tais como:</p> <p>User experience</p> <p>Sistemas adaptativos, Modelo do Usuário e Adaptabilidade/personalização</p> <p>Sistemas cientes/sensíveis ao contexto (context-aware systems)</p> <p>Sistemas/interfaces móveis e/ou ubíquos, Responsive webdesign</p> <p>Sistemas colaborativos</p> <p>interação humano-robô</p> <p>Questões culturais, Dimensões culturais - internacionalização e localização de interfaces</p> <p>Sistemas de Recomendação</p> <p>Visualização da informação</p> <p>Engenharia baseada em cenários</p> <p>Web responsiva</p> <p>Funology, gamification, jogos</p> <p>Acessibilidade</p> <p>Realidade virtual e aumentada, 3D, Interação 3D</p> <p>Ambientes EAD e IHC</p> <p>Interação natural, tangível, vestíveis, modal e multitouch</p> <p>Web semântica e ontologia</p> <p>Emoção em IHC</p> <p>Experimentos -com métricas estatística em IHC</p> <p>Qualidade</p> <p>Novas tendências</p> <p>*Tópicos a serem escolhidos conforme tendências e projetos inovadores na área.</p>
<p>12. Prova da disciplina</p>
<p>13. Apresentação Trabalho em sala</p>
<p>14. Palestra</p> <p>Palestras Oportunas com profissionais da área</p>
<p>15. Desenvolvimento do trabalho</p> <p>Aplicação do Aluno</p>

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; ABOARD, 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 - CIÊNCIA DA COMPUTAÇÃO FASE 06U

Disciplina: MFO0001 - MÉTODOS FORMAIS

Período letivo: 2018/2

Carga horária: 72

Professor: 210720047 - RAFAEL ALCESTE BERRI

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. Conhecimento dos principais métodos formais e semiformais empregados.

Objetivo específico

- 1) Conceituar especificação formal
- 2) Estudar as categorias de especificações semiformais
- 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. 2. Visão Geral, definição e motivação à Métodos formais
3. 3. Matemática discreta e lógica de primeira ordem
4. 4. Especificação de software: O elo fraco em modelagem de sistemas
4.1 Exercício prático de especificação de software
5. 5. Métodos Semiformais
5.1 Categorias de linguagens Semiformais
5.2 Avaliação de métodos Semiformais
5.3 Ferramentas automatizadas
6. 5.4 Exercício prático: projetando um software com UML
7. 6. Classificação de modelos formais.
6.1 Contextualização
8. 7. Redes de Petri
7.1 Introdução informal de Redes de Petri
7.1.1 Exercícios de especificação utilizando Redes de Petri
9. 7.2 Introdução formal à Redes de Petri
7.2.1 Exercícios de especificação formal utilizando Redes de Petri.
10. 7.3 Programando em Redes de Petri
7.4 Conceitos
7.4.1 Paralelismo
7.4.2 Conflito
7.4.3 Confusão
7.4.4 Exclusão mútua
11. 7.5 Análise de redes de petri
7.5.1 Propriedades dependentes da marcação (vivacidade, limitação, reiniciação)
7.5.2 Propriedades estruturais (invariantes de lugar, invariantes de transição)
12. 7.6 Verificação formal de software utilizando Redes de Petri.
7.6.1 Verificação com Redes de Petri - Enumeração e Estrutural
7.6.2 Verificação com Redes de Petri - Redução
7.6.3 Verificação com Redes de Petri - Ideias Finais

Plano de ensino

13. 7.7 Validação formal Redes de Petri utilizando Model-Checking.

14. 7.8 Estudo de outros tipo de redes de petri.

15. 8. Exercícios Finais

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. 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 (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 alunos serão avaliados com base no seu desempenho nas provas, trabalho final e trabalhos complementares, sendo que ao final do semestre o aluno deverá ter comparecimento mínimo de 75% às aulas e desempenho mínimo de 70% nas avaliações para não necessitar do exame.

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

$$\text{NotaFinal} = 0.3P1 + 0.3P2 + 0.3TF + 0.1TC$$

P1 - Prova 1

P2 - Prova 2

TF - Trabalho Final

TC - Trabalhos complementares (Exercícios, Trabalhos, entre outros)

Bibliografia básica

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

Bibliografia complementar

1. WOODCOCK, J. & LOOMES, M. Software Enginnering 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 - CIÊNCIA DA COMPUTAÇÃO FASE 07U

Disciplina: PIM0001 - PROCESSAMENTO DE IMAGENS

Período letivo: 2018/2

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.

Conteúdo programático

1. 1. Apresentação da Disciplina: 1.2. Contexto na grade do BCC 1.3. Avaliações 1.4. Ambientes de programação a serem utilizados 1.4. Bibliografia
2. 2. Fundamentos de imagens digitais: 2.1. Elementos da percepção visual 2.2. Sensores e aquisição de imagens participação em classe, frequência, 2.3. Representação de imagem 2.4. Amostragem, quantização, digitalização 2.5. Relacionamentos básicos entre pixels 2.6. Operações matemáticas aplicadas ao processamento de imagens 2.7. Métricas de qualidade de imagens 2.8. Transformadas de imagem
3. 3. Transformações de Imagens: 3.1. Fundamentos das transformações de imagem 3.2. Domínios: Espaço e Frequência 3.2. Exemplos ilustrativos de transformações
4. 4. Realce e restauração: 4.1. Transformações do Histograma 4.2. Filtragem no Domínio Espacial 4.3. Filtragem no Domínio Frequência 4.4. Degradação de imagens e Técnicas de Restauração
5. 5. Morfologia Matemática: 5.1. Fundamentos 5.2. Operadores morfológicos em imagens binárias e aplicações 5.3. Operadores morfológicos em imagens monocromáticas e aplicações
6. 6. Segmentação: 6.1. Detecção de Descontinuidades 6.2. Detecção de Bordas 6.3. Limiarização 6.4. Segmentação Orientada a Regiões
7. 7. Representação e Descrição: 7.1. Esquemas de Representação 7.2. Descritores
8. 8. Reconhecimento e Interpretação 8.1. Padrões e Classes 8.2. Reconhecimento de Padrões em Imagens
9. 9. Compressão de imagens 9.1. Sistemas de compressão e codificação 9.2. Padrões de compressão

Metodologia

1. Aulas expositivas acompanhadas de atividades teóricas e práticas relacionados aos conteúdos apresentados em sala. Até no máximo 20% da carga horária da disciplina

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). A média semestral é calculada pela média ponderada das notas obtidas, em pelo menos duas avaliações, envolvendo aspectos teóricos e/ou práticos desenvolvidos na disciplina. 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: d) não é permitido o uso de aparelhos de comunicação (celulares e similares) durante as provas.

Plano de ensino

Bibliografia básica

1. 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'GORMAN, 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

1. 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. ISBN9788576054016.
GONZALEZ, Rafael C.; WOODS, Richard E. Digital Image Processing. 3rd ed. Prentice Hall, 2007. ISBN013168728X.
KIUSALAAS, Jaan. Numerical Methods in Engineering with Python. Cambridge University Press, 2005. ISBN0521852870.
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. ISBN0133361659.
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.

Plano de ensino

Curso: CCI-BAC - Bacharelado em Ciência da Computação
Turma: CCI122-07U - CIÊNCIA DA COMPUTAÇÃO FASE 07U
Disciplina: TCC1003 - TRABALHO DE CONCLUSÃO DE CURSO I
Período letivo: 2018/2
Carga horária: 36
Professor: 3639428 - EVERLIN FIGHERA COSTA MARQUES

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

Objetivo geral

Objetivo específico

Conteúdo programático

Metodologia
1. Entrega de plano de trabalho. Avaliação da proposta por banca (VALIDO/INVALIDO - com ajustes) Desenvolvimento de trabalho com orientador. Entrega de monografia. Apresentação de monografia.

Sistema de avaliação
1. Defesa de monografia com avaliação de banca.

Bibliografia básica

Bibliografia complementar

Plano de ensino

Curso: CCI-BAC - Bacharelado em Ciência da Computação

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

Disciplina: ETI0001 - ÉTICA EM INFORMÁTICA

Período letivo: 2018/2

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. 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.
2. 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.
2.6. Ética da convicção e ética da responsabilidade.
2.7. Em busca de uma teoria ética unificadora.
3. 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.
3.3. Valores.
3.4. Avaliação moral.
4. 4. Realização da moral: A) Ética em sua vertente individual.
4.1. Princípios básicos.

Plano de ensino

4.2. Virtudes. 4.3. Critérios de tomada de decisão. 4.4. Processo de tomada de decisão. 4.5. Princípio do efeito duplo. 4.6. Colaboração com o mal.
5. 5. Realização da moral: B) Ética social. 5.1. Conceito, natureza e origem da sociedade. 5.1.1. Instituição natural. 5.1.2. Contratualismo. 5.2. Fundamentos da sociedade. 5.2.1. A questão da autoridade e do poder. 5.2.2. Nação, estado e sociedade. 5.3. Princípios da ética social. 5.3.1. Direitos humanos. 5.3.2. Bem comum. 5.3.3. Solidariedade. 5.3.4. Subsidiariedade. 5.4. Realização moral como empreendimento coletivo.
6. 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. 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?
7. 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. 7.9.1. Questões referentes à privacidade. 7.9.2. Questões de roubo e pirataria. 7.10. Ética, informática e legalidade. 7.10.1. Marco civil da internet. 7.10.2. Lei nº 12.737. 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 1,0 ponto por dita de atraso até a semana subsequente em que já não serão mais aceitos. Faltas em

Plano de ensino

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 ponto descontado da nota de seu seminário.

PESOS DAS AVALIAÇÕES:

[(P1) (25%)] + [(P2) (25%)] + [(Debates Coordenados) (50%)]

17/10/2018 - PROVA 1

21/11/2018 - AVALIAÇÃO DOS SEMINÁRIOS

28/11/2018 - PROVA 2

12/11/2018 - EXAME FINAL

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.
SPINELLO, Richard A. Case studies in information and computer ethics. Upper Saddle River, NJ: Prentice-Hall, c1997.
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 - CIÊNCIA DA COMPUTAÇÃO FASE 08U

Disciplina: OACO001 - ANIMAÇÃO POR COMPUTADOR

Período letivo: 2018/2

Carga horária: 72

Professor: 6651070 - ANDRE TAVARES 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. CONCEITUAR animação computadorizada e seu histórico.
INTRODUZIR conceitos básicos, aspectos e técnicas de construção de objetos 2D e 3D.
CONCEITUAR transformações geométricas (translação, rotação escala).
INTRODUZIR conceitos avançados de animação computadorizada (hierárquica, baseada em física, interativas).
DESENVOLVER protótipos de sistemas de animação computadorizada.

Conteúdo programático

1. 1. Introdução
1.1. Conceitos básicos
2. 1.2. Revisão sobre conceitos matemáticos
3. 1.3. Etapas da Animação Computacional
4. 2. Formas de Animação
2.1. Keyframing
5. 2.2. Controle Automático de Câmera
6. 2.3. Animação Hierárquica
2.3.1. Cinemática Direta
2.3.2. Cinemática Inversa
7. 2.4. Corpos Rígidos
2.4.1. Restrições Físicas
2.5. Corpos Flexíveis
2.5.1. Massa-mola
8. 3. Controle de Grupos de Objetos
3.1. Sistemas de Partículas
9. 3.2. Bandos (Flocking)
3.3. Comportamento Autônomo
10. 4 Aplicações Práticas
4.1. Animações baseadas em linha de tempo
4.2. Animações baseadas em histórias
4.3. Animações baseadas em personagens

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:
- Prova individual (P1) correspondendo a 30% da nota final;

Plano de ensino

- vários exercícios extra-classe (TC - Trabalhos Complementares) serão desenvolvidos individualmente pelos acadêmicos, sendo a média desses correspondente a 40% da nota final;
- O projeto final (PF) mediante descrição do projeto e defesa: 30%
A média geral, M, portanto, será obtida por meio da seguinte fórmula:
$$M = 3,3P1 + 0,4TC + 0,3PF$$

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.
HAWKINS, K. e ASTLE, D., "OpenGL Game Programming", Prima Tech, 2001.
ANGEL, Edward. Interactive Computer Graphics: a top-down approach with OpenGL. 2.ed. Reading: Addison-Wesley, 2000.
FOLEY, J. et al. Computer Graphics: Principles and Practice. 2. Ed. Reading: Addison-Wesley, 1990.
HEARN, Donald & BAKER, Pauline. Computer Graphics - C Version. 2 ed. Prentice Hall, 1997.
WATT, Alan. 3D Computer Graphics. Addison Wesley, 1993.
WRIGHT, R. S.; SWEET, M. OpenGL SuperBible. 2. ed. Indianapolis: Waikie Group, 2000.

Plano de ensino

Curso: CCI-BAC - Bacharelado em Ciência da Computação

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

Disciplina: ODAW001 - DESENVOLVIMENTO DE APLICAÇÕES NA WEB

Período letivo: 2018/2

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, etc.
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;
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

Plano de ensino

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.

Bibliografia básica

1. DEITEL, H.M.; Deitel, P.J.; Nieto, T.R. Internet & World Wide Web Como Programar, 2a Edição Bookman, 2003.
CLARK, Richard; STUDHOLME, Oil; MURPHY, Chistopher. Introdução ao Html5 e Css3: A Evolução da Web. Rio de Janeiro: Alta Books, 2014.
LOCKHART, Josh. PHP Moderno: Novos recursos e boas práticas. São Paulo: Novatec, 2015.
MARCON, Antonio Marcos; NEVES, Denise. Aplicações e banco de dados para internet. 2.ed. São Paulo: Livros Erica, 2000.

Bibliografia complementar

1. CASTRO, Elizabeth; HYSLOP, Bruce. Html5 e Css3 - Guia Prático e Visual, 7ª edição. Rio de Janeiro: Alta Books, 2013.
FLANAGAN, David. JavaScript: o guia definitivo. 6ª edição. Porto Alegre: Bookman, 2013.
SILVA, Maurício Samy. Web Design Responsivo - Aprenda a criar Sites que se adaptam automaticamente a qualquer dispositivo. São Paulo: Novatec, 2014.
ANSELMO, Fernando. PHP e MySQL: maior, melhor e totalmente sem cortes. Florianópolis: Visual Books, 2002
GOODMAN, Danny. JavaScript: a bíblia. Rio de Janeiro: Campus, 2001.
KURNIAWAN, Budi. Java para a Web com Servlets, JSP e EJB. Rio de Janeiro: Ciência Moderna, c2002.
ARTIGOS diversos de Periódicos e conferências com temas atuais.

Plano de ensino

Curso: CCI-BAC - Bacharelado em Ciência da Computação

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

Disciplina: OEAD001 - EDUCAÇÃO À DISTÂNCIA

Período letivo: 2018/2

Carga horária: 72

Professor: 3461858 - AVANILDE KEMCZINSKI

Ementa

1. Fundamentos da Educação a Distância (EAD). Tecnologias Educacionais. Requisitos de Software Educativo. E-learning. Cooperação e Colaboração.

Objetivo geral

1. Ao término da disciplina o aluno será capaz de caracterizar os fundamentos tratados na Educação a Distância, Tecnologias Educacionais e Requisitos de Software Educativo. Atuar no processo de soluções de problemas relacionados a especificação e avaliação de Ambientes E-learning. Assim como a especificação de recursos computacionais de cooperação e colaboração em Projetos de Ambientes E-learning.

Objetivo específico

1. Identificar os diferentes tipos possíveis de uso da tecnologia na Educação a Distância;
Identificar os fatores de qualidade desejáveis nos ambientes e-learning projetados para o uso educacional a partir de uma perspectiva técnica e pedagógica;
Tomar contato com os principais conceitos relativos a requisitos de software educacional já implementadas em artefatos de software no estágio atual de desenvolvimento da área de informática educacional;
Elaborar uma metodologia adequada ao uso da tecnologia como instrumento de planejamento e ferramenta educacional na modalidade a distância;
Avaliar ferramentas e propor projetos de software educacional (ambientes e-learning);
Avaliar ferramentas e propor projetos de software educacional com foco em ambientes colaborativos e/ou cooperativos.

Conteúdo programático

1. Aula Expositiva (Professor)
10-1.1 Apresentação da temática pelo professor
10-1.2 Dinâmica de grupo para discussão do tema
10-1.3 Análise dos resultados
2. Plano de Ensino da Disciplina
20-1.1 Apresentar e discutir objetivo geral, ementa e objetivos específicos
20-1.2 Apresentar conteúdo programático
20-1.3 Apresentar metodologia pedagógica e método de avaliação
20-1.4 Apresentar as referências Bibliográficas da disciplina
3. Instrumentalização da Disciplina
30-1.1 Solicitar inscrição dos alunos na lista da disciplina
30-1.2 Efetuar inscrição no ambiente
30-1.3 Apresentar as ferramentas a serem utilizadas na disciplina
4. Avaliação Individual
40-1.1 Avaliação teórica Individual (Prova)
5. Trabalhos em Grupos (Alunos)
50-1.1 Execução de atividades
50-1.2 Compilação da resultados
50-1.3 Geração de relatórios
6. Seminários Processo Ensino-Aprendizagem (Alunos)
60-1.1 Apresentação do tema
60-1.2 Discussões do tema
60-1.3 Avaliação do seminário
7. Seminários Avaliação Ambiente E-learning (Alunos)
65-1.1 Apresentação do tema
65-1.2 Discussão do tema
65-1.3 Avaliação do Seminário
8. Semana da Computação
70-1.1 Palestras
70-1.2 Mini-cursos
70-1.3 Oficinas
70-1.4 Visitas Técnicas

Plano de ensino

9. Fundamentos da Educação a Distância 80-1.1 Histórico 80-1.2 Modalidade 80-1.3 Geração 80-1.4 Estado da Arte
10. Tecnologias Educacionais 90-1.1 Computador Ferramenta 90-1.2 Internet e Educação 90-1.3 Ensino a Distância 90-1.4 Computador Tutor 90-1.5 Computador Tutelado 90-1.6 Sistemas de Gestão Escolar
11. Requisitos de Software Educativo 100-1.1 Conceitos e Características 100-1.2 Componentes 100-1.3 Hardware 100-1.4 Software 100-1.5 People ware
12. E-learning 110-1.1 Conceitos 110-1.2 Classificação 110-1.3 Componentes 110-1.4 Avaliação de E-learning
13. Cooperação e Colaboração 120-1.1 Conceitos 120-1.2 Técnicas 120-1.3 Ferramentas
14. Avaliação de Produto (Ambientes E-learning) 130-1.1 Estudo dos métodos de avaliação para software educacional (Ambientes E-learning) 130-1.2 Análise dos critérios de avaliação pedagógica 130-1.3 Análise dos critérios de avaliação tecnológica 130-1.4 Seleção de ferramentas para avaliação do produto 130-1.5 Escolha de uma ferramenta 130-1.6 Aplicação da avaliação com o uso da Ferramenta 130-1.7 Geração do relatório de resultados da avaliação do produto e-learning
15. Produção de artigo técnico-científico 140-1.1 Escolha de um tema de pesquisa na área de educação a distância 140-1.2 Levantamento da fundamentação teórica do tema 140-1.3 Levantamento do estado da arte do tema 140-1.4 Análise e discussão dos trabalhos relacionados 140-1.5 Geração do resultados e conclusões
16. Pesquisa sobre Ambientes E-learning 150-1.1 Definição da Referência Técnica do Software Educacional 150-1.2 Descrição geral do produto 150-1.3 Metodologia pedagógica 150-1.4 Funcionalidade do sistema 150-1.5 Mídias utilizadas 150-1.6 Ferramentas utilizadas no desenvolvimento do produto 150-1.7 Tutorias de demonstração do produto para utilização 150-1.8 Verificação da alocação do produto 150-1.9 Custo do produto 150-1.10 Conclusão
17. Fórum de Discussão 160-01 - Apresentação do tema 160-02 - Apontar bibliografia sobre o tema 160-03 - Discussão do tema 160-04 - Avaliação do tema
18. Seminário equipe 1 170-01 PEA a Distância 170-01 Evolução Histórica
19. Seminário equipe 2 180-01 - PEA - Foco na criança (pedagogia) 180-02 - PEA - Foco no adulto (andragogia) 180-03 - o Comparativo entre o modelo pedagógico e o modelo andragógico
20. Seminário equipe 3 190-01 - Fundamentos da EAD

Plano de ensino

190-02 - Vantagens 190-03 - Desvantagens
21. Seminário equipe 4 200-01 - E-learning: Contexto geral 200-02 - Conceitos (as visões dos autores)
22. Seminário equipe 5 210-01 - Ambientes E-learning: Conceitos 210-02 - Mídias utilizadas X complexidade 210-03 - Tipos de comunicação
23. Seminário equipe 6 220-01 - Ambientes E-learning (AE) 220-02 - Componentes 220-02 - Modalidades 220-03 - Classificação 220-04 - Vantagens e Desvantagens 220-05 - Exemplos de AEs
24. Seminário equipe 7 230-01 - Avaliação de Ambientes E-learning (AE) 230-01 - Conceitos de avaliação de Software 230-01 - Tipos de avaliação para AE
25. Seminário equipe 8 240-01 - Modelos de Aprendizagem
26. Seminário equipe 9 250-01 - Avaliação Pedagógica de E-learning 250-01 - MAEP (Silva, 2002): Categoria Ensino-Aprendizagem
27. Seminário equipe 10 260-01 - MAEP (Silva, 2002): 260-02 - Categoria Dispositivo de Formação 260-03 - Controle e Gestão de Processo 260-04 - Validade Político-Pedagógico
28. Seminário equipe 11 270-01 - Avaliação Técnica de E-learning 270-01 - Modelo de Avaliação de Qualidade de Software 270-01 - Visão Geral do Modelo Conceitual MA-AE 270-01 - Apresentação do SIA-AE 270-01 - Adaptação do SIA-AE ao modelo de Domínio a ao modelo do Usuário
29. Apresentação do Ambientes E-learning (pesquisa) 280-1.1 Apresentação Técnica do Software Educacional (overview) 280-1.2 Discussão sobre os recursos do produto 280-1.3 Conclusão

Metodologia

1. Aulas expositivas, aulas dialogadas leitura e discussões sobre textos científicos, atividades em laboratório, seminários, trabalhos e exercícios individuais e em grupo. Desenvolvimento de exercícios e atividades por meio do Ambiente de Ensino-Aprendizagem Adaptativo na Web - AdaptWeb® e demais ferramentas de apoio. Até 20% da carga horária poderá ser desenvolvida à distância.

Sistema de avaliação

1. Exercícios teóricos e práticos e Participação efetiva nas discussões em sala de aula - individual ou em grupo, correspondendo a 10% da média da disciplina; Apresentação de seminários - individual ou em grupo, correspondendo a 20% da média da disciplina; Prova teórica (conceitual) - individual, correspondendo a 30% da média da disciplina; Apresentação do projeto final (Software Educacional) e/ou Produção de artigo técnico-científico - individual, correspondendo a 40% da média da disciplina.
--

Bibliografia básica

1. BEHAR, Patrícia A. Modelos Pedagógicos em educação a distância. São Paulo: art Med, 2009. LITTO, Fredric M; FORMIGA, Marcos. Educação a distância: o estado da arte. 2. ed. São Paulo: Pearson/Prentice Hall, 2012. 443 p. ISBN 9788576058830(broch.). MÁTAR, João. Tutoria e interação em educação a distância. São Paulo: Cengage Learning, 2012. 207 p. (Educação e tecnologia.). ISBN 9788522111824.

Bibliografia complementar

1. BELLONI, M. L. Educação a distância. Campinas: Autores Associados, 1999.

Plano de ensino

CAMPOS, G.H.B. Metodologia para avaliação da qualidade de software educacional. Diretrizes para desenvolvedores e usuários. Tese de D.Se. COPPE/Sistemas-UFRJ, Rio de Janeiro, 1994.

FILATRO, Andrea. Design instrucional contextualizado: educação e tecnologia . 3. ed. São Paulo: Ed. SENAC, 2010. 215 p. ISBN 9788573599329(broch.).

KEMCZINSKI, A. Ensino de Graduação pela Internet: Um Modelo de Ensino-Aprendizagem Semipresencial. Universidade Federal de Santa Catarina. Programa de Pós-Graduação em Engenharia de Produção - PPGE/UFSC, Florianópolis, 2000. (Dissertação de mestrado).

KEMCZINSKI, A. Método de avaliação para ambientes e-learning. Universidade Federal de Santa Catarina. Programa de Pós-Graduação em Engenharia de Produção - PPGE/UFSC, Florianópolis, 2005. (Tese de doutorado).

MOORE, Michael G; KEARSLEY, Greg. Educação à distância: uma visão integrada. São Paulo: Thomson Learning, 2007. 398 p. ISBN 9788522105762 (broch.).

PIMENTEL, Mariano; FUKS, Hugo; SOCIEDADE BRASILEIRA DE COMPUTAÇÃO. Sistemas colaborativos. Rio de Janeiro: Campus, 2012. 375 p. (Série Campus). ISBN 9788535246698 (broch.).

SILVA, Andreza Regina Lopes da; SPANHOL, Fernando José. Design instrucional e construção do conhecimento na EaD. Jundiaí: Paco Editorial, 2014. 126 p. ISBN 9788581484334.

BOGO, L. H. Criação de Comunidades Virtuais a partir de Agentes Inteligentes: uma aplicação em e-learning. Universidade Federal de Santa Catarina. Programa de Pós-Graduação em Engenharia de Produção. Florianópolis, 2003. (Dissertação de mestrado).

SILVA, C. R. de O, . Maep: um método ergopedagógico interativo de avaliação para produtos educacionais informatizados. 224 f. Universidade Federal de Santa Catarina. Programa de Pós-graduação em Engenharia de Produção. Florianópolis, 2002. (Tese de doutorado)

SILVA, R. W. A. Educação a distância em ambientes de aprendizagem matemática auxiliada pela realidade virtual. 124 f. Universidade Federal de Santa Catarina. Programa de Pós-graduação em Engenharia de Produção. Florianópolis. 2001. Disponível em: [defesa/pdf/7882.pdf](#)>.

Acesso: set. 2004. (Dissertação de mestrado).

Sociedade da Informação Brasil (Org.) Livro Verde - Sociedade da Informação no Brasil, Ministério da Ciência e Tecnologia, setembro de 2000.

Plano de ensino

Curso: CCI-BAC - Bacharelado em Ciência da Computação

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

Disciplina: OIRC001 - INTERCONEXÃO DE REDES DE COMPUTADORES

Período letivo: 2018/2

Carga horária: 72

Professor: 2933900 - JANINE KNISS

Ementa

1. Estudo de serviços e protocolos necessários a implementação da infra-estrutura de redes de comunicação de dados. Apresentação de protocolos de roteamento, técnicas de switching e redes sem fio.

Objetivo geral

1. Desenvolver no aluno a capacidade de projetar e implementar uma infra-estrutura de redes de comunicação de dados.

Objetivo específico

1. Discutir as principais características dos protocolos de roteamento;
 - Identificar e implementar/simular diferentes protocolos da camada de enlace de dados;
 - Conceituar a problemática do escalonamento de endereços;
 - Desenvolver projeto de redes de computadores utilizando sumarização de rotas;
 - Apresentar as principais redes sem fios.

Conteúdo programático

1. 1. Introdução: Revisão dos conceitos básicos
 - 1.1. Modelo OSI;
 - 1.2. Modelo TCP/IP.
2. 2. Introdução: Conceitos básicos
 - 2.1. Comunicação fim-a-fim;
 - 2.2. Funcionamento de um roteador;
 - 2.3. Internetworking - roteamento e protocolos de roteamento;
 - 2.4. Internetworking - conectividade e roteamento estático.
3. 3. Protocolos de roteamento dinâmico
 - 3.1. Conceito de vetor distância;
 - 3.2. Conceito de estado do link.
4. 4. Links WAN e switching
 - 4.1. Conexão ponto-a-ponto;
 - 4.2. Protocolos de encapsulamento;
 - 4.3. Switches: conceitos básicos e configurações;
 - 4.4. Redes locais virtuais (VLANs);
 - 4.5. Projeto de redes.
5. 5. Tópicos avançados
 - 5.1. Tradução de endereços de rede;
 - 5.2. Controle de acesso (camada 3);
 - 5.3. VLSM;
 - 5.4. Sumarização de rotas.
6. 6. Redes sem fio.

Metodologia

1. Aulas Expositivo-Dialogadas; Uso de Laboratório; Desenvolvimento de atividades individuais e em grupo. Desenvolvimento de exercícios e atividades extras.
 - Será utilizado o sistema Moodle onde serão disponibilizados arquivos utilizados na disciplina, bem como, avisos e dúvidas quanto aos assuntos da disciplina.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. 2 Provas teóricas: Prova 1 (30%) e Prova 2 (30%);
Exercícios (15% da nota final) e Trabalhos práticos (25% da nota final)

Plano de ensino

Participação efetiva nas aulas.

Bibliografia básica

1. KUROSE, J. F.; ROSS, K. W. Computer Networking - A top-down approach featuring the Internet. 3th ed. Addison-Wesley Co., 2003.
TANENBAUM, Andrew S. Redes de Computadores. Rio de Janeiro. Editora Campus, 4ª. Edição, 2003.
SOARES, L. F. G. Redes de computadores: das LANS, MANS E WANS as redes ATM. Rio de Janeiro, Editora Campus, 2000.

Bibliografia complementar

1. STALLINGS, William WIRELESS COMMUNICATIONS & NETWORKS. Prentice Hall, 2004.
Cisco Systems Inc. Fundamentals Of Wireless LANS. Cisco Press, 2003.

Plano de ensino

Curso: CCI-BAC - Bacharelado em Ciência da Computação

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

Disciplina: OMIC001 - MICROPROCESSADORES

Período letivo: 2018/2

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 Estrutura do Microprocessador
2. 2. Microcontroladores
2.1 Microprocessador e Microcontrolador
2.2 Microcontroladores e Periféricos
2.3 Arquiteturas dos microcontroladores PIC (Microchip) e AVR (Atmel)
3. 3. Programação Assembly
3.1 Conjunto de Instruções
3.2 O Montador Assembly e o Formato Intel HEX
3.3 Gravando o Programa no Microcontrolador
3.4 Acessando Periféricos em Assembly
4. 4. Programação em Linguagem C
4.1 Diferenças entre Compiladores
4.2 Passos da Compilação
4.3 Acessando Periféricos em Linguagem C
5. 5. 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:

Plano de ensino

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

Bibliografia básica

1. TANENBAUM, Andrew S. Organização estruturada de computadores. 5. ed. São Paulo: Pearson, c2007. 449 p. ISBN 8521611706 (broch.).

STALLINGS, William. Arquitetura e organização de computadores. 8. ed. São Paulo: Pearson Education do Brasil, 2010. 624 p. ISBN 9788576055648 (broch.).

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

Bibliografia complementar

1. TAUB, Herbert; BARBOSA, Fernando Fontes. Circuitos digitais e microprocessadores

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

MCROBERTS, Michael. Arduino básico. São Paulo: Novatec, 2011. 453 p. ISBN 9788575222744 (broch.).

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

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

Plano de ensino

Curso: CCI-BAC - Bacharelado em Ciência da Computação

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

Disciplina: OPRP001 - PROGRAMAÇÃO PARALELA

Período letivo: 2018/2

Carga horária: 72

Professor: 6675298 - GUILHERME PIEGAS KOSLOVSKI

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. Introdução à programação paralela; Motivação, modelos, definições e conceitos básicos da área, computadores paralelos e elementos para análise de desempenho do paralelismo.
2. Aplicações práticas; Desafios de programação paralela; Tendências; Análise de problemas solucionáveis;
3. Algoritmos e estratégias de particionamento e divisão para conquistar aplicados a problemas clássicos.
4. Introdução a bibliotecas de desenvolvimento de software paralelos; Alguns estudos de caso: MPI, OpenMP e CUDA.
5. Projeto final da disciplina; Elaboração e desenvolvimento;

Metodologia

1. Aulas Expositivo-Dialogadas; Uso de Laboratório; Desenvolvimento de atividades individuais e em grupo. Ocasionalmente ocorrerá o desenvolvimento de atividades através de um ambiente de auxílio para aprendizado a distância.

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.

$$\text{Nota Final} = A1 * 0,30 + A2 * 0,35 + A3 * 0,35$$

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.

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

Plano de ensino

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.
10. Artigos da área.

Plano de ensino

Curso: CCI-BAC - Bacharelado em Ciência da Computação

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

Disciplina: OSRC001 - SEGURANÇA EM REDES DE COMPUTADORES

Período letivo: 2018/2

Carga horária: 72

Professor: 3877850 - RAFAEL RODRIGUES OBELHEIRO

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. Capacitar o aluno a compreender os princípios de segurança de sistemas e de segurança de redes de computadores.

Objetivo específico

1. - Conceituar segurança computacional e suas propriedades fundamentais;
- Compreender os princípios de autenticação, controle de acesso e políticas de segurança;
- Compreender os mecanismos básicos de criptografia e seus usos;
- Compreender os princípios de segurança de software;
- Compreender os aspectos de segurança de redes TCP/IP e em geral;
- Compreender os mecanismos básicos de ataques e controle de acesso em redes;
- Compreender os princípios de detecção de intrusões e tratamento de incidentes de segurança.

Conteúdo programático

1. Introdução
Apresentação da disciplina
Plano de ensino
Método de avaliação
2. Conceitos básicos de segurança
Atributos de segurança: confidencialidade, integridade, disponibilidade
Estratégias de segurança: prevenção, detecção, recuperação
Ameaças, vulnerabilidades, ataques
Princípios de análise de riscos
3. Ameaças e princípios de segurança
Ameaças de segurança: definição e principais tipos
Perfil dos atacantes
Princípios de segurança de Saltzer & Schroeder
4. Autenticação
Senhas
Outras credenciais de conhecimento
Credenciais de posse
Credenciais biométricas
Autenticação multifatores
5. Políticas de segurança
Políticas formais de segurança
Políticas informais de segurança
Estudo de caso: política de segurança da UDESC
6. Controle de acesso
Princípios de controle de acesso
Modelo matriz de acesso e controle de acesso discricionário
Controle de acesso obrigatório
7. Controle de acesso discricionário
Modelo matriz de acesso
Estudo de caso: controle de acesso discricionário no Unix
8. Controle de acesso obrigatório
Rótulos de acesso
Modelo Bell-LaPadula
Modelo Biba
9. Segurança de comunicações

Plano de ensino

Criptografia de chave simétrica Criptografia de chave pública Mecanismos criptográficos de confidencialidade e integridade
10. Infraestrutura criptográfica Protocolos de estabelecimento de chaves Kerberos Infraestruturas de chaves públicas
11. Laboratório de criptografia Atividades práticas usando ferramentas criptográficas
12. Malware Histórico de malware Tipos de malware Técnicas de detecção de malware
13. Segurança de software Segurança no desenvolvimento de software Vulnerabilidades comuns e suas defesas
14. Princípios de funcionamento e de segurança do TCP/IP Estratégias de ataque a redes Firewalls e proxies
15. Princípios de TCP/IP Princípios de funcionamento do TCP/IP Princípios de segurança do TCP/IP
16. Estratégias de ataque a redes Varreduras de redes
17. Laboratório Wireshark/Nmap Atividades práticas usando ferramentas de segurança (Wireshark/Nmap)
18. Laboratório ARP spoofing
19. Firewalls e proxies Firewalls Filtros de pacotes Proxies de circuito Gateways de aplicação
20. Laboratório de firewall Atividades práticas usando ferramenta de configuração de firewall (fwbuilder)
21. Detecção de intrusões Sistemas de detecção de intrusões IDSs baseados em rede e IDSs baseados em host IDSs baseados em assinaturas e IDSs baseados em anomalias Arquitetura de IDSs Limitações de IDSs
22. Laboratório de IDS Atividades práticas usando ferramentas de IDS (Snort)

Metodologia

1. Aula expositiva dialogada. Desenvolvimento de atividades individuais e em grupo. Atividades práticas em laboratório. Até 16 horas-aula (menos que 20% da carga horária da disciplina) poderão ser desenvolvidas na modalidade à distância.

Sistema de avaliação

1. - Duas provas escritas (P1, P2);
- Trabalho em grupo (T);
- Exercícios e atividades de laboratório (EL).

Média final = $0,3 \cdot P1 + 0,3 \cdot P2 + 0,3 \cdot T + 0,1 \cdot EL$

Bibliografia básica

1. ANDERSON, Ross J. Security Engineering: a Guide to Building Dependable Distributed Systems, 2nd Ed. Indianapolis, IN: John Wiley & Sons, 2008.

BISHOP, Matt. Computer Security: Art and Science. Boston, MA: Addison-Wesley, 2002.

STALLINGS, William; BROWN, Lawrie. Segurança de Computadores: Princípios e Práticas, 2ª Ed. Rio de Janeiro: Elsevier, 2014.

Plano de ensino

Bibliografia complementar

1. KUROSE, James F; ROSS, Keith W. Redes de Computadores e a Internet: Uma Abordagem Top-down, 5ª Ed. São Paulo: Pearson, 2011.
SCHNEIER, Bruce. Segurança.com. Rio de Janeiro: Campus, 2001.
TANENBAUM, Andrew S. Sistemas Operacionais Modernos, 3ª Ed. São Paulo: Pearson, 2010.

Plano de ensino

Curso: CCI-BAC - Bacharelado em Ciência da Computação

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

Disciplina: TCC2003 - TRABALHO DE CONCLUSÃO DE CURSO II

Período letivo: 2018/2

Carga horária: 36

Professor: 3639428 - EVERLIN FIGHERA COSTA MARQUES

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

Objetivo específico

Conteúdo programático

Metodologia

1. Desenvolvimento de trabalho com orientador. Entrega de monografia. Defesa da monografia para banca.

Sistema de avaliação

1. Defesa de monografia com avaliação de banca.

Bibliografia básica

Bibliografia complementar