

Caracterização da disciplina

Código da disciplina:	BCM0505-15	Nome da disciplina:	Processamento da Informação						
Créditos (T-P-I):	(3-2-5)	Carga horária:	60 horas	Aula prática:	N	Câmpus:	SBC		
Código da turma:	DA1BCM0505-15SB	Turma:	A1	Turno:	Matutino	Quadrimestre:	1	Ano:	2019
Docente(s) responsável(is):		ALEXANDRE DONIZETI ALVES							

Alocação da turma

	Segunda	Terça	Quarta	Quinta	Sexta	Sábado
8:00 - 9:00				X		
9:00 - 10:00				X		
10:00 - 11:00		X (sem. I)				
11:00 - 12:00		X (sem. I)				
12:00 - 13:00						
13:00 - 14:00						
14:00 - 15:00						
15:00 - 16:00						
16:00 - 17:00						
17:00 - 18:00						
18:00 - 19:00						
19:00 - 20:00						
20:00 - 21:00						
21:00 - 22:00						
22:00 - 23:00						

Planejamento da disciplina
Objetivos gerais

Apresentar os fundamentos sobre manipulação e tratamento da Informação, principalmente por meio da explicação e experimentação dos conceitos e do uso prático da lógica de programação.

Objetivos específicos

Que o aluno seja capaz de compreender os conceitos fundamentais a respeito da manipulação e tratamento da Informação. Que o aluno entenda a lógica de programação de computadores e adquira a habilidade prática de desenvolver algoritmos básicos para modelar e solucionar problemas de natureza técnico-científica, independentemente de uma linguagem ou de um paradigma de programação específicos.

Ementa

Introdução a algoritmos. Variáveis e tipos de dados. Operadores aritméticos, lógicos e precedência. Métodos/Funções e parâmetros. Estruturas de seleção. Estruturas de repetição. Vetores. Matrizes. Entrada e saída de dados. Depuração. Melhores práticas de programação.

Conteúdo programático

Aula	Conteúdo	Estratégias didáticas	Avaliação
1	Apresentação; Alg. Sequencial	Aula expositiva e Exercícios	Por meio de exercícios
2	Algoritmos Sequenciais	Aula expositiva e Exercícios	Por meio de exercícios
3	Estrutura de Seleção	Aula expositiva e Exercícios	Por meio de exercícios
4	Estr. de Repetição (enquanto)	Aula expositiva e Exercícios	Por meio de exercícios
5	Estr. de Repetição (para)	Aula expositiva e Exercícios	Por meio de exercícios
6	Estr. de Repetição (validação)	Aula expositiva e Exercícios	Por meio de exercícios
7	Teste de Mesa e Revisão	Aula expositiva e Exercícios	Por meio de exercícios
8	Primeira Prova	Prova individual	Prova escrita
9	Modularização (função)	Aula expositiva e Exercícios	Por meio de exercícios
10	Vetores (com módulos)	Aula expositiva e Exercícios	Por meio de exercícios
11	Vetores (com módulos)	Aula expositiva e Exercícios	Por meio de exercícios
12	Matrizes (com módulos)	Aula expositiva e Exercícios	Por meio de exercícios
13	Matrizes (com módulos)	Aula expositiva e Exercícios	Por meio de exercícios
14	Revisão	Aula expositiva e Exercícios	Por meio de exercícios
15	Segunda Prova	Prova individual	Prova escrita
16	Prova Substitutiva	Prova individual	Prova escrita
17	Mecanismo de Recuperação	Prova individual	Prova escrita

Descrição dos instrumentos e critérios de avaliação qualitativa

Ferramentas: linguagem de programação Java

Crítérios de Avaliação: Os alunos são avaliados semanalmente através de exercícios práticos realizados em laboratório. Isso permite identificar as dificuldades enfrentadas pelos alunos desde o início do quadrimestre. Os conceitos de teoria são avaliados principalmente em duas avaliações escritas onde os alunos devem demonstrar a capacidade de interpretar código e também de elaborar soluções de problemas e traduzir a solução em um algoritmo formal. Além disso, os alunos são avaliados em duas provas práticas, onde devem mostrar a capacidade de implementar soluções usando uma linguagem de programação. As atividades semanais realizadas na disciplina valem 10% de bônus na Primeira Prova e na Segunda Prova. Plágios serão punidos com o conceito F na prova. A primeira prova vale 25%, a segunda prova vale 35% e as avaliações em laboratório valem 40% do conceito final da disciplina. A prova de recuperação será realizada junto com a prova de recuperação da parte prática (Prova escrita). A média final da disciplina (teoria e prática) será calculada considerando 50% do conceito final (antes do mecanismo de recuperação) e 50% do conceito obtido no mecanismo de recuperação. Os pesos não determinísticos nos cálculos dos conceitos.

Horário de atendimento: quinta-feira, das 12:00 às 13:00 horas, na sala A2-S101-SB, SBC.

Referências bibliográficas básicas

1. FORBELLONE, André Luiz Villar; EBERSPACHER, Henri Frederico. Lógica de programação: a construção de algoritmos e estruturas de dados. 3 ed. São Paulo: Prentice Hall, 2005. 218 p.
2. SEBESTA, Robert W. Conceitos de linguagens de programação. 5 ed. Porto Alegre: Bookman, 2003. 638 p.
3. Ascensio,A.F.;Campos,E.A.,FundamentosdaProgramaçãodeComputadores, Pearson, 3a edição, 2012.

Referências bibliográficas complementares

1. BOENTE, Alfredo. Aprendendo a programar em Pascal: técnicas de programação. 2003. Rio de Janeiro: Braport, 2003. 266 p.
2. Deitel P.; Deitel, H. "Java - Como Programar" - 8a Ed. São Paulo: Prentice Hall Brasil 2010, I.S.B.N.: 9788576055631 pp 1152.
3. Flanagan, D. "Java, o guia essencial" 5a ed. (série O'Reilly) Bookman Cia Ed 2006 ISBN 8560031073, 1099 pp.
4. SEDGEWICK, Robert; WAYNE, Kevin Daniel. Introduction to programming in Java: an interdisciplinary approach. Boston: Pearson Addison-Wesley, 2007. 723 p
5. Puga, S., Lógica de programação e estruturas de dados com aplicações em Java, Pearson Prentice Hall, 2a edição, 2009