

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:	DB3BCM0505-15SB	Turma:	B3	Turno:	Matutino	Quadrimestre:	1	Ano:	2019
Docente(s) responsável(is):		MARCIO K. OIKAWA (T) / ANA C. Q. SIMÕES (P)							

Alocação da turma						
	Segunda	Terça	Quarta	Quinta	Sexta	Sábado
8:00 - 9:00	Prática	Teoria (sem. I)				
9:00 - 10:00	Prática	Teoria (sem. I)				
10:00 - 11:00				Teoria		
11:00 - 12:00				Teoria		
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
11/2 (P)	Apresentação; Apresentação do ambiente de programação.	Aula expositiva e exercícios	Exercícios
12/2 (T)	Algoritmos sequenciais, Teste de mesa	Aula expositiva e exercícios	Exercícios
21/2 (T)	Algoritmos sequenciais com módulo	Aula expositiva e exercícios	Exercícios
18/2 (P)	Algoritmos sequenciais com módulo	Exercícios em laboratório	Exercícios
21/2 (T)	Estruturas de seleção simples e composta com módulo	Aula expositiva e exercícios	Exercícios
25/2 (P)	Estruturas de seleção com módulo	Exercícios em laboratório	Exercícios
26/2 (T)	Estruturas de repetição ENQUANTO/FAÇA-ENQUANTO	Aula expositiva e exercícios	Exercícios
28/2 (T)	Estruturas de repetição PARA	Aula expositiva e exercícios	Exercícios
04/3 (P)	Recesso (carnaval)	Recesso	Recesso
07/3 (T)	Estruturas de repetição - validação de dados	Aula expositiva e exercícios	Exercícios
11/3 (P)	Repetição com módulo, validação de dados	Exercícios em laboratório	Exercícios
12/3 (T)	Teste de mesa, revisão	Aula expositiva e exercícios	Exercícios
14/3 (T)	Prova 1 - Teoria	Avaliação individual	Avaliação escrita
18/3 (P)	Prova 1 - Prática	Avaliação individual	Avaliação no laboratório
21/3 (T)	Introdução a vetores	Aula expositiva e exercícios	Exercícios
25/3 (P)	Vetor com módulo	Exercícios em laboratório	Exercícios
26/3 (T)	Vetores com modularização	Aula expositiva e exercícios	Exercícios
28/3 (T)	Vetores com modularização	Aula expositiva e exercícios	Exercícios
01/4 (P)	Vetor com módulo	Exercícios em laboratório	Exercícios
04/4 (T)	Introdução a matrizes	Aula expositiva e exercícios	Exercícios
08/4 (P)	Recesso	Recesso	Recesso
09/4 (T)	Matrizes com modularização	Aula expositiva e exercícios	Exercícios
11/4 (T)	Matrizes com modularização	Aula expositiva e exercícios	Exercícios
15/4 (P)	Matrizes com módulo	Exercícios em laboratório	Exercícios
18/4 (T)	Revisão	Aula expositiva e exercícios	Exercícios
22/4 (P)	Matrizes com módulo	Exercícios em laboratório	Exercícios
23/4 (T)	Prova 2 - Teoria	Avaliação individual	Avaliação escrita
25/4 (T)	Prova substitutiva (Teoria)	Avaliação individual	Avaliação escrita
29/4 (P)	Prova 2 - Prática	Avaliação individual	Avaliação no laboratório
02/5 (T)	Vista de provas	Avaliação individual	Avaliação no laboratório
Até 06/5	Prova substitutiva (Prática) - extra classe		
10/5	Avaliação de recuperação (REC)	Avaliação individual	Avaliação escrita

Descrição dos instrumentos e critérios de avaliação qualitativa

Ferramentas: linguagem de programação Java; Ambientes de programação (software livre); Computadores.

Crítérios de Avaliação: Os alunos são avaliados por meio de três instrumentos principais: listas de exercícios práticos semanais, avaliações escritas e avaliações usando o computador em laboratório. O cálculo do conceito final será realizado considerando os seguintes critérios:

CÁLCULO DO CONCEITO DE TEORIA (T):

O conceito de teoria é calculado por meio de avaliações escritas, onde os alunos devem demonstrar a capacidade de interpretar problemas, ler códigos e elaborar soluções do forma algorítmica.

A composição do conceito para atividades teóricas obedecerá à seguinte distribuição de pesos:

Prova 1 - 40%

Prova 2 - 60%

CÁLCULO DO CONCEITO DE PRÁTICA (P):

O conceito da Prática considera avaliações onde o aluno deve mostrar capacidade de programar soluções usando uma linguagem de programação.

A composição do conceito para atividades práticas obedecerá à seguinte distribuição de pesos e atividades:

Prova 1 - 30%

Prova 2 - 50%

Exercícios semanais - 20%

CÁLCULO DO CONCEITO FINAL:

Uma vez calculados os conceitos de atividades de Teoria e Prática, o conceito final será calculado a partir da tabela abaixo:

Conceito (T)	Conceito (P)	Conceit o Final
A	A	A
	B	A
	C	B
	D	B
B	A	B
	B	B
	C	B
	D	C
C	A	B
	B	C
	C	C
	D	C
D	A	C
	B	C
	C	D
	D	D
F	Qualquer conceito	F
Qualquer conceito	F	F

AValiação Substitutiva (SUB):

Em cumprimento à Resolução ConsEPE no. 227/2018, os alunos que não puderem comparecer a alguma das avaliações presenciais nos casos previstos na resolução citada terão direito a uma avaliação substitutiva. Para tal, está previsto no cronograma uma data específica, no final do quadrimestre a atendimento a todos os alunos atendidos pela resolução. O conteúdo da avaliação substitutiva é o conteúdo

integral do quadrimestre e o conceito (ou nota) obtido nessa avaliação substituirá o conceito "F" atribuído à atividade na qual o aluno se ausentou.

AValiação DE RECUPERAÇÃO (REC):

Em cumprimento à Resolução ConsEPE no. 182/2014, todos os alunos que obtiverem conceito final igual a "D" ou "F" terão direito à realização de avaliação de recuperação, que seguirá os seguintes critérios:

- A composição do conceito final após a recuperação será formado segundo a tabela abaixo:

Conceito final antes da REC	REC	Conceito final do quadrimestre
D	A	C
	B	C
	C	D
	D	D
	F	D*
F	A	C
	B	D
	C	D
	D	F
	F	F

* Para fins de cálculo do conceito final do quadrimestre, garante-se ao aluno o maior conceito entre o obtido antes e após a realização da REC.

PLÁGIOS:

Por considerar o uso frequente de atividades em computador e internet, a avaliação dos exercícios envolvendo codificação de algoritmos está sujeita a plágios durante o seu desenvolvimento. A fim de preservar o compromisso da universidade com o caráter pedagógico das atividades e o compromisso ético com a propriedade e integridade intelectual, casos suspeitos de plágio serão **severamente** punidos com a **anulação integral de todas as atividades** envolvidas no caso.

REPROVAÇÃO POR AUSÊNCIAS:

Serão considerados reprovados por ausência os alunos que apresentarem número de faltas superior a 25% do número de aulas, somadas a Teoria e a Prática. Para esses casos, será atribuído conceito final "O".

ATIVIDADES DE APOIO (HORÁRIO DE ATENDIMENTO):

Em cumprimento à Resolução CONSUNI no. 183/2017, esta turma prevê os seguintes horários de atendimento extraclasse, para atividades de apoio aos alunos:

- Segunda-feira, das 12:00h às 13:00h, na sala A2-L001-SB - profa. Ana C. Q. Simões
- Quinta-feira, das 12:00h às 13:00h, na sala 273, Bloco Delta, SBC - prof. Marcio K. Oikawa

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