

<b>Caracterização da disciplina</b>										
Código da disciplina:	<b>BCM0505-15</b>		Nome da disciplina:		<b>Processamento da Informação</b>					
Créditos (T-P-I):	<b>(3-2-5)</b>		Carga horária:	<b>60 horas</b>	Aula prática:	<b>N</b>	Câmpus:	<b>SBC</b>		
Código da turma:	<b>DA1BCM0505-15SB</b>		Turma:	<b>A1</b>	Turno:	<b>Matutino</b>	Quadrimestre:	<b>3</b>	Ano:	<b>2022</b>
Docente(s) responsável(is):			<b>MARCIO K. OIKAWA (T) / MARCIO K. OIKAWA (P)</b>							

<b>Alocação da turma</b>						
	Segunda	Terça	Quarta	Quinta	Sexta	Sábado
8:00 - 9:00		Teoria (sem. I)	Teoria			
9:00 - 10:00		Teoria (sem. I)	Teoria			
10:00 - 11:00	Prática					
11:00 - 12:00	Prática					
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						

<b>Planejamento da disciplina</b>			
<b>Objetivos gerais</b>			
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.			
<b>Objetivos específicos</b>			
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.			
<b>Ementa</b>			
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.			
<b>Conteúdo programático</b>			
<b>Aula</b>	<b>Conteúdo</b>	<b>Estratégias didáticas</b>	<b>Avaliação</b>
19/09 (P)	Exercícios de ambientação com programação. Introdução. Algoritmos sequenciais.	Resolução de problemas. Exemplos.	Exercícios práticos
20/09 (T)	Apresentação da disciplina. Noções básicas de programação de computadores. Noções de algoritmos. Variáveis. Operadores matemáticos. Operadores de atribuição. Precedência. Programação de operações em fluxo sequencial.	Aula expositiva em vídeo. Resolução de problemas. Exemplos.	Não há
21/09 (T)	Estruturas de controle de fluxo. Estrutura condicional. Operadores lógicos e relacionais. Introdução a modularização.	Aula expositiva dialogada. Resolução de problemas. Exemplos.	Não há
26/09 (P)	Exercícios com algoritmos sequenciais e condicionais.	Resolução de problemas. Exemplos.	Exercícios práticos
28/09 (T)	Modularização	Aula expositiva dialogada. Resolução de problemas. Exemplos	Não há
03/10 (P)	Exercícios com modularização	Resolução de problemas. Exemplos.	Exercícios práticos
04/10 (T)	Estruturas de controle de fluxo. Estruturas de repetição.	Aula expositiva dialogada. Resolução de problemas. Exemplos	Não há
05/10 (T)	Repetição baseada em condições lógicas (while). Repetição baseada em contagens (for)	Aula expositiva dialogada. Resolução de problemas. Exemplos	Não há
10/10 (P)	Exercícios com repetição.	Resolução de problemas. Exemplos.	Exercícios práticos
12/10 (T)	Recesso	Recesso	Recesso
17/10 (P)	Exercícios com repetição.	Resolução de problemas. Exemplos.	Exercícios práticos
18/10 (T)	Prova 1 (teoria)	Avaliação presencial	Avaliação presencial
19/10 (T)	Vetores e strings	Aula expositiva dialogada. Resolução de problemas. Exemplos	Não há
24/10 (P)	Exercícios com strings.	Resolução de problemas. Exemplos.	Exercícios práticos
26/10 (T)	Operações com vetores. Busca. Rastreamento. Ordenação.	Aula expositiva dialogada. Resolução de problemas.	Não há

31/10 (P)	Exercícios com vetores	Exemplos Resolução de problemas. Exemplos.	Exercícios práticos
01/11 (T)	Algoritmos de ordenação de vetores.	Aula expositiva dialogada. Resolução de problemas. Exemplos	Não há
02/11 (T)	Recesso	Recesso	Recesso
07/11 (P)	Exercícios com ordenação de vetores.	Resolução de problemas. Exemplos.	Exercícios práticos
09/11 (T)	Matrizes. Conceito. Operadores básicos. Operações.	Aula expositiva dialogada. Resolução de problemas. Exemplos	Não há
14/11 (P)	Recesso	Recesso	Recesso
15/11 (T)	Recesso	Recesso	Recesso
16/11 (T)	Matrizes. Manipulação. Operações matemáticas.	Aula expositiva dialogada. Resolução de problemas. Exemplos	Não há
21/11 (P)	Exercícios com matrizes.	Resolução de problemas. Exemplos.	Exercícios práticos
23/11 (T)	Matrizes. Operações matemáticas.	Aula expositiva dialogada. Resolução de problemas. Exemplos	Não há
28/11 (P)	Exercícios com matrizes.	Resolução de problemas. Exemplos.	Exercícios práticos
29/11 (T)	Revisão	Aula expositiva dialogada. Resolução de problemas. Exemplos	Não há
30/11 (T)	Prova 2	Avaliação	Avaliação
05/12 (P)	Resolução de exercícios.	Resolução de problemas. Exemplos.	Não há
07/12 (T)	Prova sub (teoria)	Avaliação	Avaliação
12/12 (T)	Correção de provas e composição de conceitos.	Não há.	Não há.
15/12 (T)	Correção de provas e composição de conceitos.	Não há.	Não há.
16/12 (P)	Prova REC	Avaliação	Avaliação

**Descrição dos instrumentos e critérios de avaliação qualitativa**

**Ferramentas:** Linguagem de programação Python; Ambientes de desenvolvimento (software livre); Computadores; Navegadores web; Acesso à internet.

**Critérios de Avaliação:**

O cálculo do conceito final (CF) utilizará dois conceitos parciais: conceito do conteúdo teórico (T) e conceito de exercícios práticos (E):

CALCULO DO CONTEÚDO TEÓRICO (T):

A avaliação será realizada por meio de três provas, sendo que cada uma possui os seguintes pesos:

- Prova 1 (P1): peso 1
- Prova 2 (P2): peso 2

Cada prova (P1, P2 e P3) possuirá uma pontuação individual no intervalo de 0-100. A média ponderada (M) das três

provas será calculado seguindo a fórmula:

$$M = (P1 + P2 \times 2)/3$$

O conceito final T será obtido a partir do mapeamento da tabela a seguir:

Conceito (T/E)	Médias(M/ME)
<b>A</b>	90 – 100
<b>B</b>	75 – 89
<b>C</b>	60 – 74
<b>D</b>	50 – 59
<b>F</b>	0 – 49

CÁLCULO DAS ATIVIDADES PRÁTICAS (E):

O conceito de atividades práticas (E) será obtido pela média aritmética ponderada dos exercícios desenvolvidos durante o quadrimestre regular. Cada exercício terá uma nota (N) e um peso (W). O cálculo da média de exercícios (ME) será dado pela expressão:

$$ME = (N_1 \cdot W_1 + N_2 \cdot W_2 + N_3 \cdot W_3 + \dots + N_k \cdot W_k) / (W_1 + W_2 + W_3 + \dots + W_k)$$

onde k é o número de exercícios práticos aplicados no quadrimestre.

O conceito E é o mapeamento de ME para o conceito correspondente usando a mesma tabela acima.

CONCEITO FINAL (CF):

O conceito final da disciplina utilizará a seguinte tabela para sua composição:

Conceito Teoria (T)	Conceito Prática (E)	Conceito 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, definimos os critérios para avaliação substitutiva.

A prova substitutiva é destinada àqueles que perderam uma das provas teóricas do curso, independentemente do motivo.

*Obs:* Quem perdeu todas as provas só poderá substituir uma única nota com a prova sub. A composição da segunda nota deverá ser tratada com o professor para realização de nova avaliação.

Para a composição das atividades práticas, os alunos que não puderam entregar seus exercícios por motivos de força maior previstos na Resolução ConsEPE no. 227/2018, poderão requerer novo prazo para entrega dos mesmos ou, se for o caso, nova bateria de exercícios com pontuação equivalente ao período requerido.

**AVALIAÇÃO DE RECUPERAÇÃO (REC):**

Em cumprimento à Resolução ConsEPE no. 182/2014, todos os alunos que obtiverem conceito final (CF) 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 formada 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:**

Como a composição da nota final e, conseqüentemente, do conceito final será baseada no desenvolvimento de exercícios práticos de programação, esta disciplina será rigorosa com relação a utilização de códigos de programação plagiados. 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 ausências (conceito “O”) os estudantes que acumularem mais de 25% de faltas sobre o total de aulas do quadrimestre.

**ATIVIDADES DE APOIO (HORÁRIO DE ATENDIMENTO):**

Em cumprimento à Resolução CONSUNI no. 183/2017, os horários de atendimento serão presenciais, indicados a seguir:

- Marcio Katsumi Oikawa: quartas-feiras, 11h-13h.
- Marcio Katsumi Oikawa: segundas-feiras, 12h-13h.

**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