

Plano de Ensino

1. IDENTIFICAÇÃO

Curso: Ciência da Computação

Componente curricular: Algoritmo e Programação - Turma B

Fase: Primeiro

Ano/semestre: 2015/1

Número de créditos: 4

Carga horária – Hora aula: 72

Carga horária – Hora relógio: 60

Professor: Dr. Rafael Torchelsen

Atendimento ao Aluno: quinta-feira das 10:00 às 12:00

2. OBJETIVO GERAL DO CURSO

O curso tem por objetivo a formação integral de novos cientistas e profissionais da computação, os quais deverão possuir conhecimentos técnicos e científicos e serem capazes de aplicar estes conhecimentos, de forma inovadora e transformadora, nas diferentes áreas de conhecimento da Computação. Adicionalmente, os egressos do curso deverão ser capazes de adaptar-se às constantes mudanças tecnológicas e sociais, e ter uma formação ao mesmo tempo cidadã, interdisciplinar e profissional

3. EMENTA

Conceito e construção de algoritmos. Tipos básicos de dados. Comandos de atribuição, condicionais e de repetição. Registros. Vetores e Matrizes. Modularização.

4. OBJETIVOS

4.1 GERAL

Desenvolver o raciocínio lógico para o desenvolvimento de algoritmos.

4.2 ESPECÍFICOS

Compreender como o computador executa programas. Desenvolver algoritmos simples utilizando os conceitos aprendidos.

5. CRONOGRAMA E CONTEÚDOS PROGRAMÁTICOS

ENCONTRO	CONTEÚDO
1	Introdução à disciplina; apresentação do plano de ensino; estrutura lógica dos computadores
2	Algoritmos, fluxograma, pseudo-código, linguagens de programação, compiladores
3	Algoritmos com entrada, saída e processamento
4	Introdução à linguagem de programação C (código-fonte, compilador, arquivo .c, printf(), IDE)
5	Introdução a variáveis (memória, sintaxe, tipos, atribuição); básico sobre printf()
6	Operações aritméticas com variáveis; básico sobre scanf()
7	Condições simples e operadores de comparação (==, !=, >=, <=, etc)
8	Explicação sobre if, if..else, if..else if, else
9	Condições compostas e tabela verdade; Condições compostas com parênteses; Procedência de operadores;

10	Estruturas de repetição (while e do...while) e iterações
11	Estruturas de repetição (for)
12	Exercícios
13	Exercícios
14	Primeira prova (P1), conteúdo: Tudo visto até o dia da prova
15	Modularização e funções (protótipo, parâmetros, retorno, escopo)
16	Recuperação NP1, conteúdo: mesmo de P1
17	Bibliotecas de funções
18	Introdução a vetores
19	Aprofundamento sobre vetores
20	Vetores bidimensionais (matrizes)
21	Aprofundamento sobre matrizes
22	Introdução a structs
23	Aprofundamento sobre structs
24	Ponteiros
25	Ponteiros
26	Segunda prova (P2), conteúdo: Tudo visto até o dia da prova
27	Apresentação do trabalho final
28	Recuperação NP2, conteúdo: mesmo de P2
29	Apresentação do trabalho final

6. PROCEDIMENTOS METODOLÓGICOS

Conduzir a disciplina com aulas expositivas/dialogadas enquanto discutidos os itens de cunho teórico, evoluindo em tópicos específicos para exercícios práticos, demonstrações, contextualização baseada em publicações atualizadas. Uso de atividades em laboratórios com o objetivo de apresentar/exercitar os conceitos estudados.

7. AVALIAÇÃO DO PROCESSO ENSINO-APRENDIZAGEM

Uso de abordagens tais como: provas teóricas e práticas, avaliação escrita em aula, exercícios extraclasse, trabalhos de implementação, entre outros.

As avaliações serão agrupadas em dois momentos Notas Parciais 1 e 2 (NP1 e NP2, respectivamente). A NP1 será composta por uma avaliação escrita e prática (P1), trabalhos extraclasse (G1) realizados até a data da prova, com o seguinte cálculo:

$$NP1 = P1 + G1$$

$$\text{onde } 0 \leq P1 \leq 7 \text{ e } 0 \leq G1 \leq 3$$

sendo G1 calculado da seguinte forma:

$$G1 = T1 + T2 + \dots Tn$$

onde T_i representa a nota de um trabalho, onde o peso e a quantidade de trabalhos são decididos em aula, antes de sua aplicação.

A NP2 será composta por uma avaliação escrita e prática (P2), trabalhos extraclasse (G2)

realizados até a data da prova, com o seguinte cálculo:

$$NP2 = P2 + G2$$

$$\text{onde } 0 \leq P2 \leq 7 \text{ e } 0 \leq G2 \leq 3$$

sendo G2 calculado da seguinte forma:

$$G2 = T1 + T2 + \dots Tn$$

onde T_i representa a nota de um trabalho, onde o peso e a quantidade de trabalhos são decididos em aula.

A média final (MF) será calculada como $MF = (NP1 + NP2) / 2$.

Em caso de plágio as seguintes regras serão aplicadas.

Prova:

- O aluno recebe nota zero na prova onde o plágio foi detectado, além disso, por demonstrar prática não aceitável o caso será levado ao conhecimento do colegiado;

Trabalhos:

- É permitido usar conteúdo da internet, livros, colegas, etc., contanto que uma citação seja feita. A nota do trabalho será proporcional ao conteúdo original;
- Caso seja detectado plágio o aluno recebe zero no trabalho em questão, além disso, por demonstrar prática não aceitável o caso será levado ao conhecimento do colegiado;

7.1 RECUPERAÇÃO: NOVAS OPORTUNIDADES DE APRENDIZAGEM E AVALIAÇÃO

As avaliações e trabalhos serão discutidos e corrigidos em sala de aula. Esta discussão tem como objetivo oferecer uma nova oportunidade de aprendizagem do conteúdo avaliado. Todos os estudantes participarão da discussão.

Se algum estudante obter notas parciais (NP) abaixo de 6,0, será oferecida uma oportunidade, através de uma prova PR com todo o conteúdo, que será utilizada para calcular a nova média como:

$$NP' = NP * 0,5 + PR * 0,5$$

8. REFERÊNCIAS

8.1 BÁSICA

1. LOPES, A., GARCIA, G. Introdução à Programação. Rio de Janeiro: Campus, 2002.
2. VILARIM, G. Algoritmos: Programação para Iniciantes. Rio de Janeiro: Ciência Moderna, 2004.
3. FORBELLONE, A. Lógica de Programação: A Construção de Algoritmos e Estruturas de Dados. São Paulo: Makron Books, 2005.
4. MANZANO, José Augusto N. G.; OLIVEIRA, Jayr Figueiredo. Algoritmos: Lógica para Desenvolvimento de Programação de Computadores. 19 ed.: Érica, 2001.

8.2 COMPLEMENTAR

1. LOPES, A.; GARCIA, G. Introdução à Programação: 500 algoritmos resolvidos. Campus, 2002.
2. ARAÚJO, E. C. Algoritmos: Fundamentos e Prática. Florianópolis: Visual Books, 2ed., 2005.
3. BORATTI, I. C. Introdução à Programação: Algoritmos. Florianópolis: Visual Books, 2ed., 2004

Professor

Coordenador do curso