



SERVIÇO PÚBLICO FEDERAL  
MINISTÉRIO DA EDUCAÇÃO  
UNIVERSIDADE FEDERAL DE UBERLÂNDIA



ANEXO DA RESOLUÇÃO Nº 30/2011, DO CONSELHO DE GRADUAÇÃO

Instituto de Física

COLEGIADO DO CURSO DE FÍSICA LICENCIATURA

PLANO DE ENSINO

1. IDENTIFICAÇÃO

<b>COMPONENTE CURRICULAR:</b> Algoritmos e Programação de Computadores				
<b>UNIDADE OFERTANTE:</b> Faculdade de Computação				
<b>CÓDIGO:</b> FACOM31302		<b>PERÍODO:</b> 3º		<b>TURMA:</b> F
<b>CARGA HORÁRIA</b>			<b>NATUREZA</b>	
<b>TEÓRICA:</b> 30	<b>PRÁTICA:</b> 30	<b>TOTAL:</b> 60	<b>OBRIGATÓRIA:</b> ( X )	<b>OPTATIVA:</b> ( )
<b>PROFESSOR:</b> Anilton Joaquim da Silva				<b>ANO/SEMESTRE:</b> 2020/AARE/ETAPA 2
<b>OBSERVAÇÕES:</b>				

2. EMENTA

Noções sobre os componentes de computadores. Uso de softwares (sistemas operacionais, internet e outros). Noções de lógica. Introdução a algoritmos. Resolução de problemas utilizando algoritmos e raciocínio lógico. Tipos de dados. Variáveis e constantes. Expressões e operadores. Estruturas de controle: Estruturas básicas, Estruturas condicionais e estruturas de repetição. Estruturas básicas de dados: vetores, matrizes e registros. Apresentação das funções, construção de módulos e escrita de arquivos.

3. JUSTIFICATIVA

O aprendizado de uma linguagem de programação estimula a criatividade, a autonomia, o desenvolvimento do raciocínio lógico e aumenta a capacidade de resolução de problemas. Através deste conhecimento os alunos podem desenvolver sua criatividade e raciocínio para encontrar soluções para problemas de diversas áreas. Quando se aprende a programar, de maneira paralela, se aprende a lidar com os desafios cognitivos que exigem principalmente a concentração e o pensamento sequencial e lógico.

A expectativa é de que no final do curso, o estudante tenha adquirido os conceitos que fundamentam a disciplina e as formas de aplicá-los na resolução de problemas reais no decorrer de sua vida acadêmica e profissional.

4. OBJETIVO

**Objetivo Geral:**

- Conceituar os principais termos da informática;
- Desenvolver habilidades para a utilização do computador como ferramenta de trabalho em sua atividade profissional;



### Objetivos Específicos:

- Utilizar o computador como ferramenta na execução de tarefas enquanto estudante e para solução de problemas em sua atividade profissional;
- Desenvolver programas em uma linguagem procedimental empregando adequadamente os recursos oferecidos por esta linguagem.

### 5. PROGRAMA

1. Noções sobre os componentes de computadores
  - 1.1. Placa mãe, dispositivos de armazenamentos, dispositivos de entrada e dispositivos de saída
  - 1.2. Terminologia (hardware, software, programa, bit, byte, códigos: Binário e ASCII)
2. Sistemas operacionais, aplicativos, rede de computadores e Internet
  - 2.1. Sistemas operacionais: *windows, linux, unix*;
  - 2.2. Comunicação em redes de computadores.
3. Introdução a Algoritmos
  - 3.1. Conceito de algoritmos e programas;
  - 3.2. Linguagens de programação;
  - 3.3. Tipos primitivos de dados e variáveis ;
  - 3.4. Expressões: aritméticas, relacionais e lógicas ;
  - 3.5. Comandos de: atribuição, entrada e saída de dados ;
  - 3.6. Estruturas de controle de fluxo: seleção e repetição.
4. Introdução à Linguagem de Programação Python
  - 4.1. A estrutura de programas em Python
  - 4.2. Definição de identificadores
  - 4.3. Tipos de dados primitivos
  - 4.4. Declaração e inicialização de variáveis
  - 4.5. Conceitos sobre entrada e saída .
5. Estrutura Sequencial – Atribuição
  - 5.1. Expressões aritméticas;
  - 5.2. Operadores e construções especiais no Python.
6. Estruturas de Seleção
  - 6.1. Dados e operadores lógicos;
  - 6.2. Avaliação de expressões lógica;
  - 6.3. Comando de seleção *if ... elif ... eles*.
7. Estruturas de Repetição
  - 7.1. Conceitos sobre laços de repetição;
  - 7.2. Laços *while*;
  - 7.3. Laços *for*;
  - 7.4. Ramificação e decisões: execução condicional, condições e laços aninhados.
8. Vetores e Matrizes
  - 8.1. Conceitos sobre arranjos;
  - 8.2. O uso de vetores e matrizes em Python.
9. Funções
  - 9.1. Projeto de programa estruturado;
  - 9.2. Funções em Python ;
  - 9.3. Estratégias para comunicação inter-funções (passagem de parâmetros e *return*) ;
  - 9.4. Funções padrão.



10. Módulos e pacotes em Python
  - 10.1. Carregando pacotes ;
  - 10.2. Usando e construindo módulos.
11. Arquivos
  - 11.1. Noções sobre arquivos;
  - 11.2. Funções para abrir e fechar arquivos ;
  - 11.3. Funções de entrada/saída para arquivos texto.

## 6. METODOLOGIA

- O conteúdo será apresentado na forma expositiva utilizando a plataforma de apoio remoto MS Teams para as atividades síncronas.
- O curso será composto por 9 horas aulas semanais, sendo 4 aulas teóricas síncronas e 5 aulas assíncronas.
- As aulas assíncronas, práticas de exercícios, serão ministradas em módulos, laboratórios, onde o aluno terá contato direto com o microcomputador e com os aplicativos utilizados no curso. Estas aulas serão ministradas através video-aulas e de listas de exercícios.
- Serão disponibilizados monitores para auxiliar o professor no esclarecimento de dúvidas e correção dos exercícios.
- O conhecimento será verificado e explorado nas aulas assíncronas.
- O aprendizado será avaliado através de provas.

O programa da disciplina seguirá o seguinte cronograma:

### α) Atividades síncronas: 30 horas

- Horários das atividades síncronas:  
segunda-feira - 19:00 - 20:40  
terça-feira - 19:00 - 20:40
- Plataforma de T.I./softwares que serão utilizados:  
**plataforma MS Teams - equipe APC Física, VisualG, e IDLE Python.**

### β) Agenda das aulas síncronas:

26/10/2020- T – Placa mãe, dispositivos de: armazenamento, entrada, e saída; Software, Redes, Internet;  
27/10/2020- T - Algoritmos: linguagem, tipos de dados, expressões, comandos de entrada e saída  
03/11/2020- T - Algoritmos: comandos de seleção e repetição;  
09/11/2020- T - Introdução à Linguagem de Programação Python;  
10/11/2020- T - Estrutura Sequencial – Atribuição;  
16/11/2020- T - Estruturas de Seleção;  
17/11/2020- T - Estruturas de Seleção;  
23/11/2020- T -Estruturas de Repetição;;  
24/11/2020- T - Estruturas de Repetição;  
30/11/2020- T - Estruturas de Repetição;  
01/12/2020- T - **Prova 1** - Programação Python;  
07/12/2020- T – Vetores e Matrizes;  
08/12/2020- T - Vetores e Matrizes;



14/12/2020- T - Funções;

15/12/2020- T - Módulos e pacotes em Python; Arquivos;

21/12/2020- T - **Prova 2** – programar em Python.

22/12/2020- T – Arquivos .

χ) **Atividades assíncronas: 42 horas**

plataforma MS Teams - equipe: APC Física - canal Geral

## 7. AVALIAÇÃO

A avaliação do aluno será feita por meio dos laboratórios individuais, das aulas assíncronas, duas provas prática, individual, a saber:

- laboratórios, individuais:
  - Algoritmos - 3 laboratórios - 15 pontos;
  - programação em Python - 5 laboratórios - 35 pontos;
- Prova 1 individual, prática - Python - 25 pontos;
- Prova 2 individual, prática - Python - 25 pontos;

### Datas, horários e Local da prova:

- 01/12/2020 - Prova1 – 19:00 até as 20:40 – Equipe APC Física, no MS Teams
- 21/12/2020 - Prova2 – 19:00 até as 20:40 – Equipe APC Física, no MS Teams

## 8. BIBLIOGRAFIA

### Básica

LUTZ, M.; ASCHER, D. **Aprendendo Python**. Porto Alegre: Bookman, 2007. 566 p.

MENEZES, N.N.C. **Introdução à Programação com Python**: algoritmos e lógica de programação para iniciantes. São Paulo: Novatec, 2010. 222 p.

ASCENCIO, Ana Fernanda Gomes; e CAMPOS, Edilene A.V. de. **Fundamentos da Programação de Computadores**. Editora Pearson Prentice Hall, 2007, 434p.

### Complementar

CAPRON, H.L; JOHNSON, J.A. **Introdução à informática**. Editora Pearson Prentice Hall, 2004, 350p.

FORBELLONE, A. L. V.; EBERSPÄCHER, H. F. **Lógica de programação**: a construção de algoritmos estruturas de dados. São Paulo: Pearson Prentice Hall, 2000. 196p.

GUIMARÃES, A. M.; LAGES, N. A. C. **Introdução à ciência da computação**. Rio de Janeiro: LTC, 1984. 165p.

## 9. APROVAÇÃO

Aprovado em reunião do Colegiado realizada em: \_\_\_\_/\_\_\_\_/\_\_\_\_

Coordenação do Curso de Graduação em: \_\_\_\_\_