

Identificação

- **Curso:** Fundamentos de Programação
- **Professor:** Albert E. F. Muritiba
- **Período:** 2024.1
- **Notas de Aula:** [slides](#)

Cronograma do Curso de Fundamentos de Programação

Semana 1: Primeiros Passos com Python

- Introdução ao ambiente de programação Python (instalação e uso do IDE).
- Primeiros códigos em Python: "Hello, World!" e operações simples.
- Variáveis e tipos de dados básicos.
- Introdução ao conceito de algoritmo de forma prática.

Semana 2: Operações Aritméticas e Entrada/Saída de Dados

- Operações aritméticas e lógicas básicas.
- Entrada (`input()`) e saída de dados (`print()`).
- Exercícios práticos simples.

Semana 3: Condicionais em Python

- Estruturas de controle: `if`, `elif`, `else`.
- Operadores relacionais e lógicos.
- Exemplos práticos de tomada de decisão.
- Introdução ao conceito de fluxo de controle de algoritmos.

Semana 4: Laços de Repetição: `while`

- Estruturas de repetição: introdução ao laço `while`.
- Controle de laços com `break` e `continue`.
- Exercícios de repetição com problemas práticos.

Semana 5: Laços de Repetição: `for`

- Introdução ao laço `for`.
- Iteração em listas e strings com `for`.
- Aplicação prática: cálculo de somatórios e manipulação de listas.

Avaliação 1: Prova 1

Semana 6: Listas em Python

- Estruturas de dados homogêneas: introdução a listas.
- Operações com listas: inserção, remoção e modificação.
- Métodos úteis de listas (`append()`, `remove()`, `sort()`, etc.).

Semana 7: Estruturas de Dados Avançadas: Tuplas e Conjuntos

- Diferença entre listas, tuplas e conjuntos.
- Aplicações práticas: onde usar cada tipo de estrutura.
- Operações com tuplas e conjuntos.

Semana 8: Matrizes e Manipulação de Dados em Python

- Implementação de matrizes usando listas aninhadas.
- Manipulação de matrizes: leitura e escrita.
- Introdução à biblioteca `numpy` (opcional, como extensão).

Semana 9: Funções em Python (Parte 1)

- Introdução ao conceito de função.
- Definição de funções simples e retorno de valores.
- Passagem de parâmetros.

Semana 10: Funções em Python (Parte 2)

- Escopo de variáveis (local e global).
- Funções com múltiplos parâmetros e valores padrão.
- Funções anônimas (`lambda`) e suas aplicações.

Semana 11: Recursividade em Python

- Conceito de recursividade e sua aplicação.
- Exemplos clássicos: fatorial, Fibonacci.
- Comparação entre recursão e iteração.

Semana 12: Dicionários em Python

- Estruturas de dados heterogêneas: introdução aos dicionários.
- Operações com dicionários: inserção, remoção e iteração.
- Aplicações práticas com dados heterogêneos.

Semana 13: Manipulação de Arquivos em Python

- Leitura e escrita de arquivos.
- Manipulação de arquivos CSV.
- Context manager (`with`) para controle de arquivos.

Avaliação 2: Prova 2

Semana 14: Revisão e Projeto Intermediário

- Revisão dos conceitos até aqui.
- Projeto prático envolvendo estruturas de controle, funções e manipulação de dados.

Semana 15: Introdução à Linguagem C (Sintaxe Básica)

- Comparação entre Python e C: principais diferenças.
- Estrutura básica de um programa em C.
- Declaração de variáveis e tipos de dados em C.

Semana 16: Ponteiros e Alocação Dinâmica em C

- Introdução aos ponteiros em C.
- Alocação dinâmica de memória.
- Diferenças de gerenciamento de memória entre Python e C.

Semana 17: Estruturas de Controle e Funções em C

- Condicionais e laços em C.
- Definição de funções e passagem de parâmetros em C.
- Comparação de controle de fluxo entre Python e C.

Semana 18: Projeto Final

- Projeto integrador que envolva Python e C.
- Manipulação de arquivos, uso de ponteiros, funções e estruturas de controle.

Avaliação

- **Provas:** 40%
- **Trabalhos Práticos:** 30%
- **Projeto Final:** 30%