



INSTITUTO FEDERAL DE EDUCAÇÃO, CIÊNCIA E TECNOLOGIA DO PIAUÍ  
CAMPUS TERESINA-CENTRAL  
DIRETORIA DE ENSINO  
Dep. de Informação, Ambiente, Saúde e Produção Alimentícia – DIASPA  
Coordenação do Curso Técnico em Desenvolvimento de Sistemas

## PLANO DE DISCIPLINA

### IDENTIFICAÇÃO

<b>CURSO</b>	Técnico em Desenvolvimento de Sistemas						
<b>Disciplina</b>	Programação Estruturada de Computadores					<b>CÓDIGO:</b>	TDS18-15
<b>MÓDULO:</b>	I	<b>SEMESTRE:</b>	1º	<b>ANO:</b>	2019	<b>CARGA HORÁRIA:</b>	102 horas
<b>DOCENTE:</b>	JOSÉ RITOMAR CARNEIRO TORQUATO						
<b>COORDENADOR:</b>	JOSÉ RITOMAR CARNEIRO TORQUATO						

### EMENTA:

Introdução à Programação de Computadores; Estruturas de Sequência, Condicionais, Repetições, Modularização. Tipos Estruturados.

### OBJETIVO GERAL DA DISCIPLINA:

Desenvolver programas utilizando uma linguagem de Programação Estruturada.

### OBJETIVOS ESPECÍFICOS DA DISCIPLINA:

- Compreender a programação estruturada de computadores utilizando a linguagem Python;
- Conhecer os comandos básicos da linguagem;
- Conhecer operadores, tipos de dados e expressões;
- Criar programas estruturados de forma modularizada;
- Aplicar estruturas condicionais e de repetição na criação de programas estruturados;
- Conhecer os principais tipos de dados estruturados: listas, tuplas e dicionários;
- Manipulação de arquivos com Python.

### CONTEÚDO PROGRAMÁTICO:

<b>01. INTRODUÇÃO À PROGRAMAÇÃO</b> 1.1. Linguagens existentes e seus níveis 1.2. O que é linguagem de máquina; 1.3. Compilador, interpretador e máquina virtual 1.4. Características da linguagem Python 1.5. O interpretador de comandos Python; 1.6. O ambiente de desenvolvimento - IDLE; 1.7. Ajuda Python; 1.8. Projeto: Arte em ASCII	<b>02. COMANDOS BÁSICOS EM PYTHON</b> 2.1. Comandos de entrada e saída de dados 2.2. Variáveis 2.3. Tipos de variáveis 2.4. Tratamento de erros 2.5. Projeto: O ano 2025
--	---

**CONTEÚDO PROGRAMÁTICO (continuação):****03. TIPOS DE DADOS E OPERADORES**

- 3.1. Tipos de dados primitivos
- 3.2. Formatação e operações com strings
- 3.2. Conversão de tipos
- 3.3. Operadores de atribuição
- 3.4. Operadores aritméticos
- 3.5. Operadores lógicos
- 3.6. Prioridade de operadores

**04. MODULARIZAÇÃO I**

- 4.1. Módulos e sub-rotinas
- 4.2. Passagem de parâmetros
- 4.3. Retorno de dados
- 4.4. Escopo de variáveis
- 4.5. Doc strings

**05. CONDICIONAIS**

- 5.1. Condicionais simples
- 5.2. Condicionais compostas
- 5.3. Condicionais com múltiplas escolhas
- 5.4. Estruturas Aninhadas
- 5.5. Projeto: Quis

**06. REPETIÇÕES**

- 6.1. Condição de parada
- 6.2. Repetição infinita: erro
- 6.3. Repetição com final determinado
- 6.4. Projeto: O poder da Tartaruga
- 6.5. Repetição com final indeterminado
- 6.6. Projeto: Porta da Fortuna

**07. LISTAS E TUPLAS**

- 7.1. Criação de listas
- 7.2. Elementos em uma lista
- 7.3. Trabalhando com índices
- 7.4. Cópia e fatiamento de lista
- 7.5. Tamanho de uma lista
- 7.6. Adição e remoção de elementos
- 7.7. Listas imutáveis: Tuplas
- 7.8. Projeto: Gerador de Cumprimentos

**08. DICIONÁRIOS**

- 8.1. Índices x chaves
- 8.2. Chaves e valores
- 8.3. Criação de dicionários
- 8.4. Adição e remoção de elementos
- 8.5. Busca de chaves

**09. MODULARIZAÇÃO II**

- 9.1. Funções recursivas
- 9.2. Parâmetros opcionais
- 9.3. Parâmetros nomeados
- 9.4. Empacotamento e desempacotamento
- 9.5. Módulos

**10. ARQUIVOS**

- 10.1. Criação de arquivos
- 10.2. Leitura e escrita em arquivos
- 10.3. Processamento de um arquivo
- 10.4. Caminhos em disco
- 10.5. Pastas ou Diretórios

**METODOLOGIA:**

- Aulas expositivas e dialogadas;
- Aulas práticas em laboratório;
- Exercícios;
- Grupos de discussão;
- Desenvolvimento de trabalhos individuais e coletivos de pesquisa;
- Exibição e discussão de filme/vídeos sobre temas referentes à disciplina.
- Desenvolvimento de projetos individuais e/ou coletivos.

**RECURSOS DIDÁTICOS:**

- Projetor multimídia;
- Quadro, pincel apagador;
- Laboratório de informática;
- Vídeo-aulas;
- Ferramentas da Internet para gameificação e outras para o ensino-aprendizagem;

#### SISTEMÁTICA DE AVALIAÇÃO:

- Qualitativa e quantitativa no decorrer do processo de aprendizagem, através da observação;
- Criação de portfólio do aluno com conteúdos relacionados à disciplina;
- Verificação de aprendizagem individual com a aplicação de provas escritas e/ou práticas.

#### REFERÊNCIAS (BÁSICA E COMPLEMENTAR):

##### BÁSICA:

DOWNEY, Allen. **Pense em Python**; São Paulo: Novatec, 2016.

MENEZES, Nilo Ney Coutinho. **Introdução a programação com Python**. São Paulo: Editora Novatec, 2010.

ASCHER, Mark Lutz David. **Aprendendo Python**; Porto Alegre: Bookman, 2007.

BORGES, Luiz Eduardo. **Python para Desenvolvedores: Aborda Python 3.3**. São Paulo: Novatec Editora, 2014.

##### COMPLEMENTAR:

WAZLAWICK, Raul Sidnei. **Introdução a Algoritmos e Programação com Python: uma abordagem dirigida por testes**. Rio de Janeiro: Elsevier, 2018.

ALVES, Fábio Junior. **Introdução à Linguagem de Programação Python**. Rio de Janeiro: Editora Ciência Moderna, 2013.

Luiz Eduardo Borges. **Python para Desenvolvedores**. São Paulo: Novatec, 2014.

Al Sweigart. **Automatize tarefas maçantes com Python**. São Paulo: Novatec, 2015.

#### Assinaturas:

---

José Ritomar Carneiro Torquato  
Professor

---

Departamento Pedagógico  
Carimbo e Assinatura

---

José Ritomar Carneiro Torquato  
Coordenador