

PLANO DE ENSINO

Disciplina: Algoritmos e Programação – Linguagens e Métodos Formais

Professor Responsável: Raul Sidnei Wazlawick

Créditos: (02 CRÉDITOS 30HS)

Semestre: 2019-2

Sala : H211 (PPGSC1)

1. Ementa Geral

Introdução aos algoritmos; Tipos básicos, variáveis e constantes; Operadores aritméticos, lógicos e relacionais; Comandos de atribuição, entrada e saída de dados; Estruturas de controle: sequencial, condicional e de repetição; Variáveis compostas; Modularização de algoritmos; Algoritmos de busca e ordenação.

2. Objetivos

Apresentar ao aluno as noções mais básicas de programação de computadores através do ensino de algoritmos com o uso de uma linguagem e ambiente de programação. O Aluno deverá desenvolver as habilidades de programação com estruturas de controle e modularização a partir do uso de uma linguagem de programação moderna.

3. Conteúdo

- 1 Introdução
 - 1.1 Porque Aprender a Programar?
 - 1.2 O que Significa "Programar" um Computador
 - 1.3 Linguagens de Programação
 - 1.4 História da Linguagem Python
 - 1.5 Como Obter um Ambiente de Programação Python
- 2 Alô Mundo
 - 2.1 Escrevendo seu Primeiro Programa
 - 2.2 Explicando a Função "*print*"
 - 2.3 Expressões com Números Inteiros
 - 2.4 Expressões com Números de Ponto Flutuante
- 3 Variáveis
 - 3.1 Significado de uma Variável
 - 3.2 Comando de Atribuição
 - 3.3 Identificador
 - 3.4 Tipos de Valores Numéricos
 - 3.5 Operações Aritméticas
 - 3.6 Formatação de Saída
 - 3.7 Entrada de Dados pelo Teclado
 - 3.8 Precedência de Operadores
 - 3.9 Constantes
 - 3.10 Arredondamento
 - 3.11 Notação Científica ou Exponencial
- 4 Repetição de Comandos um Número Fixo de Vezes

SERVIÇO PÚBLICO FEDERAL
UNIVERSIDADE FEDERAL DE SANTA CATARINA
CENTRO DE CIÊNCIAS DA SAÚDE
PROGRAMA DE MESTRADO PROFISSIONAL EM INFORMÁTICA EM SAÚDE

- 4.1 Fundamentos: Contagem simples e variável de iteração
- 4.2 Contagem a partir de Valores Quaisquer
- 4.3 Vários Comandos Subordinados a uma Estrutura de Repetição
- 4.4 Contando de n em n
- 4.5 Contando de Trás para Frente
- 4.6 Progressão Não Aritmética
- 4.7 Revisitando Variáveis: Comandos de atualização
- 4.8 Somatório
- 4.9 Fatorial
- 4.10 Aninhamento
- 5 Comandos de Seleção
 - 5.1 Fundamentos: Decisão e condição
 - 5.2 Comparação de Números de Ponto Flutuante
 - 5.3 Comando "*if*" sem "*else*"
 - 5.4 Estruturas de Seleção Aninhadas
 - 5.5 Uso Combinado de Estruturas de Seleção e Repetição
 - 5.6 Revisitando Variáveis: Variável booleana
 - 5.7 Condição Composta
 - 5.8 Seleção Múltipla
- 6 Repetição Condicionada
 - 6.1 Fundamentos
 - 6.2 Simulações com Números
 - 6.3 Repetição Infinita
 - 6.4 Revisitando Operadores Aritméticos: Divisão por zero
 - 6.5 Comparação entre Repetição Fixa e Condicionada
 - 6.6 Repetição com Condição Composta
 - 6.7 Atalhos em Expressões com "*and*" e "*or*"
 - 6.8 Cálculo de Séries
 - 6.9 Séries Alternadas
- 7 Funções
 - 7.1 Fundamentos
 - 7.2 Funções Predefinidas
 - 7.2.1 Função "*divmod*"
 - 7.2.2 Função "*len*"
 - 7.2.3 Função "*help*"
 - 7.3 Bibliotecas de Funções
 - 7.4 Como Definir Novas Funções
 - 7.5 Parâmetros Default
 - 7.6 Funções sem Retorno ou Não Frutíferas
 - 7.7 Escopo de Variáveis
 - 7.8 Alteração de Argumentos
 - 7.9 Recursividade
 - 7.10 Tópico Especial: Otimização de código

**SERVIÇO PÚBLICO FEDERAL
UNIVERSIDADE FEDERAL DE SANTA CATARINA
CENTRO DE CIÊNCIAS DA SAÚDE
PROGRAMA DE MESTRADO PROFISSIONAL EM INFORMÁTICA EM SAÚDE**

- 7.11 Criando sua Própria Biblioteca de Funções
- 8 Desenvolvimento Dirigido por Teste
 - 8.1 Tratamento de Exceções
 - 8.2 Filosofia do Desenvolvimento Dirigido por Teste
 - 8.3 Classes de Equivalência
 - 8.4 Testando Valores Inválidos
 - 8.5 Precondições
 - 8.6 Análise de Valor Limítrofe
- 9 Estruturas de Dados Primitivas em Python
 - 9.1 Listas
 - 9.1.1 Fundamentos: Iteração simples em listas
 - 9.1.2 Iteração Múltipla em Listas
 - 9.1.3 Alteração de Valores em Listas
 - 9.1.4 Mutabilidade em Listas
 - 9.1.5 Trocando Elementos de Posição
 - 9.1.6 Criando uma Lista Ordenada
 - 9.2 Tuplas
 - 9.3 Dicionários
- 10 Estruturas de Dados Derivadas
 - 10.1 Pilhas
 - 10.2 Filas
 - 10.3 Simulação Discreta com Filas
 - 10.4 Matriz
- 11 Arquivos
 - 11.1 Criação de Arquivos
 - 11.2 Escrita em Arquivos
 - 11.3 Leitura em Arquivos
 - 11.4 Operações com *Strings*
 - 11.4.1 Mensagens "*startswith*" e "*endswith*"
 - 11.4.2 Iteração por Caracteres
 - 11.4.3 Pesquisa de *Substrings*
 - 11.4.4 Maiúsculas e Minúsculas
 - 11.4.5 Formatação

4. Metodologia

Os alunos receberão material para estudar previamente. Durante as aulas haverá exposição e discussão sobre os assuntos previstos para o dia bem como a realização de exercícios em sala. A disciplina será realizada com 22 horas de aula presenciais e 8 horas de atividades à distância, compostas pela realização de exercícios de programação.

5. Cronograma

**SERVIÇO PÚBLICO FEDERAL
UNIVERSIDADE FEDERAL DE SANTA CATARINA
CENTRO DE CIÊNCIAS DA SAÚDE
PROGRAMA DE MESTRADO PROFISSIONAL EM INFORMÁTICA EM SAÚDE**

Data/Hora	Atividade/Conteúdo	Método	CH	Leituras Recomendadas
28/6	Introdução	Discussão em aula	3	Cap. 1
5/7	Primeiros programas	Discussão em aula	3	Cap. 2
12/7	Variáveis	Discussão em aula	3	Cap. 3
2/8	Repetição fixa	Discussão em aula	3	Cap. 4
9/8	Seleção	Discussão em aula	3	Cap. 5
16/8	Repetição condicionada	Discussão em aula	3	Cap. 6
23/8	Funções	Discussão em aula	3	Cap. 7
30/8	Desenvolvimento dirigido por teste	Discussão em aula	3	Cap. 8
6/9	Arquivos	Discussão em aula	3	Cap. 11

6. Avaliação:

O aluno será avaliado continuamente pela realização de trabalhos em aula e fora dela. A avaliação final será realizada através de exercícios a serem resolvidos em sala e as atividades de recuperação da mesma forma.

7. Bibliografia (no máximo 30)

Wazlawick, R. S. Introdução a Algoritmos e Programação com Python. Elsevier, 2017. (Livro texto). Disponibilizado no Moodle.

8. Bibliografia Complementar (no máximo 15)

Menezes, N. N. C. (2014). Introdução à Programação com Python, Novatec. (R\$38,33)

Ramalho, L. (2015). Python Fluente. Novatec. (R\$91,12)

Severance, C. R. (2014). Python for Informatics: Exploring Information. Amazon (e-book gratis)