

Introdução a algoritmos e pseudocódigos. Introdução à programação imperativa variáveis, constantes e expressões. Controle de fluxo de execução e repetição.

Estruturas triviais de dados vetores, matrizes e registros. Noções de funções. Comandos de atribuição e declaração de constantes, variáveis e tipos de dados. Expressões. Ponteiros. Instruções condicionais de controle de fluxo. Bibliotecas definidas pelo usuário. Recursividade. Alocação dinâmica de memória.

III- OBJETIVOS (Indicar os objetivos gerais e específicos para o componente curricular)

Objetivos Gerais: Apresentar os conceitos básicos inerentes à programação de computadores utilizando formas de representações de problemas com construções de algoritmos e programas. Objetivos Específicos: Apresentar formas léxicas para a elaboração de um algoritmo/programa, e possibilitar a sua construção utilizando uma linguagem de programação de alto nível; Desenvolver no aluno a habilidade de programação, através da utilização de elementos, variáveis, operadores, expressões, estruturas sequenciais, estruturas de repetição, tipos estruturados de dados, modularização de programas, herança, arquivos de dados, biblioteca padrão; Ao final da disciplina, o estudante deverá ser capaz de desenvolver algoritmos e programas para solucionar problemas reais.

IV- CONTEÚDO PROGRAMÁTICO (Apontar os assuntos a serem abordados no componente curricular)

Introdução e estruturação de algoritmos/programas ? Introdução aos algoritmos; estruturação, linguagem natural, fluxograma, pseudocódigo. Linguagem de programação, principais diferenças, legibilidade do código, programas e scripts Ambiente de desenvolvimento. Noções do modo interativo (IDLE) aplicações básicas, uso como calculadora, recursos básicos, auto-complete. Tipagem dinâmica e forte, constantes, variáveis e expressões. Entrada e saída de dados (modo interativo), documentação do código. Valores numéricos e cadeias de caracteres ? Nomes de variáveis e palavras reservadas, operadores lógico-matemáticos. Noções do ambiente integrado de desenvolvimento (IDE), do teste de mesa (debug) Estruturas básicas de dados ? Criação de listas, tuplas, dicionários e conjuntos Estruturas de Controle ? Comandos de condição, condições alternativas, encadeadas e aninhadas, instrução return e pass. Comandos de repetição, for e while, break e continue Refinamento de algoritmos/programas ? Definição e estruturação de funções. Funções base da linguagem; Definição de classes; Recursividade; Controle de erros; Introspecção; Utilização de módulos; Introdução à orientação a objetos Utilização de bibliotecas da linguagem; Biblioteca padrão da linguagem; Biblioteca matemática voltada ao cálculo numérico; Biblioteca arquivos de entrada e saída; Biblioteca sistema operacional; Biblioteca tempo

V- METODOLOGIA (Descrever a metodologia que será utilizada nas atividades propostas para o ensino remoto (ex: videoaula, fórum, lista de exercícios, estudos dirigidos, elaboração de projetos, produção de artigo científico, entre outros)

Encontros síncronos para explicação da metodologia da disciplina e de como o aluno deve se organizar para os estudos. Vídeo aulas passando o conteúdo da disciplina. Fórum para dirimir dúvidas. Listas de exercícios para fixação do conteúdo. Projeto de disciplina para avaliação de desempenho.

VI- PLATAFORMA (S) ESCOLHIDA (S) PARA O ENSINO REMOTO:
(Escolher uma (ou mais) plataforma (s) de ensino a ser (em) usada (s) como repositório para os conteúdos do componente curricular)

☒ Ambiente Virtual de Aprendizagem (AVA/ Moodle)

☒ Google Meet

☐ Site do docente

☐ Blog do docente

☐ Outro: _____

VII- FORMAS DE AVALIAÇÃO (Detalhar como serão os procedimentos que serão usados para compor a nota)

Listas de exercícios e Projeto de disciplina

VIII- CRONOGRAMA DO COMPONENTE CURRICULAR

SEMANA	DESCRIÇÃO DAS ATIVIDADES PLANEJADAS (destacar quando se tratar de atividade síncrona)
1	CONTEÚDOS ABORDADOS: Introdução à disciplina , arquitetura de computadores. METODOLOGIA: Encontro síncrono, vídeo aula e apostila PRÁTICAS AVALIATIVAS: lista de exercícios.
2	CONTEÚDOS ABORDADOS: Linguagem de máquina (assembly) com o simulador online METODOLOGIA: Encontro síncrono, vídeo aula PRÁTICAS AVALIATIVAS: lista de exercícios.
3	CONTEÚDOS ABORDADOS: Linguagem de programação C , conceitos básicos, declaração de variáveis, estruturas de controle METODOLOGIA: Encontro síncrono, vídeo aula

	PRÁTICAS AVALIATIVAS: lista de exercícios.
4	<p>CONTEÚDOS ABORDADOS: Linguagem de programação C, estrutura de repetição, leitura e escrita de arquivos</p> <p>METODOLOGIA: Encontro síncrono, vídeo aula</p> <p>PRÁTICAS AVALIATIVAS: lista de exercícios.</p>
5	<p>CONTEÚDOS ABORDADOS: Construção do 1º Projeto de disciplina</p> <p>METODOLOGIA: Formalização do projeto</p> <p>PRÁTICAS AVALIATIVAS: Projeto prático (1ª VA).</p>
6	<p>CONTEÚDOS ABORDADOS: Linguagem de Python, conceitos básicos, declaração de variáveis, estruturas de controle</p> <p>METODOLOGIA: Encontro síncrono, vídeo aula</p> <p>PRÁTICAS AVALIATIVAS: lista de exercícios.</p>
7	<p>CONTEÚDOS ABORDADOS: Linguagem de programação Python, conceitos básicos, declaração de variáveis, estruturas de controle</p> <p>METODOLOGIA: Encontro síncrono, vídeo aula</p> <p>PRÁTICAS AVALIATIVAS: lista de exercícios.</p>
8	<p>CONTEÚDOS ABORDADOS: Linguagem de programação Python, estruturas de dados, tratamento de erros e exceções e conceitos complementares</p> <p>METODOLOGIA: Encontro síncrono, vídeo aula</p> <p>PRÁTICAS AVALIATIVAS: lista de exercícios.</p>
9	<p>CONTEÚDOS ABORDADOS: Construção do 2º Projeto de disciplina</p> <p>METODOLOGIA: Formalização do projeto</p> <p>PRÁTICAS AVALIATIVAS: Projeto prático (2ª VA e 3ª VA)</p>
10	<p>CONTEÚDOS ABORDADOS: Construção do 2º Projeto de disciplina</p> <p>METODOLOGIA: Formalização do projeto</p> <p>PRÁTICAS AVALIATIVAS: Final (02/03)</p>

IX- BIBLIOGRAFIA

BÁSICA:

1. John V. Guttag (2013), Introduction to Computation and Programming Using Python, Spring. Cormen, T. H., Leiserson, C.
2. E., Rivest, R. L. Stein, C. (2002), Algoritmos Teoria e Prática, Editora Campus. Forbellone, A. L. V. Eberspacher, H. F. (2005),
3. Lógica de Programação - a Construção de Algoritmos e Estruturas de Dados, Pearson.

COMPLEMENTAR

1. Lopes, A. Garcia, G. (2002), Introdução a Programação, Editora Campus
2. Rossum, G. V. (2004), Tutorial de Python, Disponível gratuitamente em <http://python.org/>
3. Ziviani, N. (2004), Projeto de Algoritmos, Editora Nova Fronteira
4. Dasgupta, S., Papadimitriou, C. H. Vazirani, U. V. (2006), Algoritmos, McGraw Hill
5. Sebesta, R. W. (2001), Conceitos de Linguagens de Programação, Bookman

Garanhuns, 03 de Novembro de 2020



Daliton da Silva
SIAPE. 21150397