

TeSP - Animação e Modelação 3D

Técnico Superior Profissional

Plano: Aviso n.º 12718/2016 - 19/10/2016

Ficha da Unidade Curricular: Linguagens de Programação

ECTS: 6; Horas - Totais: 162.0, Contacto e Tipologia, TP:15.0; PL:45.0;

Ano | Semestre: 1 | S1; Ramo: Tronco comum;

Tipo: Obrigatória; Interação: Presencial; Código: 62004

Área de educação e formação: Ciências informáticas

Docente Responsável

Hélder da Corte Pestana

Professor Adjunto

Docente e horas de contacto

Ricardo Jorge Martins Lucas

Assistente Convidado, TP: 15; PL: 45;

Objetivos de Aprendizagem

Análise, desenho e implementação de algoritmos como ferramenta para a resolução de vários tipos de problemas. Introdução às estruturas de dados. Aprendizagem da linguagem de programação python e sua utilização na implementação dos algoritmos abordados no âmbito do programa. Criação de programas de complexidade pequena e média.

Objetivos de Aprendizagem (detalhado)

Análise, desenho e implementação de algoritmos como ferramenta para a resolução de vários tipos de problemas. Introdução às estruturas de dados. Aprendizagem da linguagem de programação python e sua utilização na implementação dos algoritmos abordados no âmbito do programa. Criação de programas de complexidade pequena e média.

Conteúdos Programáticos

1. Algoritmia;
2. Pseudocódigo;
3. Diagramas de Fluxo;
4. Tipos de dados e Operações Primitivas;
5. Programação estruturada;
6. Constantes, Variáveis e Expressões;
7. Estruturas de Dados básicas;
8. Funções Internas;
9. Funções;

Conteúdos Programáticos (detalhado)

1. Algoritmia;
 - a. Conceito, vantagens e desvantagens;
 - b. Escrita de algoritmos simples;
2. Pseudocódigo;
 - a. Conceito, vantagens e desvantagens;
 - b. Escrita de algoritmos em pseudocódigo simples;

- c. Estruturas de decisão e repetição (If, case, while e for)
3. Diagramas de Fluxo;
 - a. Escrita de algoritmos em pseudocódigo mais complexos;
 - b. Desenho de fluxogramas;
4. Tipos de dados e Operações Primitivas;
5. Programação estruturada;
6. Constantes e Variáveis e Expressões;
7. Estruturas de Dados básicas;
 - a. Listas;
 - b. Dicionários;
 - c. Tuplos;
 - d. Sets;
8. Funções Internas;
 - a. Built-in functions do python;
9. Funções;

Metodologias de avaliação

Aval. Continua:

- Observação em aula - 10%
- Componente de avaliação teórica - 40%, nota mínima de 7 valores.
- Componente de avaliação prática - 50%, nota mínima de 7 valores. (Mini-projeto: 25%; Projeto: 75%)

Aval. Final:

- Componente de avaliação teórica - 40%, nota mínima de 7 valores.
- Componente de avaliação prática - 60%, nota mínima de 7 valores.

Software utilizado em aula

Pycharm (python language/compiler)

Estágio

Não aplicável.

Bibliografia recomendada

Coerência dos conteúdos programáticos com os objetivos

São abordadas inicialmente os conceitos e as ferramentas para tratamento de algoritmos. Utiliza-se de seguida uma linguagem de programação, a linguagem python, para aplicar os conceitos transmitidos.

Metodologias de ensino

Aulas teórico-práticas expositivas onde se descrevem os conceitos fundamentais. Aulas práticas de resolução de casos práticos e aplicação dos conceitos a cenários de utilização real.

Coerência das metodologias de ensino com os objetivos

Os conceitos teóricos são lecionados com recurso a exemplos simples e ilustrativos. Na componente prática são abordados exercícios mais complexos para uma consolidação profunda dos conceitos.

Língua de ensino

Português

Pré requisitos

Não aplicável.

Programas Opcionais recomendados

Não aplicável.

Observações

Docente Responsável



Assinado de forma digital
por Helder Pestana
Dados: 2020.09.07
12:29:39 +01'00'

Diretor de Curso, Comissão de Curso



Assinado de forma digital
por Helder Pestana
Dados: 2020.09.07
12:29:51 +01'00'

Conselho Técnico-Científico

Jorge
Antunes

Digitally signed by Jorge Antunes
DN: cn=Jorge Antunes, ou=IPT,
ou=ESTA,
email=jorge.antunes@ipt.pt, c=PT
Adobe Acrobat Reader version:
2020.012.2014