

TeSP - Automação Industrial

Técnico Superior Profissional

Plano: Aviso n.º 11774/2016 - 27/09/2016

Ficha da Unidade Curricular: Programação I

ECTS: 5; Horas - Totais: 135.0, Contacto e Tipologia, TP:67,5;

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

Tipo: Obrigatória; Interação: Presencial; Código: 62634

Área de educação e formação: Ciências Informáticas

Docente Responsável

Fernando Sérgio Hortas Rodrigues

Docente e horas de contacto

Fernando Sérgio Hortas Rodrigues

Equiparado Assistente 1º Triénio, TP: 67,5;

Objetivos de Aprendizagem

Resolução de problemas usando algoritmos.

Construção de algoritmos usando uma linguagem de programação.

Construção de programas computacionais usando o 'C' como linguagem de programação.

Conteúdos Programáticos

Conceitos básicos acerca de computação e computadores.

Algoritmos e linguagens.

Manipulação de informação.

Estruturas de decisão.

Estruturas de repetição.

Estruturas de dados compostas (arrays; estruturas; strings).

Modularidade (Funções e Procedimentos).

Recursividade.

Apontadores.

Gestão dinâmica de memória.

Conteúdos Programáticos (detalhado)

1. Conceitos básicos acerca de computação e computadores.

1.1. História da computação

1.2. Funcionamento do computador

1.3. Ergonomia

2. Algoritmos e linguagens.

2.1. Noções de lógica

2.2. Desenvolvimentos de algoritmos

2.3. Linguagens de especificação de algoritmos

2.3.1. Fluxogramas

2.4. Linguagens de programação

- 3. Manipulação de informação.**
 - 3.1. Introdução à linguagem C
 - 3.2. Tipos de dados básicos
 - 3.3. Variáveis e constantes
 - 3.4. Leitura de dados
 - 3.5. Escrita de informação
 - 3.6. Operadores
 - 3.6.1. Sobre inteiros
 - 3.6.2. Sobre reais
 - 3.6.3. Sobre caracteres
 - 3.7. Casting

- 4. Estruturas de Decisão.**
 - 4.1. Valores lógicos
 - 4.2. Operadores lógicos e relacionais
 - 4.3. Expressões lógicas
 - 4.4. Decisão condicional
 - 4.4.1. If-else
 - 4.4.2. Ciclos encadeados
 - 4.5. Decisão por escolha
 - 4.5.1. Switch
 - 4.5.2. Break
 - 4.6. Operador de decisão ?
 - 4.7. Blocos de instruções
 - 4.7.1. Indentação

- 5. Estruturas de Repetição**
 - 5.1. Repetição com controlo inicial
 - 5.1.1. While
 - 5.1.2. For
 - 5.2. Repetição com controlo final
 - 5.2.1. Do –while
 - 5.3. Alteração do fluxo de execução do ciclo
 - 5.3.1. Break
 - 5.3.2. Continue
 - 5.4. Ciclos encadeados
 - 5.5. Ciclos infinitos
 - 5.6. Operadores de incremento e decrementos
 - 5.7. Atribuição composta

- 6. Estruturas de dados compostas (arrays; strings).**
 - 6.1. Arrays
 - 6.1.1. Noção de Array
 - 6.1.2. Declaração e inicialização de arrays
 - 6.1.3. Leitura escrita e processamento de arrays
 - 6.1.4. Arrays multidimensionais
 - 6.1.5. Passagem de arrays para funções
 - 6.2. Strings
 - 6.2.1. Strings e caracteres



- 6.2.2. Declaração e inicialização de strings
- 6.2.3. Leitura escrita e processamento de strings
- 6.2.4. Passagem de strings para funções
- 6.2.5. Biblioteca string.h

7. Modularidade (Procedimentos e Funções).

- 7.1. Abordagem top-down
- 7.2. Características das funções
- 7.3. Parâmetros de uma função
- 7.4. Instrução return
- 7.5. O tipo "void"
- 7.6. Variáveis globais e locais
- 7.7. Bibliotecas de Funções

8. Recursividade.

9. Apontadores.

- 9.1. Conceitos básicos
- 9.2. Declaração e inicialização de apontadores
- 9.3. Ponteiros e tipos de dados
- 9.4. Ponteiros e arrays
- 9.5. Aritmética de ponteiros
- 9.6. Ponteiros de ponteiros

10. Gestão dinâmica de memória.

- 10.1. Alocação e libertação de memória dinâmica
- 10.2. Utilização de memória dinâmica

11. Estruturas.

- 11.1. Conceito de estrutura.
- 11.2. Estruturas aninhadas
- 11.3. Arrays de estruturas
- 11.4. Ponteiros para estruturas
- 11.5. Passagem de estruturas para funções

Metodologias de avaliação

Frequência:

Não existe Avaliação Periódica

Aval. Contínua

Dois Testes (15% + 15%)

Aval. Final (Exame e Recurso):

-Nota Aval. Cont. (30%) ou Exame (30%)

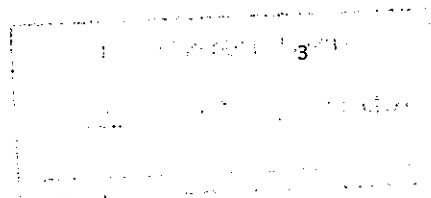
-Projeto de Programação (70%). Com discussão obrigatória

Nota mín. dos Testes e Exame: 7 val.;

Nota mín. do projeto: 10 val.

Software utilizado em aula

Visual Studio (Compilador de C)



Estágio

Não aplicável

Bibliografia recomendada

- Damas, L. (1999). *Linguagem C*. Portugal: FCA
- Kernighan, B. e Ritchie, D. (1988). *The C Programming Language*. -: Prentice Hall
- Aguilar, L. (2008). *Fundamentos de Programação: Algoritmos, Estruturas de Dados e Objetos*. -: McGraw Hill

Coerência dos conteúdos programáticos com os objetivos

A cadeira começa com os fundamentos do paradigma de programação procedimental, necessário para que os alunos resolvam problemas segundo este paradigma.

Utiliza-se a linguagem de programação C para consolidar o paradigma.

Metodologias de ensino

Aulas teórico-práticas: Exposição da matéria na componente teórica com utilização de exemplos práticos.

Resolução de problemas práticos para consolidação dos conhecimentos adquiridos.

Coerência das metodologias de ensino com os objetivos

Os conceitos teóricos são ensinados nas aulas teórico-práticas e é feita uma primeira consolidação com pequenos exercícios e demonstração de casos práticos. Será feita nas aulas práticas uma consolidação mais profunda com exercícios mais complexos.

Língua de ensino

Português

Pré-requisitos

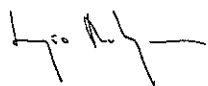
Não aplicável.

Programas Opcionais recomendados

Não aplicável.

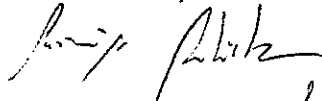
Observações

Docente Responsável



Digitally signed by Sérgio Hortas Rodrigues
DN: cn=Sérgio Hortas Rodrigues, o=IPT, ou=ESTA,
email=sergio.rodriques@ipt.pt, c=PT
Date: 2017.10.09 09:35:40 Z

Diretor de Curso, Comissão de Curso



Conselho Técnico-Científico

