

Unidade Curricular	Programação Imperativa	Área Científica	Ciências da Computação
Licenciatura em	Engenharia Informática	Escola	Escola Superior de Tecnologia e de Gestão de Bragança
Ano Letivo	2023/2024	Ano Curricular	1
Tipo	Semestral	Semestre	1
Horas totais de trabalho	162	Horas de Contacto	T - - TP 60 PL - TC - S - E - OT - O -
Nível	1-1	Créditos ECTS	6.0
Código	9119-706-1104-00-23		

T - Ensino Teórico; TP - Teórico Prático; PL - Prático e Laboratorial; TC - Trabalho de Campo; S - Seminário; E - Estágio; OT - Orientação Tutórica; O - Outra

Nome(s) do(s) docente(s) Luís Manuel Alves, Maria João Tinoco Varanda Pereira, Davide Emanuel da Silva Dias, Nelson Alexandre Perdigo Figueiredo, Pedro Gaspar Padrão Antunes Vilares

Resultados da aprendizagem e competências

No fim da unidade curricular o aluno deve ser capaz de:

1. ser capaz de estruturar um raciocínio que permita delinear um algoritmo e implementá-lo na linguagem C, para problemas computacionais de média complexidade;
2. aplicar conhecimentos fundamentais de programação em C, designadamente estruturar um programa em funções, compreender e explorar a passagem de parâmetros, e saber processar arrays e strings;
3. aplicar conhecimentos avançados de programação em C, nomeadamente utilizar apontadores e memória dinâmica, definir e utilizar estruturas e ficheiros, bem como estruturar um programa em módulos.

Pré-requisitos

Antes da unidade curricular o aluno deve ser capaz de:
Não aplicável.

Conteúdo da unidade curricular

Conceitos introdutórios sobre linguagens de programação; A linguagem C: tipos de dados e operações elementares; as instruções de seleção if, if-else e switch, e de iteração while, do-while e for; definição e uso de funções; argumentos passados por valor; vetores, arrays multidimensionais e strings; apontadores; uso de ficheiros para entrada e saída de dados; estruturas, uniões e enumerações; definição de novos tipos de dados; memória dinâmica; modularização de programas.

Conteúdo da unidade curricular (versão detalhada)

1. Conceitos introdutórios:
 - programação de computadores;
 - linguagens de programação;
 - fases de desenvolvimento de um programa;
 - a linguagem C.
2. Dados de tipo elementar:
 - tipos de dados, declaração de variáveis;
 - conceito de constante, definição de constantes simbólicas;
 - operações aritméticas, instruções, instrução de atribuição, conversões de tipo;
 - instruções de leitura e de escrita na consola.
3. Testes e condições:
 - condições e valores lógicos;
 - operadores lógicos e operadores relacionais;
 - as instruções de seleção if, if-else e switch.
4. Instruções de iteração:
 - instrução while;
 - instrução do-while;
 - instrução for.
5. Funções:
 - conceito de função e estrutura de uma função em C;
 - parâmetros passados por valor;
 - conceito de variável local/global, interna/externa e automática/estática.
6. Vetores:
 - declaração e inicialização automática de vetores;
 - passagem de vetores para funções;
 - processamento de vetores;
 - arrays multidimensionais.
7. Strings:
 - principais funções de manipulação de strings;
 - desenvolvimento de funções específicas para o tratamento de strings.
8. Apontadores:
 - noção de variável, endereço e apontador;
 - declaração e inicialização de variáveis do tipo apontador;
 - operadores de apontadores;
 - aritmética de apontadores;
 - relação entre apontadores e vetores;
 - apontadores de apontadores.
9. Ficheiros:
 - noção de ficheiro, periféricos e streams;
 - funções de manipulação de ficheiros;
 - formas de abertura de ficheiros;
 - leitura e escrita em ficheiros de texto;
 - leitura e escrita em ficheiros binários;
 - acesso sequencial e acesso direto a ficheiros;
 - deteção do final de ficheiro.
10. Estruturas, uniões e enumerações:
 - noção de estrutura, declaração e inicialização de estruturas;
 - acesso aos campos de uma estrutura;
 - passagem de estruturas para funções;
 - ficheiros de estruturas;
 - definição de tipos enumerados;
 - leitura e escrita de variáveis de tipo enumerado;
 - definição de novos tipos de dados.
11. Memória dinâmica:
 - alocação e libertação de memória;

Conteúdo da unidade curricular (versão detalhada)

- funções que retornam memória alocada dinamicamente;
 - estruturas de dados dinâmicas;
 - implementação e manipulação de listas ligadas.
12. Modularização de programas:
- Conceito de programação multi-modular;
 - Conceito de protótipo de uma função;
 - Implementação de ficheiros de cabeçalho (header files).

Bibliografia recomendada

1. Luís Damas, "Linguagem C", Tecnologias de Informação, FCA, 1999.
2. Pedro Guerreiro, "Elementos de Programação com C", Tecnologias de Informação, FCA, 2006.
3. António Rocha, "Introdução à Programação Usando C", Tecnologias de Informação, FCA, 2006.
4. Brian W. Kernighan e Dennis M. Ritchie, "The C Programming Language", Prentice-Hall, 1988.
5. R. Johnsonbaugh, and M. Kalin, "C for Scientists and Engineers", Prentice-Hall, 1997.

Métodos de ensino e de aprendizagem

O método de ensino utilizado nas aulas teóricas é o expositivo, que possibilita a transmissão de conhecimentos com continuidade e com um dispêndio mínimo de tempo. Nas aulas práticas, o método mais utilizado é o ativo, suscitando dessa forma a atividade dos alunos através da resolução de exercícios práticos. Espera-se ainda que o aluno realize um conjunto de tarefas nas horas não presenciais.

Alternativas de avaliação

1. Alternativa 1 - (Ordinário, Trabalhador) (Final)
 - Prova Intercalar Escrita - 30% (Primeiro Teste Prático realizado em computador.)
 - Prova Intercalar Escrita - 30% (Segundo Teste Prático realizado em computador.)
 - Prova Intercalar Escrita - 40% (Terceiro Teste Prático realizado em computador. A realizar na Época de Avaliação Final.)
2. Alternativa 2 - (Ordinário, Trabalhador) (Recurso, Especial)
 - Exame Final Escrito - 100%

Língua em que é ministrada

1. Português
2. Inglês

Validação Eletrónica

Lúis Manuel Alves, Maria João Tinoco Varanda Pereira	Tiago Miguel Ferreira Guimaraes Pedrosa	Luísa Maria Garcia Jorge	José Carlos Rufino Amaro
10-10-2023	25-10-2023	25-10-2023	31-10-2023