

Curso profissional de Técnico de Programador/a de Informática
Planificação Anual de Técnicas e Linguagens de Programação – 10.º ano
Ano Letivo 2024/2025

Turma: 10.º Q

Professor: Sérgio Gonçalves

1 - Estrutura e Finalidades da disciplina

A disciplina de Técnicas e Linguagens de Programação insere-se na componente técnica do Curso Profissional de Técnico de Programador/a de Informática. A disciplina está organizada no 10º ano em 4 unidades de formação de curta duração independentes, num total de 150 horas, funcionando numa distribuição de 2+2+2 segmentos de 45 minutos por semana.

Como linguagem de programação foi aprovada pelo agrupamento disciplinar o uso da linguagem C, e como ferramenta de desenvolvimento o ambiente de desenvolvimento integrado Geany.

2- Planificação

O referencial de formação poderá ser consultado na formação técnica do curso profissional de Técnico de Programador/a de Informática, do item programas no sítio da Agência Nacional para a Qualificação - <http://www.anqep.gov.pt/>

A planificação seguinte foi aprovada pelo Grupo de Recrutamento de Informática em 24.09.2024.

Período	Tópicos Programáticos	Horas (tempos)
1.º Período De 18/09 a 15/12	UFCD 0804 – Algoritmos	25h (34 tempos)
		19/09 a 31/10
	1. Linguagem estruturada: 1.1. Estruturas lógicas; 1.2. Desenho das estruturas diagramáticas; 1.3. Sintaxe da linguagem; 2. Construção de um algoritmo: 2.1. Noções de ação e estado da ação; 2.2. Ações e a sua sintaxe; 2.3. Verbos; 2.4. Sintaxe; 2.5. Alinhamento das frases; 3. Estruturas lógicas básicas e o seu controlo: 3.1. Estrutura sequencial, alternativa e repetitiva; 3.2. Condições e regras de inicialização e alteração; 3.3. Estruturas diagramáticas como representação algorítmica; 4. Tipos de dados em algoritmia: 4.1. Constantes e Variáveis; 5. Descrição da entrada e saída de dados: 5.1. Elementos de linguagem; 6. Técnicas de construção de algoritmos: 6.1. Contadores; 6.2. Totalizadores 6.3. Expressões aritméticas; 6.4. Funções predefinidas; 6.5. Validação de dados; 7. Construção de procedimentos: 7.1. Procedimentos de entrada e saída e chamada de procedimentos.	0,75h (1 tempo) 1,50h (2 tempos) 0,75h (1 tempo) 0,75h (1 tempo) 0,75h (1 tempo) 0,75h (1 tempo) 0,75h (1 tempo) 2,25h (3 tempos) 0,75h (1 tempo) 0,75h (1 tempo) 1,50h (2 tempos) 1,50h (2 tempos) 0,75h (1 tempo) 0,75h (1 tempo) 0,75h (1 tempo) 1,50h (2 tempos) 1,50h (2 tempos) 0,75h (1 tempo) 0,75h (1 tempo) 0,75h (1 tempo) 1,50h (2 tempos) 1,50h (2 tempos) 2,25h (3 tempos)
	Avaliação	3,75h (5 tempos)
	UFCD 0809 – Programação em C/C++ – fundamentos	50h (67 tempos)
	02/11 a 08/02	
	1. Estrutura de um programa em C/C++: 1.1. Função main (); 1.2. Estrutura de um programa; 2. Dados em C: 2.1. Variáveis e constantes; 2.2. Tipos de dados; 2.3. Variáveis de ponto flutuante; 3. Estudo e emprego da função printf(): 3.1. Constantes e variáveis; 3.2. Formatação; 3.3. Sequências de escape; 3.4. Carateres gráficos; 3.5. Valores de virgula-flutuante; 3.6. Vários argumentos em printf(); 4. Cadeia de carateres e entrada e saída de dados formatados: 4.1. Uma cadeia de carateres; 4.2. Função scanf(); 4.3. Códigos de formatação e modificação para a função scanf(); 4.4. Operador de endereço &; 4.5. Função strlen();	0,75h (1 tempo) 2,25h (3 tempos) 1,50h (2 tempos) 0,75h (1 tempo) 0,75h (1 tempo) 0,75h (1 tempo) 0,75h (1 tempo) 0,75h (1 tempo) 1,50h (2 tempos) 0,75h (1 tempo) 0,75h (1 tempo) 0,75h (1 tempo) 0,75h (1 tempo) 1,50h (2 tempos) 0,75h (1 tempo) 0,75h (1 tempo) 0,75h (1 tempo) 0,75h (1 tempo) 0,75h (1 tempo)

Período	Tópicos Programáticos	Horas (tempos)
2.º Período De 03/01 a 22/03	4.6. Diretiva #define;	0,75h (1 tempo)
	4.7. Funções getche(), getch(), getchar() e putchar();	0,75h (1 tempo)
	5. Operadores e expressões em C/C++:	
	5.1. Operadores e expressões aritméticas;	0,75h (1 tempo)
	5.2. Operadores de atribuição +=, -=, *=, /= e %=;	0,75h (1 tempo)
	5.3. Operadores cast;	0,75h (1 tempo)
	5.4. Operadores relacionais;	0,75h (1 tempo)
	6. Estruturas repetitivas em C/C++:	
	6.1. Implementação com a instrução while;	1,50h (2 tempos)
	6.2. Implementação com a instrução for;	1,50h (2 tempos)
6.3. Implementação com a instrução do while;	0,75h (1 tempo)	
7. Estruturas repetitivas complexas:		
7.1. Estruturas matriciais;	1,50h (2 tempos)	
7.2. Condições de controlo;	0,75h (1 tempo)	
8. Estruturas alternativas em C/C++:		
8.1. Implementação com a instrução if, if else e switch;	1,50h (2 tempos)	
8.2. Condições e os operadores lógicos;	0,75h (1 tempo)	
8.3. Alternativas múltiplas;	0,75h (1 tempo)	
8.4. Instruções break, continue e goto;	0,75h (1 tempo)	
9. Funções:		
9.1. Estrutura e argumentos de uma função;	1,50h (2 tempos)	
9.2. Variáveis locais;	0,75h (1 tempo)	
9.3. Funções recursivas;	1,50h (2 tempos)	
9.4. Classes de armazenamento;	1,50h (2 tempos)	
10. Matrizes:		
10.1. Declaração e inicialização de matrizes;	1,50h (2 tempos)	
10.2. Armazenamento e leitura de dados de uma matriz;	1,50h (2 tempos)	
10.3. Matrizes unidimensionais/multidimensionais;	0,75h (1 tempo)	
10.4. Matrizes como argumento de funções;	0,75h (1 tempo)	
11. Cadeias de caracteres (string):		
11.1. Constantes;	0,75h (1 tempo)	
11.2. Variáveis;	0,75h (1 tempo)	
11.3. Matriz de strings;	0,75h (1 tempo)	
11.4. Leitura e escrita de strings.	1,50h (2 tempos)	
	Avaliação	7,50h (10 tempos)
	UFCD 0810 – Programação em C/C++ – avançada	50h (67 tempos)
		09/02 a 14/05
	1. Apontadores:	
	1.1. Definição de apontadores;	0,75h (1 tempo)
	1.2. Declaração de apontadores;	1,50h (2 tempos)
	1.3. Apontadores para passar dados para uma função;	0,75h (1 tempo)
	1.4. Operador indireto (*);	0,75h (1 tempo)
	1.5. Atribuição de valores a variáveis apontadores;	1,13h (1,5 tempos)
	1.6. Operações com apontadores;	3,00h (4 tempos)
	1.7. Apontadores para matrizes;	0,38h (0,5 tempos)
	1.8. Apontadores para strings;	0,38h (0,5 tempos)
	1.9. Apontadores para apontadores;	0,38h (0,5 tempos)
	1.10. Apontadores para funções apontadores como argumento de uma função;	0,75h (1 tempo)
	2. Estruturas:	
	2.1. Definição e declaração de estruturas;	1,50h (2 tempos)
	2.2. Inicialização de estruturas;	0,75h (1 tempo)
	2.3. Leitura e escrita de valores nos elementos de uma estrutura;	1,50h (2 tempos)
	2.4. Matrizes de estruturas;	0,75h (1 tempo)
	2.5. Apontadores para estruturas;	0,38h (0,5 tempos)

Período	Tópicos Programáticos	Horas (tempos)
3.º Período De 08/04 a 25/06	2.6. Estruturas em lista ligada; 2.7. Declaração de listas ligadas; 2.8. Operações sobre listas ligadas; 3. Uniões: 3.1. Definição de uniões; 3.2. Declaração de uniões; 3.3. Inicialização de uniões; 3.4. Uniões de estruturas; 3.5. Problemas com uniões; 4. Impressão de caracteres por acesso direto à memória: 4.1. Sistema binário; 4.2. Sistema hexadecimal; 4.3. Sistema octal; 4.4. Operadores sobre os bits; 4.5. Conversão entre sistemas; 5. Campos bit: 5.1. Pré-processador C/C++ e as diretivas; 5.2. Diretiva #define; 5.3. Macros; 5.4. Macros e funções – comparação; 5.5. Diretiva #include; 5.6. Diretivas #undef, #if, #ifdef, #ifndef, #else, #endif e #error; 6. Livraria do C/C++: 6.1. Acesso à livraria; 6.2. #include; 6.3. Livraria math-#include; 7. Alocação e desalocação de memória: 7.1. Função malloc(); 7.2. Função calloc(); 7.3. Função free(); 8. Ficheiros em C/C++: 8.1. Níveis de leitura e escrita em ficheiros; 8.2. Abertura e fecho de ficheiros; 8.3. Leitura e escrita em ficheiros; 8.4. Condições de erro; 8.5. Buffers; 8.6. Outras funções para manipular ficheiros.	0,75h (1 tempo) 1,13h (1,5 tempos) 1,50h (2 tempos) 0,38h (0,5 tempos) 0,75h (1 tempo) 0,38h (0,5 tempos) 0,75h (1 tempo) 2,25h (3 tempos) 0,38h (0,5 tempos) 0,38h (0,5 tempos) 0,38h (0,5 tempos) 0,75h (1 tempo) 1,13h (1,5 tempos) 1,50h (2 tempos) 0,75h (1 tempo) 1,13h (1,5 tempos) 0,75h (1 tempo) 0,75h (1 tempo) 1,13h (1,5 tempos) 0,38h (0,5 tempos) 0,38h (0,5 tempos) 0,75h (1 tempo) 0,75h (1 tempo) 0,75h (1 tempo) 0,75h (1 tempo) 0,75h (1 tempo) 0,75h (1 tempo) 1,50h (2 tempos) 1,50h (2 tempos) 0,75h (1 tempo) 2,25h (3 tempos)
	Avaliação	7,50h (10 tempos)
	UFCD 0805 – Estruturas de Dados	25h (34 tempos)
		16/05 a 25/06 1. Introdução às estruturas de dados: 1.1. Estruturas predefinidas; 1.2. Estruturas construídas; 2. Estrutura de lista linear: 2.1. Desenho de estruturas; 2.2. Algoritmos de manipulação; 2.3. Acesso, instalação, supressão e atualização de elementos; 2.4. Pesquisas seletivas; 2.5. Concatenação de listas; 2.6. Explosão de listas; 2.7. Fusão de listas ordenadas; 3. Estrutura matricial unidimensional: 3.1. Desenho de estruturas; 3.2. Algoritmos de manipulação; 3.3. Percurso; 3.4. Inserção;

Período	Tópicos Programáticos	Horas (tempos)
	3.5. Atualização; 3.6. Ordenação; 4. Estrutura matricial multidimensional: 4.1. Desenho de estruturas; 4.2. Algoritmos de manipulação; 4.3. Percurso; 4.4. Inserção; 4.5. Atualização; 4.6. Ordenação; 4.7. Aplicações práticas das estruturas; 5. Estrutura de dados lista ligada: 5.1. Desenho de estruturas; 5.2. Algoritmos de manipulação; 5.3. Percurso; 5.4. Acesso, inserção e supressão de um elemento; 5.5. Pesquisa seletiva; 5.6. Algoritmos com listas ligadas bidimensionais; 5.7. Algoritmos com listas ligadas com elementos de matrizes; 6. Estruturas filas de espera e pilhas: 6.1. Implementação de filas de espera; 6.2. Estrutura pilha e os algoritmos de manipulação; 7. Estrutura de árvore binária: 7.1. Desenho de estruturas; 7.2. Algoritmos de manipulação; 7.3. Percurso: ordem central, pré ordem e pós ordem; 7.4. Acesso, inserção e supressão de um elemento.	0,19h (0,25 tempos) 0,38h (0,5 tempos) 0,38h (0,5 tempos) 0,75h (1 tempo) 0,38h (0,5 tempos) 0,19h (0,25 tempos) 0,19h (0,25 tempos) 0,38h (0,5 tempos) 0,75h (1 tempo) 0,38h (0,5 tempos) 0,75h (1 tempo) 0,38h (0,5 tempos) 0,38h (0,5 tempos) 0,38h (0,5 tempos) 0,38h (0,5 tempos) 0,75h (1 tempo) 0,75h (1 tempo) 0,38h (0,5 tempos) 0,75h (1 tempo) 0,38h (0,5 tempos) 0,38h (0,5 tempos)
	Avaliação	3,75h (5 tempos)

Datas previstas de início e fim dos módulos:

	UFCD 0804		UFCD 0809		UFCD 0810		UFCD 0805	
	início	fim	início	fim	início	fim	início	fim
10.º Q – 1º T	16.set	23.out	24.out	29.jan	29.jan	08.mai	12.mai	18.jun
10.º Q – 2º T	16.set	22.out	24.out	27.jan	27.jan	06.mai	08.mai	17.jun