

CURSO PROFISSIONALTÉCNICO PROGRAMADOR/A DE INFORMÁTICA

DISCIPLINA DE PROGRAMAÇÃO MOBILE 10º ANO

CRITÉRIOS DE AVALIAÇÃO

1. COMPETÊNCIAS GERAIS

Conhecimentos, capacidades e atitudes.

2. DOMÍNIOS e DOMÍNIOS ESPECÍFICOS/TEMAS

DOMÍNIOS	DOMÍNIOS ESPECÍFICOS/TEMAS	PONDERAÇÃO
INFORMAÇÃO E COMUNICAÇÃO	<ul style="list-style-type: none"> ▪ Utiliza instrumentos diversificados para pesquisar, avaliar, validar e mobilizar informação, de forma crítica e autónoma, transformando a informação em conhecimento. ▪ Comunica sempre de forma clara e rigorosa, utilizando corretamente diferentes linguagens e meios de comunicação. 	5%
CONHECIMENTO	<p>PROGRAMAÇÃO MOBILE</p> <ul style="list-style-type: none"> • UFCD 0809 - Programação em C/C++ - Fundamentos 	70%
	<p>PROGRAMAÇÃO MOBILE</p> <ul style="list-style-type: none"> • UFCD 0810 - Programação em C/C++ - Avançado. 	
RESOLUÇÃO DE PROBLEMAS	<ul style="list-style-type: none"> ▪ Gere projetos e toma decisões, revela capacidades de planeamento e execução. ▪ Avalia os problemas e cria soluções inovadoras. 	10%
CRIATIVIDADE	<ul style="list-style-type: none"> ▪ Desenvolve ideias e projetos contextualizados, recorrendo à imaginação. ▪ Demonstra múltiplas soluções sustentáveis para a resolução de um problema. ▪ Amplia o conhecimento a várias áreas de atuação e está aberto novas experiências. 	5%
DESENVOLVIMENTO PESSOAL E INTERPESSOAL	<ul style="list-style-type: none"> ▪ Comporta-se de forma exemplar, revela responsabilidade. ▪ Autonomia nas diferentes atividades de aprendizagem, manifesta uma atitude proativa. ▪ Trabalha em equipa; revela empatia, tolerância e capacidade de negociação para o bem comum. ▪ Consolida e aprofunda competências autorreguladoras numa perspetiva de aprendizagem ao longo da vida. 	10%

3. NÍVEIS E DESCRITORES DE DESEMPENHO

DESCRITORES DE DESEMPENHO
<ul style="list-style-type: none"> ▪ Apresenta pontualidade e assiduidade ▪ Cumpre as normas de funcionamento da sala de aula ▪ Apresenta um comportamento adequado na relação com o outro, expressando-se de forma ajustada a diferentes contextos, colaborando com os outros de forma regular ▪ Adquire e mobiliza sistematicamente o conhecimento técnico na resolução de problemas ▪ Participa ativamente em projetos/atividades da turma/PAA ▪ Desenvolve uma autonomia crescente, motivando-se para a aprendizagem, promovendo a autorregulação, o espírito de iniciativa e a gestão eficiente de tarefas ▪ Apresenta capacidade de pesquisa e inovação, respondendo adequada e originalmente na resolução de problemas

<p>PROGRAMAÇÃO MOBILE</p> <ul style="list-style-type: none"> • UFGD 0809 - Programação em C/C++ - Fundamentos 	<ul style="list-style-type: none"> ▪ Conhece situações de aplicação da linguagem C. ▪ Conhece um ambiente de programação da linguagem C. <ul style="list-style-type: none"> ○ Estrutura de um programa em C/C++: Função <i>main()</i>; Estrutura de um programa. ○ Dados em C: Variáveis e constantes; Operadores lógicos; Tipos de dados; Variáveis de ponto flutuante; Declaração; Atribuição; Instruções de entrada e de saída. ○ Estudo e emprego da função <i>printf()</i>: Constantes e variáveis; Formatação; Sequências de escape; Carateres gráficos; Valores de virgula-flutuante; Vários argumentos em <i>printf()</i>. ▪ Conhece os conceitos fundamentais das linguagens de programação. <ul style="list-style-type: none"> ○ Cadeia de carateres e entrada e saída de dados formatados: Uma cadeia de carateres; Função <i>scanf()</i>; Códigos de formatação e modificação para a função <i>scanf()</i>; Operador de endereço &; Função <i>strlen()</i>; Diretiva <i>#define</i>; Funções <i>getche()</i>, <i>getch()</i>, <i>getchar()</i> e <i>putchar()</i>. ○ Operadores e expressões em C/C++: Operadores e expressões aritméticas; Operadores de atribuição <i>+=</i>, <i>-=</i>, <i>/=</i> e <i>%=</i>; Operadores <i>cast</i>; Operadores relacionais. ▪ Conhece as principais estruturas de controlo de um programa. <ul style="list-style-type: none"> ○ Estruturas repetitivas em C/C++: Implementação com a instrução <i>while</i>; Implementação com a instrução <i>for</i>; Implementação com a instrução <i>do while</i> ○ Estruturas repetitivas complexas: Estruturas matriciais e Condições de controlo. ○ Estruturas alternativas em C/C++: Implementação com a instrução <i>if</i>, <i>if else</i> e <i>switch</i>; Condições e os operadores lógicos; Alternativas múltiplas; Instruções <i>break</i>, <i>continue</i> e <i>goto</i>. ○ Funções: Estrutura e argumentos de uma função; Variáveis locais; Funções recursivas; Classes de armazenamento ○ Matrizes: Declaração e inicialização de matrizes; Armazenamento e leitura de dados de uma matriz; Matrizes unidimensionais/multidimensionais; Matrizes como argumento de funções. ○ Cadeias de carateres (<i>string</i>): Constantes; Variáveis; Matriz de strings; Leitura e escrita de strings ▪ Aplica os conceitos a exercícios práticos de programação.
<p>PROGRAMAÇÃO MOBILE</p> <ul style="list-style-type: none"> • UFGD 0810 - Programação em C/C++ - Avançado. 	<ul style="list-style-type: none"> ▪ Aplica os conceitos de linguagem de programação C em exercícios práticos. <ul style="list-style-type: none"> ○ Apontadores: Definição de apontadores; Declaração de apontadores; Apontadores para passar dados para uma função; Operador indireto (*); Atribuição de valores a variáveis apontadores; Operações com apontadores; Apontadores para matrizes; Apontadores para strings; Apontadores para apontadores; Apontadores para funções apontadores como argumento de uma função. ○ Estruturas: Definição e declaração de estruturas; Inicialização de estruturas; Leitura e escrita de valores nos elementos de uma estrutura; Matrizes de estruturas; Apontadores para estruturas; Estruturas em lista ligada; Declaração de listas ligadas; Operações sobre listas ligadas.

	<ul style="list-style-type: none"> ○ Uniões: Definição de uniões; Declaração de uniões; Inicialização de uniões; Uniões de estruturas; Problemas com uniões. ○ Impressão de caracteres por acesso direto à memória: Sistema binário; Sistema hexadecimal; Sistema octal; Operadores sobre os bits; Conversão entre sistemas. ○ Campos bit: Pré-processor C/C++ e as diretivas; Diretiva #define; Macros; Macros e funções - comparação; Diretiva #include; Diretivas #undef, #if, #ifdef, #ifndef, #else, #endif e #error ○ Livraria do C/C++: Acesso à livraria; #include; Livraria Math-#include ○ Alocação e desalocação de memória: Função malloc(); Função calloc(); Função free() ○ Ficheiros em C/C++: Níveis de leitura e escrita em ficheiros; Abertura e fecho de ficheiros; Leitura e escrita em ficheiros; Condições de erro; Buffers; Outras funções para manipular ficheiros ▪ Sabe identificar as estruturas de dados mais indicadas num dado exercício. ▪ Sabe identificar as estruturas de controlo mais indicadas para um dado exercício.
--	---

NÍVEIS DE DESEMPENHO			
Muito Bom	Bom	Suficiente	Insuficiente
Sempre ou quase sempre 18-20	Na maior parte dos casos 14-17	Com alguma regularidade 10-13	Poucas vezes 0-9

4. INSTRUMENTOS DE AVALIAÇÃO

- Fichas de avaliação
- Trabalhos individuais/de grupo
- Trabalho de projeto
- Inquéritos/Google Forms
- Grelha de observação de aula
- Registo de ocorrências no INOVAR
- Ficha de autoavaliação da UFCD

Nota: Cada instrumento de avaliação poderá ter ponderações diferentes conforme o grau de dificuldade e/ou a abrangência de conteúdos. Os alunos serão informados acerca das respetivas ponderações e conteúdos que serão traduzidas através de grelhas de classificação a elaborar no final de cada módulo/UFCD.

NOTA FINAL:

Estes critérios de avaliação foram definidos tendo como balizas o Perfil do Aluno à Saída da Escolaridade Obrigatória, as Aprendizagens Essenciais/Referencial de Formação e o Projeto Educativo do Agrupamento.