

Programação 2024/25

LEI, LEI-PL, LEI-CE

Desafio 2

Data Limite para Submissão no InforEstudante: **29 de março de 2025 às 20.00**

Regras para a submissão do desafio

É obrigatório que cumpra todas as regras de submissão, caso contrário o seu trabalho poderá não ser avaliado.

1. A data-limite para submissão é o dia **29 de março de 2025 às 20.00**. Não serão aceites trabalhos depois dessa hora. Recomendamos que não efetuem a submissão final muito perto da hora limite.
2. Só serão aceites submissões efetuadas através do InforEstudante, na atividade criada para este efeito.
3. Devem submeter apenas um ficheiro. É o ficheiro *funcao.c*, um dos que integra o projeto que foi divulgado juntamente com o enunciado. Não devem submeter os restantes ficheiros. No ficheiro *funcao.c* devem efetuar as seguintes alterações:
 - a. Alterar o nome do ficheiro, que deve passar a ser o do vosso número de aluno e mantendo a extensão *.c*. Caso o vosso número de aluno seja 1234567, devem submeter um ficheiro com nome 1234567.c
 - b. Quando efetuarem a alteração da alínea anterior, tenham cuidado para não alterar o nome dos ficheiros de cabeçalho (*header files*) que surgem no início do código. O *refactoring* do CLion altera o nome dos ficheiros *.h* e isso não pode acontecer no ficheiro que submeterem. Confiram a linha 2 do ficheiro que deve continuar a ser `#include "funcao.h"`.
 - c. Completar a vossa identificação nas linhas 4 e 5 do ficheiro.
 - d. Escrever a função *desafio2()*, de acordo com o que é solicitado no enunciado. Esta função não deve escrever nada na consola, efetuando apenas as operações descritas no enunciado. Caso julgue relevante, pode criar funções auxiliares dentro deste ficheiro *.c*.
 - e. Os restantes ficheiros do projeto não podem ser alterados. Deve garantir que o ficheiro que entrega continua compatível com o resto do projeto.

Programação 2024/25

LEI, LEI-PL, LEI-CE

Desafio 2 – Enunciado

Considere as seguintes definições de tipos de dados:

```
// Permite armazenar uma data (dia, mês, ano)
typedef struct{
    int dia, mes, ano;
} data;

// Permite armazenar informação sobre um projeto
typedef struct dados projeto;

struct dados{
    data inicio, final;      // Datas de início e final do projeto
    int duracao;             // Duração do projeto em dias
    char pal[4][20];         // 4 palavras-chave associadas ao projeto
    int valor;               // Orçamento total do projeto
};
```

Um projeto é caracterizado pelas seguintes componentes:

1. Datas de início e final. Ao responder a este desafio, pode assumir que todos os projetos começam e acabam **no mesmo ano civil**.
2. Quatro palavras-chave que definem o âmbito do projeto. Todos os projetos têm exatamente 4 palavras-chave. As palavras estão escritas em **letra minúscula** e nenhuma palavra tem mais de 19 caracteres.

Estas componentes permitem calcular a duração e o valor do orçamento de cada projeto (consultar detalhes a seguir).

Escreva a função:

```
void desafio2(projeto a[], int tam);
```

A função recebe 2 parâmetros:

1. Endereço de uma tabela de estruturas do tipo *projeto* (*a*)
2. Dimensão da tabela de estruturas (*tam*)

Programação 2024/25

LEI, LEI-PL, LEI-CE

As estruturas armazenadas na tabela têm os campos *inicio*, *final* e *pal* completamente preenchidos. Para cada uma destas estruturas, a função deve preencher os campos *duracao* e *valor*, de acordo com as seguintes regras:

1. Colocar no campo *duracao* o número de dias que decorreram entre o início e o final do projeto (todos os projetos começam e terminam no mesmo ano civil).
2. Colocar no campo *valor* o orçamento total do projeto. O orçamento é calculado tendo em consideração as palavras-chave do projeto e deve ser efetuado da seguinte forma:
 - a. Palavras-chave que tenham exatamente 2 vogais, uma no início e outra no final da palavra, contribuem com 10 para o orçamento;
 - b. Palavras-chave que tenham 1 vogal contribuem com 5 para o orçamento;
 - c. Palavras-chave que tenham mais de 3 vogais contribuem com 1 para o orçamento.
 - d. As restantes palavras-chave contribuem com 0 para o orçamento

O valor total do orçamento de um projeto obtém-se somando as contribuições das suas palavras-chave. Pode assumir que todas as palavras estão em letras minúsculas.

Exemplos das alterações pretendidas:

1. Esta estrutura:

$\{\{12,2,2024\}, \{20,2,2024\}, 0, \{\text{"obra"}, \text{"cimento"}, \text{"cal"}, \text{"areia"}\}, 0\}$

Passa a ter o seguinte conteúdo:

$\{\{12,2,2024\}, \{20,2,2024\}, 8, \{\text{"obra"}, \text{"cimento"}, \text{"cal"}, \text{"areia"}\}, 16\}$

2. Esta estrutura:

$\{\{20,4,2024\}, \{18,6,2024\}, 0, \{\text{"papel"}, \text{"tinta"}, \text{"caneta"}, \text{"ar"}\}, 0\}$

Passa a ter o seguinte conteúdo:

$\{\{20,4,2024\}, \{18,6,2024\}, 59, \{\text{"papel"}, \text{"tinta"}, \text{"caneta"}, \text{"ar"}\}, 5\}$

3. Esta estrutura:

$\{\{31,7,2024\}, \{1,8,2024\}, 0, \{\text{"cal"}, \text{"pregos"}, \text{"fio"}, \text{"alicate"}\}, 0\}$

Passa a ter o seguinte conteúdo:

$\{\{31,7,2024\}, \{1,8,2024\}, 1, \{\text{"cal"}, \text{"pregos"}, \text{"fio"}, \text{"alicate"}\}, 6\}$