

Programação Imperativa - PBL C7

Exercício 1 (valor: 2,0 pontos): Escreva uma função na linguagem C com o seguinte protótipo:

int contar(char* texto, char chave)

A função deve retornar a quantidade de ocorrências do caracter **chave** no **texto** fornecido como parâmetro.

Deve ser escrita uma função **main** para verificar o funcionamento da função **contar**.

Exercício 2 (valor: 2,0 pontos): Escreva uma função na linguagem C com o seguinte protótipo:

int calcular_hash(char* texto, int modulo)

A função deve calcular o valor de *hash* correspondente ao **texto** fornecido como parâmetro. O cálculo deve, primeiramente, fazer a somatória dos valores ASCII de todos os caracteres do **texto** e, em seguida, obter o resto da divisão pelo **módulo** fornecido como parâmetro.

Deve ser escrita uma função **main** para verificar o funcionamento da função **calcular_hash**. O valor fornecido como módulo deve estar no intervalo $[10^6, 10^7]$.

Exercício 3 (valor: 2,0 pontos): Escreva uma função na linguagem C com o seguinte protótipo:

typedef struct { char nome[40]; int idade; } Pessoa;
void preencher(Pessoa* p)

A função deve preencher os campos da estrutura recebida como parâmetro a partir de dados fornecidos pelo usuário via teclado.

Deve ser escrita uma função **main** para verificar o funcionamento da função **preencher**.

Exercício 4 (valor: 4,0 pontos): Escreva uma função na linguagem C com o seguinte protótipo:

void transpor(double* matriz_ptr, int linhas, int colunas, double* transposta_ptr)

A função deve gerar a matriz transposta de uma matriz de valores reais recebida como parâmetro. O primeiro parâmetro aponta para o elemento da matriz na sua primeira linha e primeira coluna. O segundo parâmetro indica a quantidade de linhas da matriz original, enquanto o segundo parâmetro indica a quantidade de colunas. O quarto parâmetro indica onde deve ser armazenada a matriz transposta.