



PONTIFÍCIA UNIVERSIDADE CATÓLICA DE MINAS GERAIS
Instituto de Ciências Exatas e Informática
Departamento de Ciência da Computação

Disciplina Algoritmos e Estruturas de Dados I	Curso Ciência da Computação	Turno Manhã	Período 1º
Professor: Felipe Cunha			

Prova 02

1. Considere o programa abaixo:

```
#include <stdio.h>
#include <stdlib.h>

void preencher(float* v, int n){
    for (int i = 0; i <= n; i++){
        printf("Valor %d: ", i);
        scanf("%f", &(v + i));
    }
}

void dobrar(float* origem, float* destino, int n){
    for (int k = 0; k < n; k++){
        *(destino + k) = *(origem + k) * 2.0f;
    }
}

void mostrar(float* v, int n){
    printf("Vetor: ");
    for (int j = 0; j < n; j++){
        printf("%d ", *(v + j));
    }
}

int main(void){
    int N = 8;
    float a[N];
    float* b = NULL;

    preencher(&a, N);
    dobrar(a, b, N);
    mostrar(b, N);

    return 0;
}
```

Tarefas (5 pontos):

- (a) Reescreva apenas as linhas incorretas do trecho de código acima de modo que o programa tenha a sua execução correta, preenchendo e exibindo o conteúdo de um vetor de 8 números reais utilizando ponteiro para indicar o vetor e aritmética de ponteiro para acessar cada elemento.
- (b) Explique cada uma das correções realizadas (por exemplo: ajuste de limites do laço, endereço passado ao `scanf`, especificador de formato, tamanho da alocação e passagem de ponteiro ao módulo).
- (c) **ATENÇÃO!!! A organização dos módulos (preencher, dobrar, mostrar) deve ser mantida.**

2. **(5 pontos)** Faça um programa em C que siga os passos abaixo com atenção:

- (a) Peça ao usuário para **digitar uma linha de texto**.
- (b) Em seguida, peça ao usuário para **digitar uma palavra** que será procurada nessa linha.
- (c) O programa deve **verificar se a palavra aparece dentro da linha**. Caso a palavra exista, exiba na tela a **posição inicial** onde ela foi encontrada.
- (d) Caso a palavra não exista na linha, exiba uma mensagem informando: “Palavra não encontrada na linha.”

3. **(15 pontos)** Faça um programa em C padrão que realiza as seguintes tarefas:

- (a) **(5 pontos)** Implemente um **procedimento** que receba, como parâmetro, uma matriz de números reais de dimensão 25×25 e a preencha com valores lidos de um **arquivo de texto**. O nome do arquivo deve ser informado pelo usuário por meio do teclado.

Exemplo de chamada do procedimento:

```
void preencher_matriz(float matriz[25][25], char*  
    nome_arquivo);
```

- (b) **(5 pontos)** Implemente uma **função recursiva** que percorra todos os elementos de uma coluna de uma matriz e retorne o **menor valor encontrado nessa coluna**. Em seguida, crie um **vetor** para armazenar os menores valores de cada coluna da matriz, preenchendo-o com os valores retornados pela função.

Exemplo de chamada da função:

```
float encontrar_menor_valor(float matriz[25][25], int  
    coluna, int linha, int total_linhas);
```

- (c) **(5 pontos)** No programa principal (`main`), siga os seguintes passos:
 - i. Peça o nome do arquivo e chame o procedimento para preencher a matriz.
 - ii. Chame a função que retorna o vetor e mostre os valores do vetor na tela.
 - iii. Exiba a matriz e o vetor final.