

PROVA 2

Os exercícios a seguir devem ser resolvidos utilizando a Linguagem C e devem ser submetidos para correção conforme orientações do professor.

Exercício 1 (4,00)

Crie um programa que informa qual o maior elemento presente em alguma diagonal de uma matriz quadrada que possui apenas números positivos. Para isso, utilize funções, ponteiro para função e alocação dinâmica.

Inicialmente, crie uma função que receba os índices de um elemento da matriz e sua dimensão, e retorne 1 caso os índices compreendam uma posição de diagonal principal e 0 caso contrário. A função deve ter a seguinte assinatura:

```
int diagonal_principal(int i, int j, int n);
```

Crie uma função semelhante, mas para a diagonal secundária, com a seguinte assinatura:

```
int diagonal_secundaria(int i, int j, int n);
```

Crie uma função que recebe um ponteiro para matriz, sua dimensão e um ponteiro para uma função que decidirá qual diagonal será considerada. Sua função deve percorrer a matriz e detectar qual o maior elemento presente na diagonal indicada pelo ponteiro de função. Sua função deve ter a seguinte assinatura:

```
int maior_diagonal(int** matriz, int n, int (*diagonal)(int, int, int));
```

Como auxílio, utilize a seguinte função que aloca dinamicamente uma matriz $n \times n$:

```
int** aloca_matriz(int n) {  
    int** matriz = (int**) malloc(n * sizeof(int*));  
  
    for(int i = 0; i < n; i++) {  
        matriz[i] = (int*) malloc(n * sizeof(int));  
    }  
  
    return matriz;  
}
```

Na main, seu programa deve ler do usuário o inteiro n que indica a dimensão da matriz quadrada, deve alocar a matriz usando a função dada, e ainda ler da entrada os elementos da matriz. Depois, seu programa deve imprimir, nesta ordem, o maior valor da diagonal principal e o maior valor da diagonal secundária, utilizando as funções criadas.

Exemplo de entrada 1:

```
3
1 2 9
4 3 2
1 0 0
```

Exemplo de saída 1:

```
3 9
```

Exemplo de entrada 2:

```
4
1 0 0 0
0 1 0 0
0 0 1 0
0 0 0 1
```

Exemplo de saída 2:

```
1 0
```

Exercício 2 (3,00)

Escreva uma função que recebe um inteiro n como parâmetro, aloca dinamicamente um vetor de n elementos inteiros e, por fim, retorna o ponteiro para o primeiro elemento do vetor. A função deve ter o seguinte cabeçalho:

```
int* cria_vetor(int n);
```

Escreva uma função um vetor e seu tamanho e faça a leitura dos seus elementos da entrada padrão. A função deve ter o seguinte cabeçalho:

```
void le_vetor(int *v, int n);
```

Escreva uma função um vetor e seu tamanho e imprima os elementos na saída padrão. A função deve ter o seguinte cabeçalho:

```
void imprime_vetor(int *v, int n);
```

Na `main`, seu programa deve ler o inteiro n do usuário, alocar, ler e imprimir o vetor utilizando as funções criadas anteriormente. Ao final, o programa deve desalocar o vetor.

Exemplo de entrada 1:

```
4
1 2 3 4
```

Exemplo de saída 1:

```
1 2 3 4
```

Exemplo de entrada 2:

```
5
2 4 6 8 10
```

Exemplo de saída 2:

```
2 4 6 8 10
```

Exercício 3 (3,00)

Responda, com suas palavras, às questões a seguir.

- Explique o conceito de bibliotecas estáticas e bibliotecas dinâmicas na Linguagem C, destacando as diferenças entre elas. Descreva também as vantagens e desvantagens de cada abordagem.
- Com base em sua compreensão sobre o processo de compilação em C, descreva, em linhas gerais, as etapas que ocorrem durante a compilação de um programa. Além disso, explique quando as bibliotecas são integradas ao executável no processo de compilação.