

## **Exercícios – Aula Prática XIV**

## Exercícios

33) Construa o código fonte de uma função em C que receba um número inteiro positivo, o qual representa a posição de um determinado termo na sequência de Fibonacci, a função deve retornar o valor do termo correspondente à posição recebida. Elabore o código fonte de um programa que se utilize da função implementada.

```
unsigned int fibonacci (unsigned int p)  
{  
    unsigned int a, b;  
    for(a=0,b=1;--p;b+=a,a=b-a);  
    return a;  
}
```

```
#include <stdio.h>
unsigned int fibonacci (unsigned int p)
{
    unsigned int a, b;
    for(a=0,b=1;--p;b+=a,a=b-a);
    return a;
}
int main()
{
    int p;
    do{
        printf("\nEntre com a posicao que se deseja saber o ");
        printf("valor do elemento na sequencia de Fibonacci: ");
        scanf("%d",&p);
    } while (p<1);
    printf("\nO elemento %d encontra-se na posicao %d da%s ",
    fibonacci(p), p, "sequencia de Fibonacci.");
    return 0;
}
```

## Exercícios

34) Codifique uma função na linguagem C que receba dois números inteiros, e retorne o resultado do somatório de todos os números contidos no intervalo aberto delimitado pelos números fornecidos. Em seguida, construa um programa que se utilize de forma eficaz da função codificada.

## Exercícios

35) Implemente uma função, na linguagem C, que receba como parâmetro uma matriz 5x5 de valores reais e retorne a média aritmética dos valores contidos abaixo da diagonal principal da matriz.

```
float funcao (float matriz[5][5])
{
    int i, j, numeroDeParcelas=0;
    float aux;
    for (i=0, aux=0; i<5 ; i++)
        for (j=0; j<5; j++)
            if (i>j)
            {
                aux += matriz[i][j];
                numeroDeParcelas++;
            }
    return (aux/numeroDeParcelas);
}
```

```

#include <stdio.h>
#define numEleEmCadaDim 5
float funcao (float matriz[numEleEmCadaDim][numEleEmCadaDim])
{ /* ... */ }
int main()
{
    float vetorBidimensional[numEleEmCadaDim][numEleEmCadaDim];
    int i;
    for (i=0; i<numEleEmCadaDim*numEleEmCadaDim;)
        vetorBidimensional[i/numEleEmCadaDim][i%numEleEmCadaDim]=++i;
    for (i=0; i<numEleEmCadaDim*numEleEmCadaDim; i++)
    {
        if (!(i%numEleEmCadaDim))
            printf ("\n|");
        printf(" %5.2f",
            vetorBidimensional[i/numEleEmCadaDim][i%numEleEmCadaDim]);
        if (i%numEleEmCadaDim==numEleEmCadaDim-1)
            printf (" |");
    }
    printf ("\n\nA media dos valores contidos abaixo da diagonal principal");
    printf (" da matriz eh %f\n", funcao(vetorBidimensional));
    return 0;
}

```



## Exercícios

36) Construa um programa, na linguagem C, que manipule uma matriz  $7 \times 3$  de reais. O programa deve fornecer ao usuário um menu com as seguintes opções de manipulação:

- Atribuir valores aos elementos da matriz;
- Encontrar o menor valor pertencente à matriz;
- Encontrar o maior valor pertencente à matriz;
- Imprimir a matriz com layout adequado.

**Obs.:** As operações acima devem ser implementadas através de funções e o programa deve inicializar a matriz com valores fornecidos pelo usuário.

## Exercícios

37) Defina um tipo de dado capaz de armazenar as seguintes informações sobre um determinado aluno: nome, data de nascimento, número de matrícula, CPF e coeficiente de rendimento.

Posteriormente, construa um programa que manipule um vetor com 5 registros de alunos, onde cada registro é um elemento do tipo de dado definido. A manipulação do vetor é feita através das seguintes funções: inicializar vetor, imprimir um determinado registro com base no valor do campo CPF e imprimir um determinado registro com base em sua posição no vetor. O programa deve se utilizar de forma satisfatória das funções mencionadas.

## Exercícios

38) Defina um novo tipo de dado, na linguagem C, capaz de armazenar informações sobre um determinado cliente de um banco. As informações, sobre um cliente, necessárias para a agência bancária em questão são: nome do cliente, CPF, RG, tipo da conta (especial ou clássica), número da conta, data de abertura da conta e saldo. **Obs.:** Clientes com conta especial possuem um limite disponível de R\$ 3.000,00.

## Exercícios

39) Construa o código fonte de um programa na linguagem C que manipule uma série de registros de contas de uma determinada agência bancária. As manipulações possíveis são:

- Incluir uma nova conta;
- Excluir uma determinado conta;
- Alterar o conteúdo de um determinado registro;
- Consultar a presença de uma determinada conta;
- Consultar o saldo de uma determinada conta;
- Consultar o limite de uma determinada conta;
- Efetuar, se possível, um saque em uma determinada conta;
- Imprimir as informações sobre todas as contas.

**Observações:** Utilize o tipo de dado definido no exercício anterior; O programa deve armazenar no máximo 100 registro; Cada manipulação deve ser efetuada por uma função distinta; Não podem existir registros com o mesmo número da conta; Todas as entradas devem ser validadas.