



CENTRO UNIVERSITÁRIO INTERNACIONAL UNINTER
ESCOLA SUPERIOR POLITÉCNICA
BACHARELADO EM ENGENHARIA DA COMPUTAÇÃO
DISCIPLINA DE LÓGICA DE PROGRAMAÇÃO E ALGORITMOS

ATIVIDADE PRÁTICA

KELE PÓLVORA CAIRES – RU: 2707381

SANDRO DE ARAUJO

IGUAÍ – BAHIA

2019

1 EXERCÍCIO 1

ENUNCIADO: Faça um algoritmo em linguagem C que recebe como entrada dois valores reais (X e Y) e um símbolo de operação matemática via teclado do usuário (+, -, / ou *). O programa deverá retornar o resultado da operação recebida sobre estes dois números.

Deve ser permitido que o usuário faça cálculos com valores negativos e positivos. Também, tome cuidado com a possibilidade de uma divisão por zero. Caso o usuário tente dividir por zero, seu programa deve tratar este problema de alguma maneira.

Imprima na tela um teste do seu programa utilizando os dois últimos valores de seu RU da UNINTER como os dois valores de entrada, cada valor do seu RU será um dos valores de entrada X e Y. Por exemplo, se seu RU for 1234567, faça X = 6, Y = 7.

Solução do aluno:

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

```
int main(void)
{
    //variáveis

    int x, y;
    char op;

    //Inicio do programa

    printf("Escolha uma operacao +, -, / ou *"); // solicitando uma operação
    scanf_s("%c", &op); // guardando a variável

    //operações

    switch (op) {
    case '+':
        printf("Digite um valor:"); // solicitando a digitação do primeiro valor da operação
        scanf_s("%d", &x); // guardando o valor de x

        printf("Digite um valor:"); // solicitando a digitação do segundo valor da operação
        scanf_s("%d", &y); // guardando o valor de y

        printf("Soma = %d\n", x + y); // mostrando resultado da soma
        break;

    case '-':
        printf("Digite um valor:"); // solicitando a digitação do primeiro valor da operação
        scanf_s("%d", &x); // guardando o valor de x

        printf("Digite um valor:"); // solicitando a digitação do segundo valor da operação
        scanf_s("%d", &y); // guardando o valor de y

        printf("Subtracao = %d\n", x - y); // mostrando resultado da subtração
        break;

    case '*':
        printf("Digite um valor:"); // solicitando a digitação do primeiro valor da operação
        scanf_s("%d", &x); // guardando o valor de x
```

```

printf("Digite um valor:"); // solicitando a digitação do segundo valor da operação
scanf_s("%d", &y); // guardando o valor de y

printf("Multiplicacao = %d\n", x * y); // mostrando resultado da multiplicação
break;

case '/':

    if (y == 0) {
        printf("A operação nao pode ser dividido por 0\n"); // se y for igual a 0, mostre
a mensagem

        system("pause"); // para o programa
    }
    printf("Digite um valor:"); // solicitando a digitação do primeiro valor da operação
    scanf_s("%d", &x); // guardando o valor de x

    printf("Digite um valor:"); // solicitando a digitação do segundo valor da operação
    scanf_s("%d", &y); // guardando o valor de y

    printf("Divisao = %d\n", x / y); // mostrando resultado da divisão
    break;

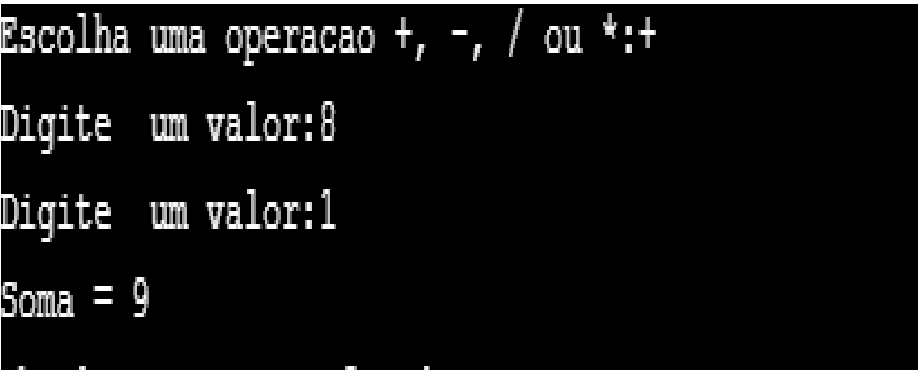
default:
    printf("Invalido!\n"); // se caso nenhuma das opções a cima forem válidas mostra a
mensagem inválida
}
system("pause"); // parou o programa
return 0;

// fim do programa

}

```

Imagem do código funcionando no computador:



```

Escolha uma operacao +, -, / ou *:+
Digite um valor:8
Digite um valor:1
Soma = 9

```

2 EXERCÍCIO 2

ENUNCIADO: Você está organizando um bolão de futebol. Segundo suas regras, os apostadores informam o placar do jogo e ganham 10 pontos se acertarem o vencedor ou se acertarem que foi empate, e ganham mais 5 pontos para o placar de cada time que acertarem. A tabela a seguir dá um exemplo, considerando que o placar real de um jogo foi 3x2.

Faça um algoritmo em linguagem C que leia como dado de entrada o placar real de um jogo e 5 apostas de jogadores para aquele jogo. Para cada aposta, deve-se ler o nome do apostador e o valor que ele apostou. Para cada aposta, calcule os pontos de cada jogador. Ao final do programa, liste na tela o nome de cada apostador e quantos pontos cada um obteve.

Imprima na tela um teste do seu programa utilizando os dois últimos valores de seu RU da UNINTER como o placar real do jogo. Por exemplo, se seu RU for 1234567, faça com que o placar real do jogo seja 6x7.

Solução do aluno:

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

int main(void)
{
    int p1, p2, r; //variáveis

    int pa1, pa2, sert, pnt[5], i;
    char spt[5][10];

    //Inicio do programa

    printf("Placar do jogo:"); // solicitando o placar do jogo
    scanf("%d", &p1); // salvando primeiro placar
    scanf("%d", &p2); //salvando segundo placar

    r = p1 - p2;

    for (i = 0; i < 5; i++) {

        //apostas feitas

        printf("Apostador:\n"); // solicitando o nome do apostador
        scanf("%s", spt[i]); // guardando

        printf("Aposta 01: "); //solicitando a aposta
        scanf("%d", &pa1); // guardando
        printf("Aposta 02: "); // solicitando a aposta
        scanf("%d", &pa2); // guardando

        // colocando os pontos conforme as apostas feitas

        pnt[i] = 0;

        if (pa1 == p1)
            pnt[i] += 5;

        r = pa1 - pa2;

        if (r > 0 && r > 0 ||
            r < 0 && r < 0 ||
            r == 0 && r == 0)
            pnt[i] += 10;
```

```

}
printf("\n\n Pontuacao \n\n:"); //mostrando a pontuação
for (i = 0; i < 5; i++) {
    printf("Apostador: %s\n", spt[i]); // mostrando todos os apostadores
    printf("Pontuacao: %d\n\n", pnt[i]); // mostrando todos os pontos dos apostadores
}

system("pause"); // parando o programa

// fim do programa
}

```

Imagem do código funcionando no computador:

```

Placar do jogo:
8
1
Apostador:
Kely
Aposta 01: 1
Aposta 02: 2
Apostador:
Lucas
Aposta 01: 1
Aposta 02: 0
Apostador:
Carol
Aposta 01: 8
Aposta 02: 1
Apostador:
Vinicius
Aposta 01: 8
Aposta 02: 1
Apostador:
Luciana
Aposta 01: 8
Aposta 02: 1

Pontuacao

:Apostador: Kely
Pontuacao: 10

Apostador: Lucas
Pontuacao: 10

Apostador: Carol
Pontuacao: 15

Apostador: Vinicius
Pontuacao: 15

Apostador: Luciana
Pontuacao: 15

```

3 EXERCÍCIO 3

ENUNCIADO: Fazer um algoritmo em linguagem C no qual o usuário vai digitando sucessivamente valores positivos e salvando estes valores em um vetor de dados. Quando o usuário entrar com um valor negativo, o programa deve parar de pedir valores e calcular a média dos valores já fornecidos (excluindo o valor negativo digitado).

Para calcular a média, crie uma função que recebe como parâmetro o vetor de dados e retorna para a função principal a média. Imprima a média na tela usando a função principal.

Solução do aluno:

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

//função da média
double cmedia(double* vet, int len)
{
    int i;
    double sm = 0;

    //inicio do programa

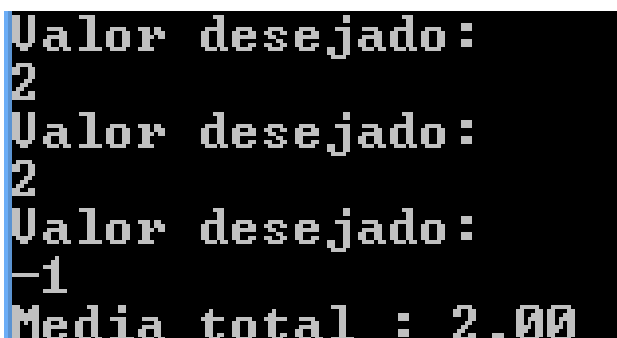
    // definindo os vetores
    for (i = 0; i < len; i++)
    {
        sm = sm + vet[i];
    }
    return sm / len;
}

int main(void)
{
    double vet[100], m;
    int i = 0;
    do
    {
        puts("Valor desejado: "); //solicitando o valor
        scanf_s("%lf", &vet[i]); // guardando o valor
        i++;
    }
    // se o valor for negativo o programa para e mostra a média
    while (vet[i - 1] >= 0);
    m = cmedia(vet, i - 1);
    printf("Media total : %.2lf\n", m);

    system("Pause"); // parando o programa

    //fim do programa
}
```

Imagem do código funcionando no computador:



```
Valor desejado:
2
Valor desejado:
2
Valor desejado:
-1
Media total : 2.00
```

4 EXERCÍCIO 4

ENUNCIADO: Faça um algoritmo em linguagem C com as seguintes funcionalidades:

- Receba uma matriz como dado de entrada. A dimensão da matriz deve ser de 3x5 (3 linhas e 5 colunas);
- Imprimir a matriz na tela, de uma forma legível;
- Encontre a matriz transposta e imprima-a na tela, de forma legível;
- Imprima na tela novamente a matriz transposta, mas agora multiplicando somente a diagonal principal pelos últimos dois números do seu RU da UNINTER;

Solução do aluno:

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

int main(void) {
    //variáveis
    int m1[3][5], m2[5][3];
    int i, j;
    int ru = 81;

    //Inicio do programa

    // entrada

    puts("Valores desejado:");
    for (i = 0; i < 3; i++)
        for (j = 0; j < 5; j++)
            scanf_s("%d", &m1[i][j]);

    // mostrando a matriz na tela
    putchar('\n');
    puts("Digitar os dados :");
    putchar('\n');
    for (i = 0; i < 3; i++) {
        putchar('\n');
        for (j = 0; j < 5; j++)
            printf("%3d ", m1[i][j]);
        putchar('\n');
    }

    // exibindo transposição

    for (i = 0; i < 3; i++)
        for (j = 0; j < 5; j++)
            m2[j][i] = m1[i][j];

    // exibe transposição na tela
    putchar('\n');
    puts("Transposicao");
    putchar('\n');
    for (i = 0; i < 5; i++) {
        putchar('\n');
        for (j = 0; j < 3; j++)
            printf("%3d ", m2[i][j]);
    }
}
```



```

// executando a multiplicação
for (i = 0; i < 3; i++)
    m2[i][i] *= ru;

//executando a multiplicação e mostrando na diagonal
putchar('\n');
puts("Diagonal");
putchar('\n');
for (i = 0; i < 5; i++) {
    putchar('\n');
    for (j = 0; j < 3; j++)
        printf("%d ", m2[i][j]);
}
putchar('\n');
system("pause"); // parando o programa

// fim do programa
}

```

Imagem do código funcionando no computador:

```

Valores desejado:
21
22
23
24
25
26
27
28
29
30
31
32
33
34

Digitar os dados :

 21  22  23  24  25
 26  27  28  29  30
 31  21  32  33  34

Transposicao

 21  26  31
 22  27  21
 23  28  32
 24  29  33
 25  30  34

Diagonal

1701 26 31
22 2187 21
23 28 2592
24 29 33
25 30 34

```