



RESOLUÇÃO

QUESTÕES:

1. Fazer uma função chamada troca que receba dois parâmetros reais X e Y e retorna as variáveis com os conteúdos trocados: se $X = 5$ e $Y = 8$, após a função, $X = 8$ e $Y = 5$.

RESOLUÇÃO EX.01:

```
#include <iostream>
#include <conio.h>
#include <locale.h>
using namespace std;

void troca(double& x, double& y);

int main()
{
    setlocale (LC_ALL, "portuguese");
    char ans;
    double x,y;
    do
    {
        cout << "Forneça o primeiro valor: ";
        cin >> x;
        cout << "Forneça o segundo valor: ";
        cin >> y;
        troca(x,y);
        cout << "Após a troca o primeiro valor será " << x;
        cout << " e o segundo valor " << y << endl;
        cout << "Outra vez? (s/n): ";
        cin >> ans;
    }
    while (ans == 's' || ans == 'S');
    cout << "Fim do programa.\n";
    getch();
}

void troca(double& x, double& y)
{
    double tmp;
    tmp = x;
    x = y;
    y = tmp;
}
```

2. Faça uma função chamada ordena que receba 3 números inteiros a , b e c , e ordena os valores de forma que a passe a ter o menor valor e c o maior valor. Use a função troca definida no exercício 1.

RESOLUÇÃO EX.02:

```
#include <iostream>
#include <conio.h>
#include <locale.h>
using namespace std;

void troca(double& x, double& y);
void ordena(int& a, int& b, int& c);

int main()
{
    setlocale (LC_ALL, "portuguese");
    char ans;
    int a,b,c;
    do
    {
        cout << "Forneça o primeiro valor: ";
        cin >> a;
        cout << "Forneça o segundo valor: ";
        cin >> b;
        cout << "Forneça o terceiro valor: ";
        cin >> c;
        ordena(a,b,c);
        cout << "Após a ordenação o resultado será " << a << ", " << b << " e "
            << c << endl;
        cout << "Outra vez? (s/n): ";
        cin >> ans;
    }
    while (ans == 's' || ans == 'S');
    cout << "Fim do programa.\n";
    getch();
}

void troca(int& x, int& y)
{
    int tmp;
    tmp = x;
    x = y;
    y = tmp;
}

void ordena(int& a, int& b, int& c){
    if (b < a) troca(a,b);
    if (c < a) troca(a,c);
    if (c < b) troca(c,b);
}
```

3. Faça uma função chamada *atribquad()* que receba como parâmetro um numero real representando a medida do lado de um quadrado, devolvendo 3 valores: a área, o perímetro e o tamanho da diagonal (dada por lado \times).

RESOLUÇÃO EX.03:

```
#include <iostream>
#include <conio.h>
#include <locale.h>
#include <cmath>
using namespace std;

double calcHipotenusa(double a, double b);
void atribquad(double l, double& area, double& perimetro, double& diagonal);

int main()
{
    setlocale (LC_ALL,"portuguese");
    char ans;
    double l, area, perimetro, diagonal;
    do
    {
        cout << "Forneça a medida do lado do quadrado: ";
        cin >> l;
        atribquad(l,area, perimetro, diagonal);
        cout << "Medida do lado do quadrado fornecida: " << l << endl;
        cout << "Área: " << area << endl;
        cout << "Perímetro: " << perimetro << endl;
        cout << "Diagonal: " << diagonal << endl;
        cout << "Outra vez? (s/n): ";
        cin >> ans;
    }
    while (ans == 's' || ans == 'S');
    cout << "Fim do programa.\n";
    getch();
}

void atribquad(double l, double& area, double& perimetro, double& diagonal)
{
    area = pow(l,2);
    perimetro = l*4;
    diagonal = calcHipotenusa(l,l);
}

double calcHipotenusa (double a, double b)
{
    double c;
    c = sqrt ( pow (a ,2) + pow (b ,2) );
    return c;
}
```

4. Escreva função chamada *tempoJogo()* que recebe 4 parâmetros inteiros, cada par de inteiros representando dois horários (horas e minutos). A função deve devolver o período em minutos decorrido entre os dois horários, e também o valor 1 (um) se o intervalo de tempo calculado for maior que 3 horas e meia e 0 (zero) caso contrário.

RESOLUÇÃO EX.04:

```
#include <iostream>
#include <conio.h>
#include <locale.h>
#include <cmath>
using namespace std;

bool tempJogo(int hi, int mi, int hf, int mf, int &periodo);

int main()
{
    setlocale (LC_ALL,"portuguese");
    char ans;
    int hi, mi, hf, mf, periodo;
    do
    {
        cout << "Forneça a hora e minuto de início: ";
        cin >> hi >> mi;
        cout << "Forneça a hora e minuto de término: ";
        cin >> hf >> mf;
        if (tempJogo(hi,mi,hf,mf,periodo))
            cout << "Intervalo calculado SUPERIOR que 3:30! Tempo total em minutos = " << periodo << endl;
        else
            cout << "Intervalo calculado INFERIOR a 3:30! Tempo total em minutos = " << periodo << endl;
        cout << "Outra vez? (s/n): ";
        cin >> ans;
    }
    while (ans == 's' || ans == 'S');
    cout << "Fim do programa.\n";
    getch();
}

bool tempJogo(int hi, int mi, int hf, int mf, int &periodo){
    periodo = abs((hi*60+mi)-(hf*60+mf));
    return (periodo<210) ? 0 : 1;
}
```

5. Faça uma função chamada *maxpot()* que receba como parâmetros três inteiros: *X*, *Y* e *MAX*. Esta função deve devolver 2 valores: o resultado de *XY* e o valor 1 (um) se este valor calculado for maior que *MAX*, ou 0 (zero) caso contrário.

RESOLUÇÃO EX.05:

```
#include <iostream>
#include <conio.h>
#include <locale.h>
#include <cmath>
using namespace std;

bool maxpot(int X, int Y, int& MAX);

int main()
{
    setlocale (LC_ALL, "portuguese");
    char ans;
    int X, Y, MAX;
    do
    {
        cout << "Forneça a valor de X: ";
        cin >> X;
        cout << "Forneça a valor de Y: ";
        cin >> Y;
        cout << "Forneça a valor de MAX: ";
        cin >> MAX;
        if (maxpot(X,Y,MAX))
            cout << "Calculo MAIOR que MAX, sendo = " << MAX << endl;
        else
            cout << "Calculo MENOR OU IGUAL a MAX, sendo = " << MAX << endl;
        cout << "Outra vez? (s/n): ";
        cin >> ans;
    }
    while (ans == 's' || ans == 'S');
    cout << "Fim do programa.\n";
    getch();
}

bool maxpot(int X, int Y, int& MAX)
{
    int max_tmp;
    max_tmp = MAX;
    MAX = X*Y;
    return (MAX<=max_tmp) ? 0 : 1;
}
```

6. Governos e empresas do mundo inteiro estão se preocupando cada vez mais com as medições de carbono (emissões anuais de dióxido de carbono na atmosfera) de prédios que queimam vários tipos de combustíveis para gerar calor, veículos que queimam combustíveis para gerar potência e outras questões similares. Muitos cientistas culpam esses gases de efeito estufa pelo fenômeno chamado aquecimento global. Crie três funções que ajudem a calcular a medição de emissão de dióxido de carbono de um prédio, um carro e uma bicicleta, respectivamente. Cada função deverá solicitar dados apropriados do usuário e, depois, calcular e exibir as respectivas medições (*pesquise alguns sites que explicam como calcular as medições de carbono*). As funções deverão utilizar sobrecarga, conforme visto em aula, retornar `void` e utilizar passagem de parâmetro por referência para retornar o valor das respectivas medições. Escreva um programa que induza o usuário a digitar o tipo de medição de emissão de dióxido de carbono a ser calculada, e depois chame a função correspondente. Para cada tipo de medição, apresente alguma informação de identificação e o objeto que deixa aquela emissão.

Informações para funções¹:

- Prédio:
 - Consumo médio mensal de energia elétrica: 1kWh/mês \approx 1.63 Kg CO₂e²;
 - Consumo médio mensal de gás: 1 Botijão/ano \approx 149.04 Kg CO₂e;
- Carro (para cada 1KM/mês rodado):
 - Gasolina:
 - * Motor 1.0(8V): \approx 1.74 Kg CO₂e;
 - * Motor 2.0(16V): \approx 3.05 Kg CO₂e;
 - Flex:
 - * Motor 1.0(8V): \approx 0.87 Kg CO₂e;
 - * Motor 2.0(16V): \approx 1.52 Kg CO₂e;
 - Diesel: \approx 2.82 Kg CO₂e;
- Bicicleta: o uso de bicicletas não gera emissão de CO₂e.

RESOLUÇÃO EX.06:

```
#include <iostream>
#include <conio.h>
#include <stdlib.h>
#include <locale.h>
using namespace std;

void emissao(double& c, double energia, double gas);
void emissao(double& c, int tmotor, double rodado, int comb);
void emissao(double& c);

int main()
{
    setlocale (LC_ALL,"portuguese");
    char ans;
    int opc, tmotor, comb;
    double c, energia, gas, rodado;

    do
    {
        system("cls");
        cout << "-----Calculo de emissao-----\n"
              << " [0] Predio \ [1] Carro \ [2] Bicicleta \n"
              << "-----\n>";
```

¹Valores referências de www.iniciativaverde.org.br

²CO₂e é uma medida internacionalmente aceita que expressa a quantidade de gases de efeito estufa (Fonte: Wikipédia)

```

        cin >> opc;
        system("cls");
        switch(opc)
        {
            case 0:
                cout << "Informe o consumo de energia mensal  
em KWh: ";
                cin >> energia;
                cout << "Informe o consumo de botijões por  
ano: ";
                cin >> gas;
                emissao(c,energia,gas);
                break;
            case 1:
                cout << "Informe a quantidade de quilômetros  
rodados por mês: ";
                cin >> rodado;
                cout << "Informe o tipo de combustível - [0]  
Gasolina / [1] Flex / [2] Diesel: ";
                cin >> comb;
                if(comb!=2)
                {
                    cout << "Informe o tipo do motor -  
[1] para 1.0(8V) / [2] para  
2.0(16V): ";
                    cin >> tmotor;
                }
                emissao(c,tmotor,rodado,comb);
                break;
            case 2:
                emissao(c);
                break;
        }
        cout << "A emissao é de " << c << "Kg CO2e\n";
        cout << "Outra vez? (s/n): ";
        cin >> ans;
    }
    while((ans == 'S')||(ans == 's'));
    cout << "Fim do programa.\n";
    getch();
}

void emissao(double& c, double energia, double gas) {
    c = energia * 1.63 + gas * 149.04/12;
}

void emissao(double& c, int tmotor, double rodado, int comb) {
    switch(comb)
    {
        case 0://gasolina
            if(tmotor==1)
                c = rodado * 1.74;
            else
                c = rodado * 3.05;
            break;
        case 1://flex
            if(tmotor==1)
                c = rodado * 0.87;
            else
                c = rodado * 1.52;
            break;
        case 2://diesel
            c = rodado * 2.82;
            break;
    }
}

void emissao(double& c) {
    c = 0;
}

```

7. Dado o código fonte do programa calculadora e arquivo executável, analise-o e represente-o, esquematicamente, através de um único **FLUXOGRAMA**.

