



AVALIATIVO EM LABORATÓRIO (*Resolução*)

1. A área de um triângulo arbitrário pode ser calculada por meio da fórmula:

$$area = \sqrt{s(s-a)(s-b)(s-c)} \quad (1)$$

onde a , b , e c são as medidas dos lados e s é o semiperímetro, definido a seguir:

$$s = (a + b + c)/2 \quad (2)$$

Escreva uma função *void* que utilize cinco parâmetros: três parâmetros chamados por valor que forneçam as medidas dos lados e dois parâmetros chamados por referência que calculem a área e o perímetro (*não o semiperímetro*).

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

void calcAP(int a, int b, int c, float& area, int& prmt);

int main()
{
    setlocale (LC_ALL, "portuguese");
    int a,b,c, prmt;
    float area;
    char ans;

    do
    {
        cout << "Forneça a medida do lado (a): ";
        cin >> a;
        cout << "Forneça a medida do lado (b): ";
        cin >> b;
        cout << "Forneça a medida do lado (c): ";
        cin >> c;
        calcAP(a,b,c,area,prmt);
        cout << "Área do Triângulo: " << area << endl;
        cout << "Perímetro do Triângulo: " << prmt << endl;
        cout << "Outra vez? (s/n): ";
        cin >> ans;
    }
    while (ans == 's' || ans == 'S');
    cout << "Fim do teste.\n";
    getch();
}

void calcAP(int a, int b, int c, float& area, int& prmt)
{
    float s = (a+b+c)/2;
    area = sqrt((s*(s-a)*(s-b)*(s-c)));
    prmt = a+b+c;
}
```

2. Escreva um programa que diga quantas moedas retornar para qualquer quantia de 1 a 99 reais. Por exemplo, se a quantia é 86 centavos, a saída deve se algo parecido com:

86 reais podem ser fornecidos como
3 de 25 centavo(s), 1 de 10 centavo(s) e 1 de 1 centavo(s)

Utilize denominações para moedas de 25 centavos, 10 centavos e 1 centavo. Não utilize moedas de 50 centavos nem de 5 centavos.

Seu programa utilizará a seguinte função (entre outras):

```
void calculaMoedas(int valorDaMoeda, int& numero, int& quantiaRestante);  
// Pré-condição: 0 < valorDaMoeda < 100; 0 <= quantiaRestante < 100.  
// Pós-condição: número fixado como igual ao número máximo  
// de moedas de denominação valorDaMoeda centavos que possa ser obtido  
// a partir de quantiaRestante centavos. quantiaRestante diminui conforme  
// o valor das moedas, ou seja, diminui o número*valorDaMoeda.
```

Por exemplo, suponha que o valor da variável *quantiaRestante* seja 86. Então, depois da seguinte chamada, o valor de número será 3 e o valor da *quantiaRestante* será 11 (porque se você retira 75 de 86, restam 11):

```
calculaMoedas(25, numero, quantiaRestante);
```

Inclua um loop que permita ao usuário repetir esse cálculo para novos dados de entrada até o usuário dizer que deseja encerrar o programa. (*Dica:* utilize divisão de inteiros e o operador % para implementar essa função).

```
# include <stdio.h>  
# include <conio.h>  
# include <locale.h>  
# include <iostream>  
# include <cmath>  
using namespace std;  
  
void calculaMoedas(int valorDaMoeda, int& numero, int& quantiaRestante);  
  
int main() {  
    setlocale (LC_ALL,"portuguese");  
    int valor, numero, quantiaRestante;  
    char ans;  
  
    do {  
        cout << "Forneça a quantia: ";  
        cin >> valor;  
        quantiaRestante = valor;  
        cout << valor << "centavos podem ser fornecidos como " << endl;  
  
        calculaMoedas(25,numero,quantiaRestante);  
        cout << numero << " de 25 centavo(s), ";  
  
        calculaMoedas(10,numero,quantiaRestante);  
        cout << numero << " de 10 centavo(s), ";  
  
        calculaMoedas(1,numero,quantiaRestante);  
        cout << numero << " de 1 centavo(s). " << endl;  
  
        cout << "Outra vez? (s/n): ";  
        cin >> ans;  
    }  
    while (ans == 's' || ans == 'S');  
    cout << "Fim do teste.\n";  
    getch();  
}  
  
void calculaMoedas(int valorDaMoeda, int& numero, int& quantiaRestante) {  
    numero = quantiaRestante / valorDaMoeda;  
    quantiaRestante = quantiaRestante % valorDaMoeda;  
}
```

3. Escreva um programa que converta da notação de 24 horas para a notação de 12 horas. Por exemplo, o programa deve converter 14:25 em 2:25 P.M. A entrada é dada em dois inteiros. Deve haver pelo menos três funções: uma para a entrada, uma para fazer a conversão e uma para a saída. Registre a informação A.M./P.M. como um valor do tipo `char`, 'A' para A.M. e 'P' para P.M. Assim, a função para efetuar as conversões terá um parâmetro formal chamado por referência de tipo `char` para registrar se é A.M. ou P.M. (a função terá outros parâmetros também). Inclua um loop que permita que o usuário repita esse cálculo para novos valores de entrada todas as vezes que desejar, até o usuário dizer que deseja encerrar o programa.

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

void entrada(int& hora, int& min);
void converte(int& hora, int& min, char& ap);
void saida(int hora, int min, char ap);

int main() {
    setlocale (LC_ALL, "portuguese");
    int horas, minutos;
    char ans, ap;

    do {
        entrada(horas, minutos);
        converte(horas, minutos, ap);
        saida(horas, minutos, ap);
        cout << "Outra vez? (s/n): ";
        cin >> ans;
    }
    while (ans == 's' || ans == 'S');
    cout << "Fim do teste.\n";
    getch();
}

void entrada(int& horas, int& minutos) {
    cout << "Informe as horas e minutos: ";
    cin >> horas >> minutos;
}

void converte(int& hora, int& min, char& ap) {
    if (hora > 12)
    {
        hora = hora % 12;
        ap = 'P';
    }
    else if (hora < 12)
        ap = 'A';
    else
        ap = 'P';
}

void saida(int horas, int minutos, char ap) {
    cout << "Hora: " << horas << ":" << minutos;
    if (ap == 'A')
        cout << "AM" << endl;
    else
        cout << "PM" << endl;
}
```

OBS.: não utilizar variável global e/ou função que retorne valor.