

## Atividade prática preparatória para a prova 1

1. [Questão 30 do Poscomp 2016] Assinale a alternativa que corresponde à saída do programa a seguir:

```
int i, x=4, w=9,q;
for (i=-1; i<20; i+=3){
    x++;
    for (q=4; q<11; q++){
        do{
            i+=3;
            w = sizeof (i);
            i=x+w;
            x= w+i;
        }while (x<15);
    }
}
printf("x: %d, i: %d", x, i);
```

- a) x: 68, i: 67.
  - b) x: 68, i: 68.
  - c) x: 69, i: 68.
  - d) x: 69, i: 69.
  - e) x: 69, i: 70.
2. Considere o programa corretamente codificado em C apresentado a seguir e faça o que se pede.

```
001 int main(void)
002 {     float a, b, c, area, semip;
003     while (1)
004     {     printf("Informe os lados do triangulo: ");
005           scanf("%f %f %f", &a, &b, &c);
006           if (a == 0.0 || b == 0.0 || c == 0.0)
007               break;
008           if (a < b + c && a > abs(b - c))
009               semip = (a + b + c) / 2;
010               area = pow(semip * (semip - a) * (semip - b) * (semip
- c), 0.5);
011               printf("\nA area eh: %f\n\n", area);
012           }
013           else
014               printf("Nao eh um triangulo\n\n");
015     }
016     return 0;
017 }
```

Reescreva o programa adaptando-o para:

- a) usar uma *struct* para armazenar as medidas dos 3 lados do triângulo;
- b) usar uma função para computar a área do triângulo, conforme o cálculo que é realizado nas linhas 009 e 010 do programa;
- c) usar uma função para verificar se as medidas fornecidas pelo usuário podem ou não formar um triângulo. A rotina deverá retornar 0 se as medidas não correspondem a um triângulo, 1 for um triângulo escaleno, 2 se for isósceles e 3 se for equilátero. Considere que para um conjunto de medidas *a*, *b* e *c* poder formar um triângulo, a condição verificada na linha 008 do programa deve ser verdadeira.

### Atividade prática preparatória para a prova 1

3. Considere o programa corretamente codificado em C apresentado a seguir e faça o que se pede nos itens a) e b).

```
001 int main(void)
002 { int n = 0, x1, x2, r, c;
003   while (n >= 0)
004   { printf("Informe o valor:\n"); scanf("%d", &n);
005     if (n >= 0)
006     { if (n == 0) r = 0;
007       else
008       { if (n == 1) r = 1;
009         else
010         { c = 2; x1 = 1; x2 = 0;
011           while (c <= n)
012           { r = x1 + x2; x2 = x1; x1 = r; c++; }
013         }
014         printf("\nO valor eh %d\n", r);
015       }
016     }
017   return 0;
018 }
```

- a) Qual o texto exato que será exibido pelo `printf` da linha 014 quando for informado no `scanf` da linha 004 o último dígito do seu RA que é diferente de zero?
- b) Reescreva o programa para que utilize uma função para executar o processamento que é realizado entre as linhas 006 e 013.
4. Uma importante medida estatística é a variância, usada para avaliar o grau de dispersão de um conjunto de números em relação à sua média. A fórmula da variância é dada por

$$V = \frac{\sum (x - m)^2}{N}$$

onde  $N$  é a quantidade de elementos e  $m$  é a média aritmética simples dos valores, que é dada por

$$m = \frac{\sum x}{N}$$

Escreva um programa que receba inicialmente um valor de  $N$  (um inteiro entre 1 e 100), indicando a quantidade de números a serem considerados. Em seguida o programa deverá receber e armazenar em um vetor  $N$  números reais de precisão simples (tipo `float` da linguagem C), calculando e imprimindo (com 4 casas depois da vírgula) a média e a variância desses valores conforme as fórmulas anteriormente apresentadas, também utilizando números reais de precisão simples. Usar uma função para calcular a média dos valores do vetor e outra função para calcular a variância. Imprimir os resultados apenas na rotina principal (a rotina `main()`).

*Exemplo de entrada:*

```
12
1.0 3.5 0.5 8.9 19.2 88.91 1.23 8.42 7.98 -12.52 128.39 -200.57
```

*Saída esperada para o exemplo de entrada:*

```
Media: 4.5783          Variancia: 5426.6636
```