

## Atividade de Avaliação 1

**(Questão 1):** Implemente a função **desviopadrao()** para calcular o desvio padrão de um conjunto de valores. A função recebe como parâmetros um vetor contendo os valores e o número de valores existentes no vetor. O desvio padrão (std) é definido pela seguinte fórmula, onde  $\bar{x}$  é a média do conjunto de valores.

$$std(x_1, x_2, \dots, x_n) = \sqrt{\frac{1}{N-1} \sum_{i=1}^N (x_i - \bar{x})^2}$$

A função **desviopadrao()** deve obedecer a seguinte assinatura:

```
double desviopadrao(double *v, int n);
```

Além de implementar a função para cálculo do desvio padrão, crie na função principal exemplos de uso da função.

**(Questão 2):** Considere uma aplicação gráfica que define dois tipos estruturados identificados pelos nomes Ponto e Circulo. O tipo estruturado Ponto representa o objeto ponto em duas dimensões, que é composto por duas coordenadas tipo float. O tipo estruturado Circulo representa o objeto círculo composto por um nome (que nunca contém mais do que 10 caracteres, p.ex. Circulo015), por um ponto que define o seu centro e por um raio com a precisão de um float. A Figura abaixo ilustra estas estruturas:

Ponto	Circulo
X	Nome
Y	Centro
	Raio

Escreva um programa completo (com includes, structs, funções e a main), em um único arquivo .c, que implementa e testa as seguintes funções:

- **criaCirculo()** – que cria um novo círculo. Esta função recebe um nome, as coordenadas x e y do centro e o raio para este novo círculo, e retorna um ponteiro para Circulo.
- **contem()** – que testa se um determinado círculo contém outro círculo dado. Esta função recebe dois ponteiros para o tipo estruturado Circulo e retorna 0 se o primeiro círculo não contém o segundo círculo e diferente de zero se contém.

Um círculo c1 contém outro círculo c2 se a distância entre os seus centros (d) somada ao raio de c2 for menor ou igual ao raio de c1, isto é:  $d + r_2 \leq r_1$ . A distância d entre dois pontos com coordenadas (x1, y1) e (x2, y2) é calculada pela seguinte fórmula:

$$d = \sqrt{(x_2 - x_1)^2 + (y_2 - y_1)^2}$$

Veja um exemplo da função main que cria os 3 círculos conforme a figura e testa se o primeiro círculo contém um dos dois outros:

```
int main(void)
{
    Circulo * c1, * c2, * c3;
    c1 = criaCirculo("Circulo01", 3.0, 4.0, 2.0);
    c2 = criaCirculo("Circulo02", 3.0, 3.0, 1.0);
    c3 = criaCirculo("Circulo03", 1.0, 5.0, 1.0);
    printf("%d\n", contem(c1, c2));
    printf("%d\n", contem(c1, c3));
    free(c1);
    free(c2);
    free(c3);
    return 0;
}
```

