



PONTIFÍCIA UNIVERSIDADE CATÓLICA DE MINAS GERAIS
Instituto de Ciências Exatas e Informática
Departamento de Ciência da Computação

Disciplina Algoritmos e Estruturas de Dados I	Curso Ciência da Computação	Turno Manhã	Período 1º
Professor Felipe Cunha (felipe@pucminas.br)			

Lista de Exercícios 02 - Correção

1. Fazer um programa para ler dois números inteiros e calcular e imprimir a diferença desses números. Depois, se o segundo número for diferente de zero, fazer a divisão do primeiro pelo segundo.

```
1 #include <stdio.h>
2
3 int main() {
4     int num1, num2;
5     float divisao;
6
7     // Leitura dos dois números inteiros
8     printf("Digite o primeiro número inteiro: ");
9     scanf("%d", &num1);
10
11     printf("Digite o segundo número inteiro: ");
12     scanf("%d", &num2);
13
14     // Calcula e imprime a diferença
15     int diferenca = num1 - num2;
16     printf("Diferença: %d\n", diferenca);
17
18     // Verifica se o segundo número é diferente de zero
19     if (num2 != 0) {
20         divisao = (float)num1 / num2;
21         printf("Divisão: %.2f\n", divisao);
22     } else {
23         printf("Não é possível dividir por zero.\n");
24     }
25
26     return 0;
27 }
```

2. Ler um número inteiro, verificar e escrever se ele é divisível ou não por 7.

```
1 #include <stdio.h>
2
3 int main() {
4     int numero;
5
6     // Leitura do número
7     printf("Digite um número inteiro: ");
8     scanf("%d", &numero);
9
10    // Verificação de divisibilidade por 7
11    if (numero % 7 == 0) {
12        printf("O número %d é divisível por 7.\n", numero);
13    } else {
14        printf("O número %d não é divisível por 7.\n", numero);
15    }
16
17    return 0;
18 }
```

3. Ler três números reais, verificar se podem ou não ser lados de um triângulo.

```
1 #include <stdio.h>
2
3 int main() {
4     float a, b, c;
5
6     // Leitura dos três lados
7     printf("Digite o primeiro lado: ");
8     scanf("%f", &a);
9
10    printf("Digite o segundo lado: ");
11    scanf("%f", &b);
12
13    printf("Digite o terceiro lado: ");
14    scanf("%f", &c);
15
16    // Verificação da condição de existência de um triângulo
17    if (a + b > c && a + c > b && b + c > a) {
18        printf("Os valores podem formar um triângulo.\n");
19    } else {
20        printf("Os valores NÃO podem formar um triângulo.\n");
21    }
22
23    return 0;
24 }
```

4. Ler os coeficientes de uma equação do primeiro grau ($ax + b = 0$), calcular e escrever a raiz da equação.

```
1 #include <stdio.h>
2
3 int main() {
4     float a, b, x;
5
6     // Leitura dos coeficientes
7     printf("Digite o coeficiente a: ");
8     scanf("%f", &a);
9
10    printf("Digite o coeficiente b: ");
11    scanf("%f", &b);
12
13    // Verificação se a é diferente de zero
14    if (a != 0) {
15        x = -b / a;
16        printf("A raiz da equação é: %.2f\n", x);
17    } else {
18        printf("Coeficiente 'a' não pode ser zero em uma equação do\nprimeiro grau.\n");
19    }
20
21    return 0;
22 }
```

5. Ler os coeficientes e os termos independentes de um sistema de equações lineares 2×2 . Verificar e escrever se ele tem solução única, infinitas soluções ou não tem solução. Usar determinantes.

```
1 #include <stdio.h>
2
3 int main() {
4     float a1, b1, c1, a2, b2, c2;
5     float D, Dx, Dy;
6
7     // Leitura dos coeficientes
8     printf("Digite os coeficientes da primeira equação (a1 b1 c1): ");
9     scanf("%f %f %f", &a1, &b1, &c1);
10
11     printf("Digite os coeficientes da segunda equação (a2 b2 c2): ");
12     scanf("%f %f %f", &a2, &b2, &c2);
13
14     // Cálculo dos determinantes
15     D = a1 * b2 - a2 * b1;
16     Dx = c1 * b2 - c2 * b1;
17     Dy = a1 * c2 - a2 * c1;
18
19     // Verificação das soluções
20     if (D != 0) {
21         float x = Dx / D;
22         float y = Dy / D;
23         printf("Solução única:\n");
24         printf("x = %.2f, y = %.2f\n", x, y);
25     } else {
26         if (Dx == 0 && Dy == 0) {
27             printf("O sistema possui infinitas soluções.\n");
28         } else {
29             printf("O sistema não possui solução.\n");
30         }
31     }
32
33     return 0;
34 }
```

6. Ler os coeficientes de uma equação do 2º grau, determinar e escrever suas raízes reais ou imaginárias.

```
1 #include <stdio.h>
2 #include <math.h>
3
4 int main() {
5     float a, b, c;
6     float delta, x1, x2, parteReal, parteImaginaria;
7
8     // Leitura dos coeficientes
9     printf("Digite os coeficientes a, b e c da equação do 2º grau:\n");
10    scanf("%f %f %f", &a, &b, &c);
11
12    // Verifica se é uma equação do 2º grau
13    if (a == 0) {
14        printf("Não é uma equação do segundo grau.\n");
15        return 0;
16    }
17
18    // Cálculo do discriminante
19    delta = b * b - 4 * a * c;
20
21    // Verificação das raízes
22    if (delta > 0) {
23        x1 = (-b + sqrt(delta)) / (2 * a);
24        x2 = (-b - sqrt(delta)) / (2 * a);
25        printf("Raízes reais e diferentes:\n");
26        printf("x1 = %.2f\n", x1);
27        printf("x2 = %.2f\n", x2);
28    } else if (delta == 0) {
29        x1 = -b / (2 * a);
30        printf("Raízes reais e iguais:\n");
31        printf("x1 = x2 = %.2f\n", x1);
32    } else {
33        parteReal = -b / (2 * a);
34        parteImaginaria = sqrt(-delta) / (2 * a);
35        printf("Raízes complexas (imaginárias):\n");
36        printf("x1 = %.2f + %.2fi\n", parteReal, parteImaginaria);
37        printf("x2 = %.2f - %.2fi\n", parteReal, parteImaginaria);
38    }
39
40    return 0;
41 }
```

7. Ler dois valores reais, determinar e escrever o maior deles, ou os dois se forem iguais.

```
1 #include <stdio.h>
2
3 int main() {
4     float valor1, valor2;
5
6     // Leitura dos dois valores reais
7     printf("Digite o primeiro valor: ");
8     scanf("%f", &valor1);
9
10    printf("Digite o segundo valor: ");
11    scanf("%f", &valor2);
12
13    // Comparação dos valores
14    if (valor1 > valor2) {
15        printf("O maior valor é: %.2f\n", valor1);
16    } else if (valor2 > valor1) {
17        printf("O maior valor é: %.2f\n", valor2);
18    } else {
19        printf("Os dois valores são iguais: %.2f\n", valor1);
20    }
21
22    return 0;
23 }
```

8. Ler as medidas dos lados de um triângulo. Verificar e escrever se ele é EQUILÁTERO, ISÓSCELES OU ESCALENO.

```
1 #include <stdio.h>
2
3 int main() {
4     float lado1, lado2, lado3;
5
6     // Entrada dos lados do triângulo
7     printf("Digite o primeiro lado do triângulo: ");
8     scanf("%f", &lado1);
9
10    printf("Digite o segundo lado do triângulo: ");
11    scanf("%f", &lado2);
12
13    printf("Digite o terceiro lado do triângulo: ");
14    scanf("%f", &lado3);
15
16    // Verifica se os lados formam um triângulo válido
17    if (lado1 + lado2 > lado3 && lado1 + lado3 > lado2 && lado2 + lado3 > lado1) {
18        // Verifica o tipo de triângulo
19        if (lado1 == lado2 && lado2 == lado3) {
20            printf("O triângulo é EQUILÁTERO.\n");
21        } else if (lado1 == lado2 || lado1 == lado3 || lado2 == lado3) {
22            printf("O triângulo é ISÓSCELES.\n");
23        } else {
24            printf("O triângulo é ESCALENO.\n");
25        }
26    } else {
27        printf("Os lados informados NÃO formam um triângulo válido.\n");
28    }
29
30    return 0;
31 }
```


9. Ler as medidas dos lados de um triângulo. Verificar e escrever se ele é ACUTÂNGULO, RETÂNGULO ou OBTUSÂNGULO.

```
1 #include <stdio.h>
2
3 int main() {
4     float a, b, c, maior;
5
6     // Entrada dos lados
7     printf("Digite os três lados do triângulo:\n");
8     scanf("%f %f %f", &a, &b, &c);
9
10    // Verifica se os lados formam um triângulo válido
11    if (a + b > c && a + c > b && b + c > a) {
12        // Identifica o maior lado
13        if (a >= b && a >= c) {
14            maior = a;
15            a = b;
16            b = c;
17        } else if (b >= a && b >= c) {
18            maior = b;
19            b = a;
20            a = c;
21        } else {
22            maior = c;
23        }
24
25        // Classificação quanto aos ângulos
26        if (maior * maior == a * a + b * b) {
27            printf("O triângulo é RETÂNGULO.\n");
28        } else if (maior * maior < a * a + b * b) {
29            printf("O triângulo é ACUTÂNGULO.\n");
30        } else {
31            printf("O triângulo é OBTUSÂNGULO.\n");
32        }
33    } else {
34        printf("Os lados informados NÃO formam um triângulo válido.\n");
35    }
36
37    return 0;
38 }
```

10. Ler uma data (dia, mês e ano). Verificar e escrever se o ano é bissexto.

Obs.: Um ano é bissexto se é divisível por 4 e não é divisível por 100 ou então se é divisível por 400.

```
1 #include <stdio.h>
2
3 int main() {
4     int dia, mes, ano;
5
6     // Entrada da data
7     printf("Digite o dia: ");
8     scanf("%d", &dia);
9
10    printf("Digite o mês: ");
11    scanf("%d", &mes);
12
13    printf("Digite o ano: ");
14    scanf("%d", &ano);
15
16    // Verificação de ano bissexto
17    if ((ano % 4 == 0 && ano % 100 != 0) || (ano % 400 == 0)) {
18        printf("O ano %d é bissexto.\n", ano);
19    } else {
20        printf("O ano %d NÃO é bissexto.\n", ano);
21    }
22
23    return 0;
24 }
```

11. Ler dois valores reais, determinar e escrever o menor e o maior.

```
1 #include <stdio.h>
2
3 int main() {
4     float valor1, valor2;
5
6     // Entrada dos dois valores
7     printf("Digite o primeiro valor: ");
8     scanf("%f", &valor1);
9
10    printf("Digite o segundo valor: ");
11    scanf("%f", &valor2);
12
13    // Verificação do maior e menor
14    if (valor1 > valor2) {
15        printf("Maior: %.2f\n", valor1);
16        printf("Menor: %.2f\n", valor2);
17    } else if (valor2 > valor1) {
18        printf("Maior: %.2f\n", valor2);
19        printf("Menor: %.2f\n", valor1);
20    } else {
21        printf("Os dois valores são iguais: %.2f\n", valor1);
22    }
23
24    return 0;
25 }
```

12. Ler dois caracteres e escrevê-los em ordem alfabética.

```
1 #include <stdio.h>
2
3 int main() {
4     char c1, c2;
5
6     // Entrada dos dois caracteres
7     printf("Digite o primeiro caractere: ");
8     scanf(" %c", &c1); // Espaço antes de %c para ignorar espaços em
9                          // branco
10
11     printf("Digite o segundo caractere: ");
12     scanf(" %c", &c2);
13
14     // Comparação e exibição em ordem alfabética
15     if (c1 < c2) {
16         printf("Ordem alfabética: %c %c\n", c1, c2);
17     } else if (c2 < c1) {
18         printf("Ordem alfabética: %c %c\n", c2, c1);
19     } else {
20         printf("Os dois caracteres são iguais: %c\n", c1);
21     }
22
23     return 0;
24 }
```

13. Ler dois valores reais. Calcular e escrever o quociente do primeiro pelo segundo se este for diferente de zero.

```
1 #include <stdio.h>
2
3 int main() {
4     float valor1, valor2, quociente;
5
6     // Entrada dos valores
7     printf("Digite o primeiro valor: ");
8     scanf("%f", &valor1);
9
10    printf("Digite o segundo valor: ");
11    scanf("%f", &valor2);
12
13    // Verificação do divisor
14    if (valor2 != 0) {
15        quociente = valor1 / valor2;
16        printf("Quociente: %.2f\n", quociente);
17    } else {
18        printf("Erro: divisão por zero não é permitida.\n");
19    }
20
21    return 0;
22 }
```

14. Ler um número real. Determinar e escrever se ele é maior, menor ou igual a zero.

```
1 #include <stdio.h>
2
3 int main() {
4     float numero;
5
6     // Entrada do número
7     printf("Digite um número real: ");
8     scanf("%f", &numero);
9
10    // Verificação do sinal
11    if (numero > 0) {
12        printf("O número é maior que zero.\n");
13    } else if (numero < 0) {
14        printf("O número é menor que zero.\n");
15    } else {
16        printf("O número é igual a zero.\n");
17    }
18
19    return 0;
20 }
```

15. Ler um número inteiro. Se ele for par calcular e escrever o seu quadrado e se for ímpar calcular e escrever seu cubo.

```
1 #include <stdio.h>
2
3 int main() {
4     int numero;
5
6     // Entrada do número
7     printf("Digite um número inteiro: ");
8     scanf("%d", &numero);
9
10    // Verificação de paridade e cálculo
11    if (numero % 2 == 0) {
12        int quadrado = numero * numero;
13        printf("O número é par. Seu quadrado é: %d\n", quadrado);
14    } else {
15        int cubo = numero * numero * numero;
16        printf("O número é ímpar. Seu cubo é: %d\n", cubo);
17    }
18
19    return 0;
20 }
```

16. Ler um número inteiro.

- Se ele for 1 ou 2, ler as medidas dos lados de um retângulo, calcular e escrever a área do círculo circunscrito a este retângulo.
- Se for 3, 4 ou 5, ler a medida do raio de um círculo. Calcular e escrever a área do quadrado inscrito no mesmo.
- Se for outro número inteiro que não estes, ler a medida do lado de um quadrado, calcular e escrever a área da coroa circular formada pelo círculo inscrito e circunscrito a este quadrado.

```
1 #include <stdio.h>
2 #include <math.h>
3
4 int main() {
5     int opcao;
6     float lado1, lado2, raio, lado, area;
7
8     printf("Digite um número inteiro: ");
9     scanf("%d", &opcao);
10
11     if (opcao == 1 || opcao == 2) {
12         // Retângulo ? círculo circunscrito
13         printf("Digite os dois lados do retângulo:\n");
14         scanf("%f %f", &lado1, &lado2);
15
16         // Diâmetro = diagonal do retângulo
17         float diametro = sqrt(lado1 * lado1 + lado2 * lado2);
18         float raio_circ = diametro / 2;
19         area = M_PI * raio_circ * raio_circ;
20
21         printf("Área do círculo circunscrito: %.2f\n", area);
22
23     } else if (opcao == 3 || opcao == 4 || opcao == 5) {
24         // Círculo ? quadrado inscrito
25         printf("Digite o raio do círculo: ");
26         scanf("%f", &raio);
27
28         // Lado do quadrado inscrito = raio * sqrt(2)
29         float lado_quadrado = raio * sqrt(2);
30         area = lado_quadrado * lado_quadrado;
31
32         printf("Área do quadrado inscrito: %.2f\n", area);
33
34     } else {
35         // Quadrado ? coroa circular (círculo circunscrito - círculo
           inscrito)
36         printf("Digite o lado do quadrado: ");
37         scanf("%f", &lado);
38
39         float raio_inscrito = lado / 2;
40         float raio_circunscrito = lado / sqrt(2);
41
42         float area_inscrito = M_PI * raio_inscrito * raio_inscrito;
43         float area_circunscrito = M_PI * raio_circunscrito *
           raio_circunscrito;
44
45         area = area_circunscrito - area_inscrito;
46
47         printf("Área da coroa circular: %.2f\n", area);
48     }
49 }
```



```
50     return 0;  
51 }
```

17. Ler um caractere e a medida do raio de um círculo.

- Se o caractere lido for T, calcular e escrever a área do triângulo inscrito.
- Se for Q, calcular e escrever a área do quadrado inscrito.
- Se for H, calcular e escrever a área do hexágono inscrito.
- Se for outro caractere que não estes, escrever uma mensagem apropriada.

```
1 #include <stdio.h>
2 #include <math.h>
3
4 int main() {
5     char figura;
6     float raio, area;
7
8     // Entrada dos dados
9     printf("Digite um caractere (T, Q ou H): ");
10    scanf(" %c", &figura); // espaço antes de %c para ignorar espaços
    em branco
11
12    printf("Digite a medida do raio do círculo: ");
13    scanf("%f", &raio);
14
15    // Verificação do caractere e cálculo da área
16    if (figura == 'T' || figura == 't') {
17        // Triângulo equilátero inscrito
18        float lado = raio * sqrt(3); // fórmula derivada da geometria
19        area = (sqrt(3) / 4) * lado * lado;
20        printf("Área do triângulo inscrito: %.2f\n", area);
21
22    } else if (figura == 'Q' || figura == 'q') {
23        // Quadrado inscrito
24        float lado = raio * sqrt(2);
25        area = lado * lado;
26        printf("Área do quadrado inscrito: %.2f\n", area);
27
28    } else if (figura == 'H' || figura == 'h') {
29        // Hexágono regular inscrito
30        float lado = raio; // em um hexágono inscrito, o lado = raio
31        area = (3 * sqrt(3) / 2) * lado * lado;
32        printf("Área do hexágono inscrito: %.2f\n", area);
33
34    } else {
35        printf("Caractere inválido. Use T, Q ou H.\n");
36    }
37
38    return 0;
39 }
```