

LISTA DE EXERCÍCIOS

Página 1 de 3

Curso <i>Bacharelado em Ciência da Computação</i>			Unidade <i>Instituto de Ciências Exatas e Tecnológicas</i>
Disciplina <i>AP1 – Algoritmos e Programação 1</i>			
Nome do(a) acadêmico(a)			Assinatura
Nº de matrícula	Turma <i>1º Período</i>	Data <i>07/04/2025</i>	Professor(a) <i>Ana Paula Freitas Vilela Boaventura</i>

MATHEUS AGUIAR DOS SANTOS FERREIRA

Acesse o código fonte em:

<https://github.com/theprogmtheus/AP1/tree/master/LAB5>

1.

```
LAB5 > C exc1lab5.c > main()
1  #include <stdio.h>
2
3  int main()
4  {
5
6      int op;
7
8      printf("Escolha um exercício da lista de 1 a 11: ");
9      scanf("%d", &op);
10
11      switch (op)
12      {
13      case 1:
14          printf("1 - Crie um programa em C, que usando switch case, crie um menu para escolher um dos exercícios da lista. ");
15          break;
16
17      case 2:
18          printf("2 - Escreva um programa em C para simular uma pequena conversa do computador com o usuário.");
19          break;
20
21      case 3:
22          printf("3 - Fazer um programa em C que solicita um número decimal e imprime o correspondente em hexa e octal.");
23          break;
24
25      case 4:
26          printf("4 - Escreva um programa em C que indique qual o número de bytes que ocupam todos os tipos básicos em C.");
27          break;
28
29      case 5:
30          printf("5 - Qual a diferença entre os comandos: ++a e a++? Escreva um trecho de código para exemplificar a diferença.");
31          break;
32
33      case 6:
34          printf("6 - Faça um programa em C que leia um determinado valor do tipo inteiro, usando o operador "?", verifique se o valor digitado é par ou ímpar. ");
35          break;
36
37      case 7:
38          printf("7 - Construa um programa em C que, tendo como dados de entrada dois pontos quaisquer do plano P(x1,y1) e Q(x2, y2), imprima a distância entre eles.");
39          break;
40
41      case 8:
42          printf("8 - Desenvolva um programa em C que simule um menu de operações bancárias.");
43          break;
44
45      case 9:
46          printf("9 - Faça um programa em C que leia um número de 3 algarismos e construa outro número de quatro algarismos, de acordo com a seguinte regra: ");
47          break;
48
```

2.

```
LAB5 > C exc2lab5.c > main()
1  #include <stdio.h>
2
3  int main()
4  {
5      int idade;
6
7      printf("Qual a sua idade?\n");
8      scanf("%d", &idade);
9
10     printf("Você tem %d anos", idade);
11
12     return 0;
13 }
```

3.

```
LAB5 > C exc3lab5.c > main()
1  #include <stdio.h>
2
3  int main()
4  {
5      int num;
6      printf("Digite um número decimal:");
7      scanf("%d", &num);
8      printf("Numero em Hexa: %x\n", num);
9      printf("Numero em Octa: %o\n", num);
10 }
```

4.

```
LAB5 > C exc4lab5.c > main()
1  #include <stdio.h>
2
3  v int main()
4  {
5      int a;
6      double b;
7      float c;
8      char d;
9
10     printf("Tamanhos em bytes dos tipos em C:\n");
11     printf("int: %d bytes.\n", sizeof(a));
12     printf("double: %d bytes.\n", sizeof(b));
13     printf("float: %d bytes.\n", sizeof(c));
14     printf("char: %d bytes.\n", sizeof(d));
15 }
```

5.

```
LAB5 > C exc5lab5.c > main()
1  #include <stdio.h>
2
3  // Aqui vou explicar a diferença entre ++num ou num++
4  int main()
5  {
6      /*
7       * Antes de mais nada precisamos entender o operador ++, ele é usado para incrementar +1.
8       * Entenda que sempre que você chamar ++ para uma variável você vai estar atribuindo +1 nela, ex:
9       * int num = 0;
10      * num++;
11      *
12      * Agora o num vale 1, já que iniciou com o valor 0 e foi incrementado logo depois.
13      *
14      * Certo, agora qual a diferença entre incrementar antes da variável ou incrementar depois da variável?
15      * Entende-se o seguinte:
16      *
17      * num++ = use primeiro, depois incremente.
18      * ++num = incremente primeiro, depois use.
19      *
20      * Vamos fazer alguns exemplos em código:
21      */
22
23      // Primeiro vamos instanciar duas variáveis inteiras, 'a' e 'b'
24      int a = 0;
25      int b = 1;
26      // vamos atribuir o valor de 'b' na variável 'a', de forma que 'a' tenha um novo valor.
27      // Mas vamos incrementar o valor de 'b' logo depois.
28      a = b++;
29      /*
30      * Qual o valor de 'a' e o valor de 'b' agora?
31      * Exatamente! a recebe o valor de 'b' antes que ele seja incrementado, então a=1
32      * Logo depois incrementamos em 'b', então b=2
33      *
34      * Logo: a=1 e b=2
35      */
36      printf("a=%d, b=%d\n", a, b);
37      // Agora vamos incrementar antes..
38      b = ++a;
39      /*
40      * Qual o valor das variáveis agora?
41      * Seguindo o mesmo raciocínio de antes, temos que 'b' recebe ++a (incrementa antes, atribui depois)
42      * se a=1 e incrementamos antes, então 'b'=2
43      * e se incrementamos o valor de 'a' então o valor dele também mudou para 2.
44      *
45      * Logo: a=2 e b=2
46      */
47      printf("a=%d, b=%d\n", a, b);
48  }
```

6.

```
LAB5 > C exc6lab5.c > main()
1  #include <stdio.h>
2
3  int main()
4  {
5      int num;
6      printf("Informe um valor: ");
7      scanf("%d", &num);
8
9      printf("O valor informado é %s!", ((num % 2) == 0) ? "PAR!" : "ÍMPAR!");
10
11
12      // Aqui estamos usando o operador ternário para retornar um valor dinâmico de acordo com a condição proposta.
13
14      // (condição) ? (valor retornado caso verdadeiro) : (valor retornado caso falso)
15
16      // (numero é par) ? (retorna "PAR!") : (retorna "ÍMPAR!")
17  }
```

7.

```
LAB5 > C exc7lab5.c > main()
1  #include <stdio.h>
2  #include <math.h>
3
4  int main()
5  {
6      double Ax, Ay;
7      double Bx, By;
8      double distance;
9
10     printf("Ponto A, x=");
11     scanf("%lf", &Ax);
12     printf("Ponto A, y=");
13     scanf("%lf", &Ay);
14
15     printf("Ponto B, x=");
16     scanf("%lf", &Bx);
17     printf("Ponto B, y=");
18     scanf("%lf", &By);
19
20     distance = sqrt(pow((Bx - Ax), 2) + pow((By - Ay), 2));
21
22     printf("A distancia entre os pontos A(%.1f, %.1f) e B(%.1f, %.1f) é igual a: %.3f", Ax, Ay, Bx, By, distance);
23 }
```

8.

```
LAB5 > C exc8lab5.c > main()
1  #include <stdio.h>
2
3  int main()
4  {
5      int op;
6      double balance = 1000;
7      double value;
8
9      printf("MENU DE OPERAÇÕES\n");
10     printf("-----\n");
11     printf("1 - Consultar saldo\n");
12     printf("2 - Realizar saque\n");
13     printf("3 - Realizar depósito\n");
14     printf("4 - Sair\n");
15     printf("-----\n");
16     printf("Operação: ");
17     scanf("%d", &op);
18
19     switch (op)
20     {
21     case 1:
22         printf("Saldo atual: %.2f\n", balance);
23         break;
24
25     case 2:
26         printf("Informe o valor do saque: ");
27         scanf("%lf", &value);
28         if (balance >= value)
29         {
30             balance -= value;
31             printf("Saque realizado com sucesso: %.2f\n", value);
32             printf("Saldo atual: %.2f\n", balance);
33         }
34         else
35         {
36             printf("Saldo insuficiente\n");
37         }
38         break;
39     case 3:
40         printf("Informe o valor do depósito: ");
41         scanf("%lf", &value);
42         balance += value;
43         printf("Depósito realizado com sucesso: %.2f\n", value);
44         printf("Saldo atual: %.2f\n", balance);
45         break;
46
47     case 4:
48         printf("Saindo do sistema...\n");
```


9.

```
LAB5 > C exc9lab5.c > main()
1  #include <stdio.h>
2  int main()
3  {
4
5      int in, out;
6
7      printf("Informe um número com 3 algarismos: ");
8      scanf("%d", &in);
9
10     if ((in >= 100) && (in < 1000))
11     {
12         out = in * 10;
13
14         // 438
15         int first = in / 100; // 4
16         int second = (in - (first * 100)) / 10; // 3
17         int third = (in - ((first * 100) + (second * 10))) / 8; // 8
18
19         int digit = (first + (second * 3) + (third * 5)) % 7;
20         out += digit;
21
22         printf("Código: %d\n", out);
23     }
24     else
25     {
26         printf("Valor inválido.");
27     }
28 }
```

10.

```
LAB5 > C exc10lab5.c > main()
1  #include <stdio.h>
2
3  int main()
4  {
5      int a, b, out;
6
7      printf("Informe dois números inteiros: ");
8      scanf("%d %d", &a, &b); // recebe dois numeros inteiros
9
10     if (a > b)
11     {
12         if (a % 2 == 0)
13             printf("O primeiro número é PAR!\n");
14
15         if (a % b == 0)
16             printf("O primeiro número é multiplo do segundo!\n");
17
18         printf("%d", (a > b) ? (a - b) : (b - a));
19     }
20     else
21     {
22         out = (a + b);
23         printf("%d", out);
24     }
25
26     return 0;
27 }
```

11.

```
LAB5 > C exc11lab5.c > main()
1  #include <stdio.h>
2  #include <math.h>
3
4  int main()
5  {
6      double a, b, c, d, root1, root2;
7      printf("Informe os valores de A B C: ");
8      scanf("%lf %lf %lf", &a, &b, &c);
9      if (a == 0)
10     {
11         printf("valor é inválido, pois não forma uma equação do segundo grau\n");
12     }
13     else
14     {
15         d = (pow(b, 2) - (4 * a * c));
16         if (d > 0)
17         {
18             root1 = (-b + sqrt(d)) / (2 * a);
19             root2 = (-b - sqrt(d)) / (2 * a);
20
21             printf("X1: %lf\nX2: %lf\n", root1, root2);
22         }
23         else
24         {
25             if (d == 0)
26             {
27                 root1 = (-b + sqrt(d)) / (2 * a);
28                 printf("X: %lf\n", root1);
29             }
30             else
31             {
32                 printf("equação não possui raízes reais\n");
33             }
34         }
35     }
36
37     printf("algoritmo será encerrado.");
38
39     return 0;
40 }
```