

## roteiro-01/ex01-01.c

```
1  #include <stdio.h>
2
3  #define TAMANHO 10
4
5  int negativos (float* vet, int N) {
6      int total = 0;
7      for (int i = 0; i < N; i++) {
8          if (vet[i] < 0) total++;
9      }
10     return total;
11 }
12
13 int main () {
14     float vetor[TAMANHO] = { 10, -9, 5, -1, -2, 4, 3, 0, 6, -7 };
15
16     printf("[");
17     for (int i = 0; i < TAMANHO; i++) {
18         printf("%.2f", vetor[i]);
19         if (i < TAMANHO - 1) printf(", ");
20     }
21     printf("]\n");
22
23     printf("Quantidade de negativos = %d\n", negativos(vetor, TAMANHO));
24 }
25
```



**gabriel-dp@gabriel-dp: ~/Desktop/dev/c/lab2/roteiro-01**



```
gabriel-dp@gabriel-dp:~/Desktop/dev/c/lab2/roteiro-01$ ./ex01-01  
[10.00, -9.00, 5.00, -1.00, -2.00, 4.00, 3.00, 0.00, 6.00, -7.00]  
Quantidade de negativos = 4  
gabriel-dp@gabriel-dp:~/Desktop/dev/c/lab2/roteiro-01$
```

## roteiro-01/ex01-02.c

```
1  #include <stdio.h>
2  #include <stdlib.h>
3  #include <time.h>
4  #define TAMANHO 10
5
6  void preenche(double* vet, int n) {
7      srand(time(NULL));
8      for (int i = 0; i < n; i++) {
9          vet[i] = (rand() % 200) - 100;
10     }
11 }
12
13 double max(double* vet, int n) {
14     double maior = vet[0];
15     for (int i = 1; i < n; i++) {
16         if (vet[i] > maior) maior = vet[i];
17     }
18     return maior;
19 }
20
21 double min(double* vet, int n) {
22     double menor = vet[0];
23     for (int i = 1; i < n; i++) {
24         if (vet[i] < menor) menor = vet[i];
25     }
26     return menor;
27 }
28
29 double media(double* vet, int n) {
30     double total = 0;
31     for (int i = 0; i < n; i++) {
32         total += vet[i];
33     }
34     return (total / n);
35 }
36
37 int main() {
38     double vetor[TAMANHO];
39     preenche(vetor, TAMANHO);
40
41     printf("[");
42     for (int i = 0; i < TAMANHO; i++) {
43         printf("%.0f ", vetor[i]);
44         if (i < TAMANHO - 1) printf(", ");
45     }
46     printf("]\n");
47
48     printf("min = %.0f\n", min(vetor, TAMANHO));
49     printf("max = %.0f\n", max(vetor, TAMANHO));
50     printf("media = %.2f\n", media(vetor, TAMANHO));
51
52     return 0;
53 }
54
```



gabriel-dp@gabriel-dp: ~/Desktop/dev/c/lab2/roteiro-01



```
gabriel-dp@gabriel-dp:~/Desktop/dev/c/lab2/roteiro-01$ ./ex01-02
```

```
[3 , 50 , -66 , -100 , 59 , 83 , -80 , 27 , 90 , 68 ]
```

```
min = -100
```

```
max = 90
```

```
media = 13.40
```

```
gabriel-dp@gabriel-dp:~/Desktop/dev/c/lab2/roteiro-01$
```

## roteiro-01/ex01-03.c

```
1  #include <stdio.h>
2  #include <string.h>
3
4  typedef struct {
5      char nome[20];
6      int matricula;
7      double nota;
8  } Aluno;
9
10 void imprimeAluno(Aluno aluno) {
11     printf("Nome = %s | Matricula = %d | Nota = %.2f\n", aluno.nome,
12         aluno.matricula, aluno.nota);
13 }
14
15 void imprimeMax(Aluno* alunos, int n) {
16     int max = 0;
17     for (int i = 1; i < n; i++) {
18         if (alunos[i].nota > alunos[max].nota) max = i;
19     }
20     printf("Maior - ");
21     imprimeAluno(alunos[max]);
22 }
23
24 void imprimeMin(Aluno* alunos, int n) {
25     int min = 0;
26     for (int i = 1; i < n; i++) {
27         if (alunos[i].nota < alunos[min].nota) min = i;
28     }
29     printf("Menor - ");
30     imprimeAluno(alunos[min]);
31 }
32
33 int main() {
34     Aluno a, b, c;
35     strcpy(a.nome, "Ana");
36     a.matricula = 1;
37     a.nota = 10;
38     strcpy(b.nome, "Davi");
39     b.matricula = 2;
40     b.nota = 9.9;
41     strcpy(c.nome, "Prenassi");
42     c.matricula = 3;
43     c.nota = 9.8;
44
45     Aluno alunos[] = {a, b, c};
46     imprimeMax(alunos, sizeof(alunos) / sizeof(alunos[0]));
47     imprimeMin(alunos, sizeof(alunos) / sizeof(alunos[0]));
48
49     return 0;
50 }
```



**gabriel-dp@gabriel-dp: ~/Desktop/dev/c/lab2/roteiro-01**



```
gabriel-dp@gabriel-dp:~/Desktop/dev/c/lab2/roteiro-01$ ./ex01-03
```

```
Maior - Nome = Ana | Matricula = 1 | Nota = 10.00
```

```
Menor - Nome = Prenassi | Matricula = 3 | Nota = 9.80
```

```
gabriel-dp@gabriel-dp:~/Desktop/dev/c/lab2/roteiro-01$
```

## roteiro-01/ex01-04.c

```
1  #include <math.h>
2  #include <stdio.h>
3
4  int raizes(float a, float b, float c, float* x1, float* x2) {
5      float delta = ((b * b) - (4 * a * c));
6
7      if (delta < 0) return 0;
8
9      if (delta == 0) {
10         *x1 = ((-b) / (2 * a));
11         return 1;
12     }
13
14     *x1 = ((-b + sqrt(delta)) / (2 * a));
15     *x2 = ((-b - sqrt(delta)) / (2 * a));
16     return 2;
17 }
18
19 int main() {
20     float a, b, c, x1, x2;
21     printf("ax^2: ");
22     scanf("%f", &a);
23     printf("bx: ");
24     scanf("%f", &b);
25     printf("c: ");
26     scanf("%f", &c);
27
28     switch (raizes(a, b, c, &x1, &x2)) {
29         case 1:
30             printf("uma raiz real\n");
31             printf("x1 = %f", x1);
32             break;
33         case 2:
34             printf("duas raizes reais\n");
35             printf("x1 = %f | x2 = %f", x1, x2);
36             break;
37         default:
38             printf("nao existe raiz real\n");
39             break;
40     }
41     printf("\n");
42
43     return 0;
44 }
```



**gabriel-dp@gabriel-dp: ~/Desktop/dev/c/lab2/roteiro-01**



**gabriel-dp@gabriel-dp:~/Desktop/dev/c/lab2/roteiro-01\$ ./ex01-04**

ax^2: 2

bx: -5

c: -7

duas raizes reais

x1 = 3.500000 | x2 = -1.000000

**gabriel-dp@gabriel-dp:~/Desktop/dev/c/lab2/roteiro-01\$**



## roteiro-01/ex02-01.c

```
1  #include <stdio.h>
2
3  void printCrescente(int i, int max) {
4      if (i > max) return;
5      printf("(%d)", i);
6      printCrescente(i + 1, max);
7  }
8
9  void printDecrescente(int min, int i) {
10     if (i < min) return;
11     printf("(%d)", i);
12     printDecrescente(min, i - 1);
13 }
14
15 int main () {
16     printCrescente(1, 5);
17     printf("\n");
18     printDecrescente(1, 5);
19     printf("\n");
20
21     return 0;
22 }
```



**gabriel-dp@gabriel-dp: ~/Desktop/dev/c/lab2/roteiro-01**



**gabriel-dp@gabriel-dp:~/Desktop/dev/c/lab2/roteiro-01\$ ./ex02-01**

(1)(2)(3)(4)(5)

(5)(4)(3)(2)(1)

**gabriel-dp@gabriel-dp:~/Desktop/dev/c/lab2/roteiro-01\$**

## roteiro-01/ex02-02.c

```
1  #include <stdio.h>
2  #include <string.h>
3  #include <ctype.h>
4
5  #define MAX_CHAR 255
6
7  void imprimeLetrasDecrescente(char* vetor, int atual) {
8      if (atual > MAX_CHAR) {
9          printf("\n");
10         return;
11     }
12
13     int length = strlen(vetor);
14     for (int i = 0; i < length; i++) {
15         if (vetor[i] == (char)(MAX_CHAR - atual)) printf("%c", (char)(MAX_CHAR -
16         atual));
17     }
18     imprimeLetrasDecrescente(vetor, atual + 1);
19 }
20
21 int main () {
22     imprimeLetrasDecrescente("gabriel", 0);
23
24     return 0;
25 }
```



**gabriel-dp@gabriel-dp: ~/Desktop/dev/c/lab2/roteiro-01**



**gabriel-dp@gabriel-dp:~/Desktop/dev/c/lab2/roteiro-01\$ ./ex02-02**

rligeba

**gabriel-dp@gabriel-dp:~/Desktop/dev/c/lab2/roteiro-01\$**

## roteiro-01/ex02-03.c

```
1  #include <stdio.h>
2
3  int somaIntervaloRecursivo(int atual, int inf, int sup) {
4      if (atual < inf) return 0;
5      return atual + somaIntervaloRecursivo(atual - 1, inf, sup);
6  }
7
8  int main() {
9      int inf, sup, soma = 0;
10
11     printf("limite inferior: ");
12     scanf("%d", &inf);
13     printf("limite superior: ");
14     scanf("%d", &sup);
15     printf("somatorio = %d\n", somaIntervaloRecursivo(sup, inf, sup));
16
17     return 0;
18 }
```



**gabriel-dp@gabriel-dp: ~/Desktop/dev/c/lab2/roteiro-01**



**gabriel-dp@gabriel-dp:~/Desktop/dev/c/lab2/roteiro-01\$ ./ex02-03**

limite inferior: 5

limite superior: 10

somatorio = 45

**gabriel-dp@gabriel-dp:~/Desktop/dev/c/lab2/roteiro-01\$**

## roteiro-01/ex02-04.c

```
1  #include <stdio.h>
2
3  int multiplica(int m, int n) {
4      if (n == 0) return 0;
5      return m + multiplica(m, n - 1);
6  }
7
8  int main () {
9      int n, m;
10     printf("Primeiro numero: ");
11     scanf("%d", &n);
12     printf("Segundo numero: ");
13     scanf("%d", &m);
14
15     printf("Produto = %d\n", multiplica(n, m));
16     return 0;
17 }
```



**gabriel-dp@gabriel-dp: ~/Desktop/dev/c/lab2/roteiro-01**



**gabriel-dp@gabriel-dp:~/Desktop/dev/c/lab2/roteiro-01\$ ./ex02-04**

Primeiro numero: 10

Segundo numero: 20

Produto = 200

**gabriel-dp@gabriel-dp:~/Desktop/dev/c/lab2/roteiro-01\$**



## roteiro-01/ex02-05.c

```
1  #include <stdio.h>
2
3  #define N 10
4
5  void preencheRecursivo(int i, int n, int* vetor) {
6      if (i >= n) return;
7      vetor[i] = i + 1;
8      preencheRecursivo(i + 1, n, vetor);
9  }
10
11 void imprimeVetorCrescente(int i, int n, int* vetor) {
12     if (i >= n) return;
13     printf("%d ", vetor[i]);
14     imprimeVetorCrescente(i + 1, n, vetor);
15 }
16
17 void imprimeVetorDecrescente(int i, int n, int* vetor) {
18     if (i >= n) return;
19     printf("%d ", vetor[n - 1 - i]);
20     imprimeVetorDecrescente(i + 1, n, vetor);
21 }
22
23 int maiorRecursivo(int i, int n, int* vetor) {
24     int next = i < n - 1 ? maiorRecursivo(i + 1, n, vetor) : vetor[i];
25     return (vetor[i] > next) ? vetor[i] : next;
26 }
27
28 int main() {
29     int vetor[N];
30     preencheRecursivo(0, N, vetor);
31     imprimeVetorCrescente(0, N, vetor);
32     printf("\n");
33     imprimeVetorDecrescente(0, N, vetor);
34     printf("\n");
35     printf("maior = %d\n", maiorRecursivo(0, N, vetor));
36
37     return 0;
38 }
39
```



gabriel-dp@gabriel-dp: ~/Desktop/dev/c/lab2/roteiro-01



gabriel-dp@gabriel-dp:~/Desktop/dev/c/lab2/roteiro-01\$ ./ex02-05

1 2 3 4 5 6 7 8 9 10

10 9 8 7 6 5 4 3 2 1

maior = 10

gabriel-dp@gabriel-dp:~/Desktop/dev/c/lab2/roteiro-01\$

## roteiro-01/ex02-06.c

```
1  #include <stdio.h>
2
3  void rec_hanoi (int n, char origem, char dest, char trab) {
4      if (n > 0) {
5          rec_hanoi(n - 1, origem, trab, dest);
6          printf("Mova o disco %d para o pino %c para %c\n", n, origem , dest);
7          rec_hanoi(n - 1, trab, dest, origem);
8      }
9  }
10
11 int main () {
12     int discos;
13     printf("Digite a quantidade de discos:\n");
14     scanf("%d", &discos);
15
16     printf("-----RECURSIVO-----\n");
17     rec_hanoi(discos, 'A', 'C', 'B');
18     printf("-----\n");
19
20     return 0;
21 }
```



gabriel-dp@gabriel-dp: ~/Desktop/dev/c/lab2/roteiro-01



gabriel-dp@gabriel-dp:~/Desktop/dev/c/lab2/roteiro-01\$ ./ex02-06

Digite a quantidade de discos: 4

-----RECURSIVO-----

Mova o disco 1 para o pino A para B

Mova o disco 2 para o pino A para C

Mova o disco 1 para o pino B para C

Mova o disco 3 para o pino A para B

Mova o disco 1 para o pino C para A

Mova o disco 2 para o pino C para B

Mova o disco 1 para o pino A para B

Mova o disco 4 para o pino A para C

Mova o disco 1 para o pino B para C

Mova o disco 2 para o pino B para A

Mova o disco 1 para o pino C para A

Mova o disco 3 para o pino B para C

Mova o disco 1 para o pino A para B

Mova o disco 2 para o pino A para C

Mova o disco 1 para o pino B para C

-----

gabriel-dp@gabriel-dp:~/Desktop/dev/c/lab2/roteiro-01\$