

# UFSJ - Ciências da Computação

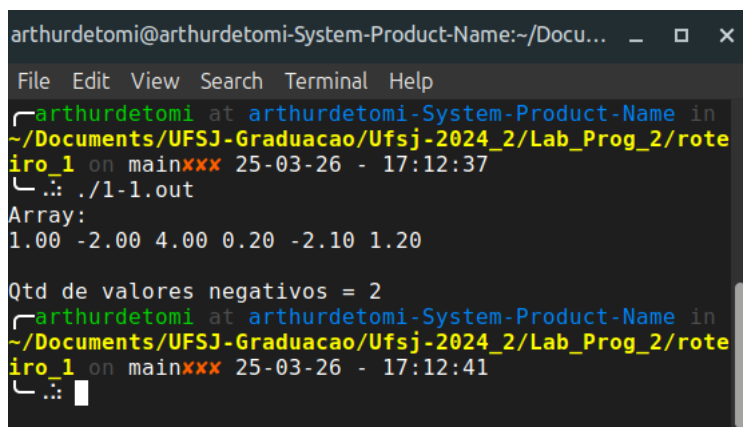
## Laboratório de Programação 2

Nome: Geraldo Arthur Detomi

### Exercício 1.1

```
1 #include <stdio.h>
2
3 #define SIZE 6
4
5 int negativos(float *vet, int n) {
6     int count = 0;
7
8     for (int i = 0; i < n; i++) {
9         if (vet[i] < 0) {
10             count++;
11         }
12     }
13
14     return count;
15 }
16
17 void print_array(float *vet, int n) {
18     printf("Array:\n");
19     for (int i = 0; i < n; i++) {
20         printf("%.2f ", vet[i]);
21     }
22     printf("\n\n");
23 }
24
25 int main() {
26     float vet[] = {1.0f, -2.0f, 4.0f, 0.2f, -2.1f, 1.2f};
27
28     print_array(vet, SIZE);
29
30     printf("Qtd de valores negativos = %d\n",
31           negativos(vet, sizeof(vet) / sizeof(int)));
32
33     return 0;
34 }
```

Saida do terminal:



```
arthurdetomi@arthurdetomi-System-Product-Name:~/Docu... _ □ ×
File Edit View Search Terminal Help
└─ arthurdetomi at arthurdetomi-System-Product-Name in
  ~/Documents/UFSJ-Graduacao/Ufsj-2024_2/Lab_Prog_2/roteiro_1 on mainxxx 25-03-26 - 17:12:37
└─ ./1-1.out
Array:
1.00 -2.00 4.00 0.20 -2.10 1.20

Qtd de valores negativos = 2
└─ arthurdetomi at arthurdetomi-System-Product-Name in
  ~/Documents/UFSJ-Graduacao/Ufsj-2024_2/Lab_Prog_2/roteiro_1 on mainxxx 25-03-26 - 17:12:41
└─
```

### Exercício 1.2

```
1 #include <limits.h>
2 #include <stdio.h>
3 #include <stdlib.h>
4 #include <time.h>
5
6 int generate_random_number(int max) { return rand() % max + 1; }
7
8 void print_array(int *vet, int n) {
9     for (int i = 0; i < n; i++) {
10         printf("%d ", vet[i]);
11     }
12     printf("\n");
13 }
14
15 void fill_array(int *vet, int n) {
```

```

16  srand(time(NULL));
17
18  for (int i = 0; i < n; i++) {
19      vet[i] = generate_random_number(100);
20  }
21 }
22
23 int get_max_value_from_array(int *vet, int n) {
24     int max = INT_MIN;
25
26     for (int i = 0; i < n; i++) {
27         if (vet[i] > max) {
28             max = vet[i];
29         }
30     }
31
32     return max;
33 }
34
35 int get_min_value_from_array(int *vet, int n) {
36     int min = INT_MAX;
37
38     for (int i = 0; i < n; i++) {
39         if (vet[i] < min) {
40             min = vet[i];
41         }
42     }
43
44     return min;
45 }
46
47 double get_median_from_array(int *vet, int n) {
48     int sum = 0;
49
50     for (int i = 0; i < n; i++) {
51         sum += vet[i];
52     }
53
54     return (double)sum / n;
55 }
56
57 int main() {
58     int vet[10];
59
60     int n = sizeof(vet) / sizeof(int);
61
62     printf("Preenchendo array com valores aleatórios...\n");
63     fill_array(vet, n);
64
65     print_array(vet, n);
66     printf("Maior elemento do vetor = %d\n", get_max_value_from_array(vet, n));
67     printf("Menor elemento do vetor = %d\n", get_min_value_from_array(vet, n));
68     printf("Media dos elementos do vetor = %.2lf\n",
69         get_median_from_array(vet, n));
70
71     return 0;
72 }

```

Saída do terminal:

```

arthurdetomi@arthurdetomi-System-Product-Name:~/Documents/UFSJ-Graduacao/Ufsj-2024_2/... _ □ ×
File Edit View Search Terminal Help
[arthurdetomi at arthurdetomi-System-Product-Name in ~/Documents/UFSJ-Graduacao
/Ufsj-2024_2/Lab_Prog_2/roteiro_1 on mainxxx 25-03-26 - 17:17:39
└─.: ./1-2.out
Preenchendo array com valores aleatórios...
22 48 30 38 86 75 14 91 79 60
Maior elemento do vetor = 91
Menor elemento do vetor = 14
Media dos elementos do vetor = 54.30
[arthurdetomi at arthurdetomi-System-Product-Name in ~/Documents/UFSJ-Graduacao
/Ufsj-2024_2/Lab_Prog_2/roteiro_1 on mainxxx 25-03-26 - 17:17:44
└─.: █

```

### Exercício 1.3

```

1  #include <stdio.h>
2  #include <stdlib.h>
3  #include <string.h>
4

```

```
5 typedef struct aluno {
6     int matricula;
7     char nome[50];
8     double nota;
9 } aluno;
10
11 void printf_aluno(aluno a) {
12     printf("Nome: %s\nMatricula:%d\nNota:%.2lf\n\n", a.nome, a.matricula, a.nota);
13 }
14
15 int find_index_min_value_array(double *arr, int n) {
16     int index = 0;
17
18     double min = arr[0];
19
20     for (int i = 0; i < n; i++) {
21         if (arr[i] < min) {
22             min = arr[i];
23             index = i;
24         }
25     }
26
27     return index;
28 }
29
30 int find_index_max_value_array(double *arr, int n) {
31     int index = 0;
32
33     double max = arr[0];
34
35     for (int i = 0; i < n; i++) {
36         if (arr[i] > max) {
37             max = arr[i];
38             index = i;
39         }
40     }
41
42     return index;
43 }
44
45 void print_names_min_and_max_grade(aluno *alunos, int n) {
46     double *notas = (double *)malloc(n * sizeof(double));
47
48     for (int i = 0; i < n; i++) {
49         printf("nota = %lf\n", alunos[i].nota);
50         notas[i] = alunos[i].nota;
51     }
52
53     int min_index_pos = find_index_min_value_array(notas, n);
54     int max_index_pos = find_index_max_value_array(notas, n);
55
56     printf("\nAluno com menor nota:\n");
57     printf_aluno(alunos[min_index_pos]);
58     printf("Aluno com maior nota:\n");
59     printf_aluno(alunos[max_index_pos]);
60
61     free(notas);
62     notas = NULL;
63 }
64
65 int main() {
66     aluno alunos[3];
67     for (int i = 0; i < 3; i++) {
68         aluno a;
69
70         printf("Aluno %d cadastro:\n", i + 1);
71         printf("Matricula :");
72         scanf("%d", &a.matricula);
73         printf("Nome:");
74
75         getchar();
76         fgets(a.nome, 50, stdin);
77         a.nome[strlen(a.nome) - 1] = '\0';
78
79         printf("Nota:");
80         scanf("%lf", &a.nota);
81         printf("\n");
82
83         alunos[i] = a;
84     }
85 }
```

```

86 |     printf("\n");
87 |     print_names_min_and_max_grade(alunos, 3);
88 |
89 |     return 0;
90 | }

```

Saída do terminal:

```

arthurdetomi at arthurdetomi-System-Product-Name in ~/Documents/UFSJ-Graduacao/Ufsj-2024_2/Lab_Prog_2/rot
.: ./1-3.out
Aluno 1 cadastro:
Matricula :212
Nome:João
Nota:5.4

Aluno 2 cadastro:
Matricula :209
Nome:Lucas
Nota:6.7

Aluno 3 cadastro:
Matricula :223
Nome:Maria
Nota:2.5

nota = 5.400000
nota = 6.700000
nota = 2.500000
Aluno com menor nota:
Nome: Maria
Matricula:223
Nota:2.50

Aluno com maior nota:
Nome: Lucas
Matricula:209
Nota:6.70

```

#### Exercício 1.4

```

1 | #include <math.h>
2 | #include <stdio.h>
3 |
4 | double calculate_delta(float a, float b, float c) {
5 |     return pow(b, 2) - 4 * a * c;
6 | }
7 |
8 | // Retorna 1 se existem raizes e 0 caso não exista preenche X1 e X2 caso tenha
9 | // raizes
10 | int raizes(float A, float B, float C, float *X1, float *X2) {
11 |     double delta = calculate_delta(A, B, C);
12 |
13 |     if (delta < 0) {
14 |         return 0;
15 |     }
16 |
17 |     *X1 = (-B + sqrt(delta)) / (2 * A);
18 |     *X2 = (-B - sqrt(delta)) / (2 * A);
19 |
20 |     return (delta == 0.0) ? 1 : 2;
21 | }
22 |
23 | int main() {
24 |
25 |     float A, B, C, X1, X2;
26 |
27 |     printf("Digite os coeficientes Ex(a b c): \n");
28 |     scanf("%f %f %f", &A, &B, &C);
29 |
30 |     int qtd_raizes = raizes(A, B, C, &X1, &X2);
31 |
32 |     if (qtd_raizes == 2) {
33 |         printf("A quantidade de raizes é %d, x1 = %.2f, x2 = %.2f\n", qtd_raizes,
34 |             X1, X2);
35 |     } else if (qtd_raizes == 1) {
36 |         printf("A quantidade de raizes é %d, x1 = x2 = %.2f\n", qtd_raizes, X1);
37 |     } else {
38 |         printf("Não possui raizes reais\n");
39 |     }
40 |
41 |     return 0;

```

42 | }

Saída do terminal:

```

arthurdetomi at arthurdetomi-System-Product-Name in
.: ./1-4.out
Digite os coeficientes Ex(a b c):
1 -3 2
A quantidade de raizes é 2, x1 = 2.00, x2 = 1.00
arthurdetomi at arthurdetomi-System-Product-Name in
.: ./1-4.out
Digite os coeficientes Ex(a b c):
1 2 5
Não possui raizes reais
arthurdetomi at arthurdetomi-System-Product-Name in
.: ./1-4.out
Digite os coeficientes Ex(a b c):
1 0 0
A quantidade de raizes é 1, x1 = x2 = 0.00
arthurdetomi at arthurdetomi-System-Product-Name in

```

## Exercício 2.1

```

1 #include <stdio.h>
2
3 void print_ascending_order(int n) {
4     if (n == 0) {
5         return;
6     }
7     print_ascending_order(n - 1);
8     printf("%d\n", n);
9 }
10
11 void print_descending_order(int n) {
12     if (n == 0) {
13         return;
14     }
15
16     printf("%d\n", n);
17
18     print_descending_order(n - 1);
19 }
20
21 int main() {
22     printf("Ascending order:\n");
23     print_ascending_order(5);
24
25     printf("\n");
26
27     printf("Descending order:\n");
28     print_descending_order(5);
29
30     return 0;
31 }

```

Saída do terminal:

```

arthurdetomi at arthurdetomi-System-Product-Name in
.: ./2-1.out
Ascending order:
1
2
3
4
5

Descending order:
5
4
3
2
1

```

## Exercício 2.2

```

1 #include <stdio.h>
2 #include <string.h>
3
4 void solve(char *word, int n) {
5     if (n == 0) {
6         return;
7     }
8

```

```

9   solve(word + 1, n - 1);
10
11   printf("%c", word[0]);
12 }
13
14 void print_descending_order(char *word, int n) {
15     solve(word, n);
16     printf("\n");
17 }
18
19 int main() {
20     char word[50];
21
22     scanf("%s", word);
23
24     int size = strlen(word);
25
26     print_descending_order(word, size);
27
28     return 0;
29 }

```

Saida do terminal:

```

└─ arthurdetomi at arthurdetomi-System-Product-Name in
└─ .: ./2-2.out
word
drow

```

### Exercício 2.3

```

1  #include <stdio.h>
2
3  int min(int a, int b) { return (a < b) ? a : b; }
4  int max(int a, int b) { return (a > b) ? a : b; }
5
6  int calculate_interval_sum(int a, int b) {
7      int min_value = min(a, b);
8      int max_value = max(a, b);
9
10     if (a == b) {
11         return min_value;
12     }
13
14     return max_value + calculate_interval_sum(min_value, max_value - 1);
15 }
16
17 int main() {
18     int a, b;
19
20     scanf("%d %d", &a, &b);
21
22     printf("Soma = %d\n", calculate_interval_sum(a, b));
23 }

```

Saida do terminal:

```

└─ arthurdetomi at arthurdetomi-System-Product-Name in
└─ .: ./2-3.out
1 4
Soma = 10
└─ arthurdetomi at arthurdetomi-System-Product-Name in
└─ .: ./2-3.out
1 5
Soma = 15

```

### Exercício 2.4

```

1  #include <stdio.h>
2
3  int mult(int m, int n) {
4      if (n == 0) {
5          return 0;
6      }
7
8      return m + mult(m, n - 1);
9  }
10
11 int main() {
12     int a, b;
13
14     scanf("%d %d", &a, &b);
15 }

```

```
16     printf("%d x %d = %d\n", a, b, mult(a, b));
17
18     return 0;
19 }
```

### Exercício 2.5

```
1  #include <stdio.h>
2  #include <stdlib.h>
3  #include <time.h>
4
5  #define SIZE 10
6
7  int generate_random_number(int max) { return rand() % max + 1; }
8
9  void print_array(int *vet, int n) {
10     for (int i = 0; i < n; i++) {
11         printf("%d ", vet[i]);
12     }
13     printf("\n");
14 }
15
16 void fill_array(int *arr, int n) {
17     if (n == 0) {
18         return;
19     }
20
21     arr[0] = generate_random_number(100);
22
23     fill_array(arr + 1, n - 1);
24 }
25
26 void print_array_crescent(int *arr, int n) {
27     if (n == 0) {
28         return;
29     }
30
31     printf("%d\n", arr[0]);
32
33     print_array_crescent(arr + 1, n - 1);
34 }
35
36 void print_array_decreasing(int *arr, int n) {
37     if (n == 0) {
38         return;
39     }
40
41     print_array_decreasing(arr + 1, n - 1);
42
43     printf("%d\n", arr[0]);
44 }
45
46 int find_max(int *arr, int n) {
47     if (n == 1) {
48         return arr[0];
49     }
50
51     int max = find_max(arr + 1, n - 1);
52
53     return (arr[0] > max) ? arr[0] : max;
54 }
55
56 int main() {
57     int arr[SIZE] = {0};
58
59     srand(time(NULL));
60
61     printf("Array antes de ser preenchido...\n");
62     print_array(arr, SIZE);
63
64     fill_array(arr, SIZE);
65
66     printf("Array após ser preenchido...\n");
67     print_array(arr, SIZE);
68
69     printf("\nEm ordem crescente usando recursão...\n");
70     print_array_crescent(arr, SIZE);
71
72     printf("\nEm ordem decrescente usando recursão...\n");
73     print_array_decreasing(arr, SIZE);
74
75     int max_value = find_max(arr, SIZE);
```

```

76 |     printf("O maior valor é: %d\n", max_value);
77 |
78 |     return 0;
79 | }

```

Saida do terminal:

```

[arthurdetomi at arthurdetomi-System-Product-Name in
.: ./2-5.out
Array antes de ser preenchido...
0 0 0 0 0 0 0 0 0
Array após ser preenchido...
62 78 12 7 85 95 73 25 58 86

Em ordem crescente usando recursão...
62
78
12
7
85
95
73
25
58
86

Em ordem decrescente usando recursão...
86
58
25
73
95
85
7
12
78
62
O maior valor é: 95

```

#### Exercício 2.6

```

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 do 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 |     printf("-----RECURSIVO-----\n");
16 |     rec_hanoi(discos, 'A', 'C', 'B');
17 |     printf("-----\n");
18 |     return 0;
19 | }

```

Saida do terminal:



```
arthurdetomi at arthurdetomi-System-Product-Name in ~/Documents/UFSJ-Graduacao/Ufsj-2024_2/Lab_Prog_2/roteiro_1 on main*** 25-03-26 - 17:28:23
$ ./2-6.out
Digite a quantidade de discos:
2
-----RECURSIVO-----
Mova o disco 1 do pino A para B
Mova o disco 2 do pino A para C
Mova o disco 1 do pino B para C
-----
arthurdetomi at arthurdetomi-System-Product-Name in ~/Documents/UFSJ-Graduacao/Ufsj-2024_2/Lab_Prog_2/roteiro_1 on main*** 25-03-26 - 17:28:24
$ ./2-6.out
Digite a quantidade de discos:
3
-----RECURSIVO-----
Mova o disco 1 do pino A para C
Mova o disco 2 do pino A para B
Mova o disco 1 do pino C para B
Mova o disco 3 do pino A para C
Mova o disco 1 do pino B para A
Mova o disco 2 do pino B para C
Mova o disco 1 do pino A para C
-----
arthurdetomi at arthurdetomi-System-Product-Name in ~/Documents/UFSJ-Graduacao/Ufsj-2024_2/Lab_Prog_2/roteiro_1 on main*** 25-03-26 - 17:28:26
$ ./2-6.out
Digite a quantidade de discos:
4
-----RECURSIVO-----
Mova o disco 1 do pino A para B
Mova o disco 2 do pino A para C
Mova o disco 1 do pino B para C
Mova o disco 1 do pino A para B
Mova o disco 1 do pino C para A
Mova o disco 2 do pino C para B
Mova o disco 1 do pino A para B
Mova o disco 4 do pino A para C
Mova o disco 1 do pino B para C
Mova o disco 2 do pino B para C
Mova o disco 1 do pino C para A
Mova o disco 3 do pino B para A
Mova o disco 1 do pino A para B
Mova o disco 2 do pino A para C
Mova o disco 1 do pino B para C
-----
arthurdetomi at arthurdetomi-System-Product-Name in ~/Documents/UFSJ-Graduacao/Ufsj-2024_2/Lab_Prog_2/roteiro_1 on main*** 25-03-26 - 17:28:27
$ ./2-6.out
Digite a quantidade de discos:
5
-----RECURSIVO-----
Mova o disco 1 do pino A para C
Mova o disco 2 do pino A para B
Mova o disco 1 do pino C para B
Mova o disco 3 do pino A para C
Mova o disco 1 do pino B para A
Mova o disco 2 do pino B para C
Mova o disco 1 do pino A para C
Mova o disco 4 do pino A para B
Mova o disco 1 do pino C para B
Mova o disco 2 do pino C para A
Mova o disco 1 do pino B para A
Mova o disco 3 do pino C para B
Mova o disco 1 do pino A para C
Mova o disco 2 do pino A para B
Mova o disco 1 do pino C para B
Mova o disco 5 do pino A para C
Mova o disco 1 do pino B para A
Mova o disco 2 do pino B para C
Mova o disco 1 do pino A para C
Mova o disco 3 do pino B para A
Mova o disco 1 do pino C para B
Mova o disco 2 do pino C para A
Mova o disco 1 do pino B para A
Mova o disco 4 do pino B para C
Mova o disco 1 do pino A para C
Mova o disco 2 do pino A para B
Mova o disco 1 do pino C para B
Mova o disco 3 do pino A para C
Mova o disco 1 do pino B para A
Mova o disco 2 do pino B para C
Mova o disco 1 do pino A para C
-----
arthurdetomi at arthurdetomi-System-Product-Name in ~/Documents/UFSJ-Graduacao/Ufsj-2024_2/Lab_Prog_2/roteiro_1 on main*** 25-03-26 - 17:28:53
```