

Lista 10 de Laboratório de Programação II

Nome : Messias Feres Curi Melo

Matrícula : 2022003764

Problema 1 - Algoritmos de Ordenação

1.1

main.c

```
C main.c > main()
1  #include <stdio.h>
2  #include <stdlib.h>
3
4  void imprimeVetor(int* v, int n){
5      int i, prim = 1;
6      printf("[");
7      for(i=0; i<n; i++){
8          if(prim){ printf("%d", v[i]); prim = 0; }
9          else printf(", %d", v[i]);
10         printf("]\n");
11     }
12
13     void troca(int* a, int *b){
14         int aux = *a;
15         *a = *b;
16         *b = aux;
17     }
18
19     void InsertionSort(int *v, int n){
20         int i, j, atual;
21         for(i=1; i < n; i++){
22             atual = v[i];
23             for(j=i; (j>0) && (atual < v[j-1]); j--){
24                 v[j] = v[j-1];
25             }
26             v[j] = atual;
27         }
28     }
29
30     void SelectionSort(int *v, int n){
31         int i, j, menor;
32         for(i=0; i < n-1; i++){
33             menor = i;
34             for(j=i+1; j < n; j++){
35                 if (v[j] < v[menor])
36                     menor = j;
37             }
38             if(i != menor){
39                 troca(&v[i], &v[menor]);
40             }
41         }
42     }
43
44     void BubbleSort(int *v, int n){
45         int i, j;
46         for(i=0; i<n-1; i++){
47             for(j=0; j<n-i-1; j++){
48                 if (v[j]>v[j+1]) {
```

```

49         troca(&v[j], &v[j+1]);
50     }
51 }
52 }
53
54 int main(){
55     int *v;
56     int n;
57     printf("Digite o tamanho do vetor: ");
58     scanf("%d", &n);
59     v = (int*) malloc (n*sizeof(int));
60
61     printf("\n-> Digite os valores do vetor(Insertion)\n");
62     for(int i = 0; i<n; i++){
63         printf("- Digite o valor [%d]: ", i+1);
64         scanf("%d", &v[i]);
65     }
66     printf("\n-> Ordenando e imprimindo como InsertionSort\n");
67     InsertionSort(v, n);
68     imprimeVetor(v, n);
69
70     printf("\n-> Digite os valores do vetor(Selection)\n");
71     for(int i = 0; i<n; i++){
72         printf("- Digite o valor [%d]: ", i+1);
73         scanf("%d", &v[i]);
74     }
75     printf("\n-> Ordenando e imprimindo como SelectionSort\n");
76     SelectionSort(v, n);
77     imprimeVetor(v, n);
78
79     printf("\n-> Digite os valores do vetor(Bubble)\n");
80     for(int i = 0; i<n; i++){
81         printf("- Digite o valor [%d]: ", i+1);
82         scanf("%d", &v[i]);
83     }
84     printf("\n-> Ordenando e imprimindo como BubbleSort\n");
85     BubbleSort(v, n);
86     imprimeVetor(v, n);
87
88     free(v);
89     return 0;
90 }

```

Console

```
root@DESKTOP-CJCQ0HI:/mnt/c/Users/MessiasFCM/Desktop/roteiro10# gcc main.c -o tp1
root@DESKTOP-CJCQ0HI:/mnt/c/Users/MessiasFCM/Desktop/roteiro10# ./tp1
Digite o tamanho do vetor: 5

-> Digite os valores do vetor(Insertion)
- Digite o valor [1]: 10
- Digite o valor [2]: 34
- Digite o valor [3]: 1
- Digite o valor [4]: 5
- Digite o valor [5]: 12

-> Ordenando e imprimindo como InsertionSort
[1, 5, 10, 12, 34]

-> Digite os valores do vetor(Selection)
- Digite o valor [1]: 342
- Digite o valor [2]: 3
- Digite o valor [3]: 54
- Digite o valor [4]: 2
- Digite o valor [5]: 64

-> Ordenando e imprimindo como SelectionSort
[2, 3, 54, 64, 342]

-> Digite os valores do vetor(Bubble)
- Digite o valor [1]: 23
- Digite o valor [2]: 1
- Digite o valor [3]: 98
- Digite o valor [4]: 34
- Digite o valor [5]: 5

-> Ordenando e imprimindo como BubbleSort
[1, 5, 23, 34, 98]
root@DESKTOP-CJCQ0HI:/mnt/c/Users/MessiasFCM/Desktop/roteiro10#
```

1.2

main.c

```
atv2 > C main.c > BubbleSort(int *, int)
1  #include <stdio.h>
2  #include <stdlib.h>
3
4  void imprimeVetor(int* v, int n){
5      int i, prim = 1;
6      printf("[");
7      for(i=0; i<n; i++){
8          if(prim){ printf("%d", v[i]); prim = 0; }
9          else printf(", %d", v[i]);
10         printf("]\n");
11     }
12
13     void troca(int* a, int *b){
14         int aux = *a;
15         *a = *b;
16         *b = aux;
17     }
18
19     void InsertionSort(int *v, int n){
20         int i, j, atual;
21         for(i=1; i < n; i++){
22             atual = v[i];
23             for(j=i; (j>0) && (atual > v[j-1]); j--){
24                 v[j] = v[j-1];
25             }
26             v[j] = atual;
27         }
28     }
29
30     void SelectionSort(int *v, int n){
31         int i, j, menor;
32         for(i=0; i < n-1; i++){
33             menor = i;
34             for(j=i+1; j < n; j++){
35                 if (v[j] > v[menor])
36                     menor = j;
37             }
38             if(i != menor){
39                 troca(&v[i], &v[menor]);
40             }
41         }
42     }
43
44     void BubbleSort(int *v, int n){
45         int i, j;
46         for(i=0; i<n-1; i++){
47             for(j=0; j<n-i-1; j++){
```

```

48         if (v[j]<v[j+1]) {
49             troca(&v[j], &v[j+1]);
50         }
51     }
52 }
53
54 int main(){
55     int *v;
56     int n;
57     printf("Digite o tamanho do vetor: ");
58     scanf("%d", &n);
59     v = (int*) malloc (n*sizeof(int));
60
61     printf("\n-> Digite os valores do vetor(Insertion)\n");
62     for(int i = 0; i<n; i++){
63         printf("- Digite o valor [%d]: ", i+1);
64         scanf("%d", &v[i]);
65     }
66     printf("\n-> Ordenando e imprimindo como InsertionSort\n");
67     InsertionSort(v, n);
68     imprimeVetor(v, n);
69
70     printf("\n-> Digite os valores do vetor(Selection)\n");
71     for(int i = 0; i<n; i++){
72         printf("- Digite o valor [%d]: ", i+1);
73         scanf("%d", &v[i]);
74     }
75     printf("\n-> Ordenando e imprimindo como SelectionSort\n");
76     SelectionSort(v, n);
77     imprimeVetor(v, n);
78
79     printf("\n-> Digite os valores do vetor(Bubble)\n");
80     for(int i = 0; i<n; i++){
81         printf("- Digite o valor [%d]: ", i+1);
82         scanf("%d", &v[i]);
83     }
84     printf("\n-> Ordenando e imprimindo como BubbleSort\n");
85     BubbleSort(v, n);
86     imprimeVetor(v, n);
87
88     free(v);
89     return 0;
90 }

```

Console

```
7root@DESKTOP-CJCQ0HI:/mnt/c/Users/MessiasFCM/Desktop/roteiro10# gcc main.c -o tp2
root@DESKTOP-CJCQ0HI:/mnt/c/Users/MessiasFCM/Desktop/roteiro10# ./tp2
Digite o tamanho do vetor: 5

-> Digite os valores do vetor(Insertion)
- Digite o valor [1]: 2
- Digite o valor [2]: 4
- Digite o valor [3]: 1
- Digite o valor [4]: 7
- Digite o valor [5]: 3

-> Ordenando e imprimindo como InsertionSort
[1, 2, 3, 4, 7]

-> Digite os valores do vetor(Selection)
- Digite o valor [1]: 32
- Digite o valor [2]: 643
- Digite o valor [3]: 6
- Digite o valor [4]: 34
- Digite o valor [5]: 3

-> Ordenando e imprimindo como SelectionSort
[3, 6, 32, 34, 643]

-> Digite os valores do vetor(Bubble)
- Digite o valor [1]: 23
- Digite o valor [2]: 52
- Digite o valor [3]: 12
- Digite o valor [4]: 87
- Digite o valor [5]: 45

-> Ordenando e imprimindo como BubbleSort
[12, 23, 45, 52, 87]
root@DESKTOP-CJCQ0HI:/mnt/c/Users/MessiasFCM/Desktop/roteiro10#
```

1.3

main.c

```
atv3 > C main.c > main()
1  #include <stdio.h>
2  #include <stdlib.h>
3  #include <time.h>
4  #include <string.h>
5
6  int comp, mov;
7
8  int* copiaVetor(int* v, int n){
9      int i;
10     int *v2;
11     v2 = (int*) malloc (n*sizeof(int));
12     for(i=0; i<n; i++) v2[i] = v[i];
13     return v2;
14 }
15 void imprimeVetor(int* v, int n){
16     int i, prim = 1;
17     printf("[");
18     for(i=0; i<n; i++)
19         if(prim){ printf("%d", v[i]); prim = 0; }
20         else printf(", %d", v[i]);
21     printf("]\n");
22 }
23
24 void preencheAleatorio(int* v, int n, int ini, int fim){
25     int i;
26     for(i=0; i<n; i++)
27         v[i] = ini + rand() % (fim-ini + 1);
28 }
29
30 void troca(int* a, int *b){
31     int aux = *a;
32     *a = *b;
33     *b = aux;
34 }
35
36 void InsertionSort(int *v, int n){
37     int i, j, atual;
38     for(i=1; i < n; i++){
39         atual = v[i];
40         comp++;
41         for(j=i; (j>0) && (atual < v[j-1]); j--){
42             v[j] = v[j-1];
43             comp++;
44             mov++;
45         }
46         v[j] = atual;
47     }
48 }
```

```

49
50 void SelectionSort(int *v, int n){
51     int i, j, menor;
52     for(i=0; i < n-1; i++){
53         menor = i;
54         for(j=i+1; j < n; j++){
55             comp++;
56             if (v[j] < v[menor])
57                 menor = j;
58         }
59         if(i != menor){
60             troca(&v[i], &v[menor]);
61             mov++;
62         }
63     }
64 }
65
66 void BubbleSort(int *v, int n){
67     int i, j;
68     for(i=0; i<n-1; i++)
69         for(j=0; j<n-i-1; j++){
70             comp++;
71             if (v[j]>v[j+1]) {
72                 troca(&v[j], &v[j+1]);
73                 mov++;
74             }
75         }
76 }
77
78 int main(){
79     clock_t t;
80
81     FILE *arquivo;
82     int quantidade, escolha, i;
83
84
85     arquivo = fopen("../valores/100-contrario.txt", "r");
86
87     if(arquivo == NULL){
88         printf("-> Erro ao abrir o arquivo.\n");
89         return 1;
90     }
91
92     fscanf(arquivo, "%d", &quantidade);
93     int *vetor = (int *)malloc(quantidade * sizeof(int));
94
95     if(vetor == NULL){
96         printf("-> Erro ao alocar memória.\n");

```



```

97     return 1;
98 }
99
100 for(i = 0; i < quantidade; i++){
101     fscanf(arquivo, "%d", &vetor[i]);
102 }
103
104 printf("-> Leitura do arquivo com sucesso!\n\n");
105
106 do{
107     printf("-> Menu de Ordenação x=-\n");
108     printf("1 - InsertionSort\n");
109     printf("2 - SelectionSort\n");
110     printf("3 - BubbleSort\n");
111     printf("4 - Sair\n\n");
112     printf("-> Qual a sua escolha: ");
113     scanf("%d", &escolha);
114
115     comp = 0; mov = 0;
116     if(escolha == 1){
117         t = clock();
118         InsertionSort(vetor, quantidade);
119         t = clock() - t;
120     }else if(escolha == 2){
121         t = clock();
122         SelectionSort(vetor, quantidade);
123         t = clock() - t;
124     }else if(escolha == 3){
125         t = clock();
126         BubbleSort(vetor, quantidade);
127         t = clock() - t;
128     }else{
129         printf("-> Escolha Inválida!");
130         break;
131     }
132
133     printf("-> Informacoes:\n");
134     printf("- Tempo Execucao:  %f seconds.\n", ((float)t)/CLOCKS_PER_SEC);
135     printf("- Comparacoes:  %d\n", comp);
136     printf("- Movimentacoes:  %d\n", mov);
137     printf("- Memoria (bytes):  %ld\n", quantidade*sizeof(int));
138     printf("- Valores do vetor: ");
139     imprimeVetor(vetor, quantidade);
140 }while(escolha!=4);
141
142 fclose(arquivo);
143 free(vetor);
144
145 return 0;
}

```

Console 100-contrario

```
messiasfcm@MessiasFOM: /mnt/c/Users/Messi/OneDrive/Área de Trabalho/roteiro10/roteiro10/atv3$ gcc main.c -o tp3
messiasfcm@MessiasFOM: /mnt/c/Users/Messi/OneDrive/Área de Trabalho/roteiro10/roteiro10/atv3$ ./tp3
-> Leitura do arquivo com sucesso!

--x Menu de Ordenação x--
1 - InsertionSort
2 - SelectionSort
3 - BubbleSort
4 - Sair

-> Qual a sua escolha: 1
-> Informacoes:
- Tempo Execuciao: 0.000010 seconds.
- Comparacoes: 5049
- Movimentacoes: 4950
- Memoria (bytes): 400
- Valores do vetor: [2, 4, 6, 8, 10, 12, 14, 16, 18, 20, 22, 24, 26, 28, 30, 32, 34, 36, 38, 40, 42, 44, 46, 48, 50, 52, 54, 56, 58, 60, 62, 64, 66, 68, 70, 72, 74, 76, 78, 80, 82, 84, 86, 88, 90, 92, 94, 96, 98, 100, 102, 104, 106, 108, 110, 112, 114, 116, 118, 120, 122, 124, 126, 128, 130, 132, 134, 136, 138, 140, 142, 144, 146, 148, 150, 152, 154, 156, 158, 160, 162, 164, 166, 168, 170, 172, 174, 176, 178, 180, 182, 184, 186, 188, 190, 192, 194, 196, 198, 200]
--x Menu de Ordenação x--
1 - InsertionSort
2 - SelectionSort
3 - BubbleSort
4 - Sair

-> Qual a sua escolha: 2
-> Informacoes:
- Tempo Execuciao: 0.000008 seconds.
- Comparacoes: 4950
- Movimentacoes: 0
- Memoria (bytes): 400
- Valores do vetor: [2, 4, 6, 8, 10, 12, 14, 16, 18, 20, 22, 24, 26, 28, 30, 32, 34, 36, 38, 40, 42, 44, 46, 48, 50, 52, 54, 56, 58, 60, 62, 64, 66, 68, 70, 72, 74, 76, 78, 80, 82, 84, 86, 88, 90, 92, 94, 96, 98, 100, 102, 104, 106, 108, 110, 112, 114, 116, 118, 120, 122, 124, 126, 128, 130, 132, 134, 136, 138, 140, 142, 144, 146, 148, 150, 152, 154, 156, 158, 160, 162, 164, 166, 168, 170, 172, 174, 176, 178, 180, 182, 184, 186, 188, 190, 192, 194, 196, 198, 200]
--x Menu de Ordenação x--
1 - InsertionSort
2 - SelectionSort
3 - BubbleSort
4 - Sair

-> Qual a sua escolha: 3
-> Informacoes:
- Tempo Execuciao: 0.000009 seconds.
- Comparacoes: 4950
- Movimentacoes: 0
- Memoria (bytes): 400
- Valores do vetor: [2, 4, 6, 8, 10, 12, 14, 16, 18, 20, 22, 24, 26, 28, 30, 32, 34, 36, 38, 40, 42, 44, 46, 48, 50, 52, 54, 56, 58, 60, 62, 64, 66, 68, 70, 72, 74, 76, 78, 80, 82, 84, 86, 88, 90, 92, 94, 96, 98, 100, 102, 104, 106, 108, 110, 112, 114, 116, 118, 120, 122, 124, 126, 128, 130, 132, 134, 136, 138, 140, 142, 144, 146, 148, 150, 152, 154, 156, 158, 160, 162, 164, 166, 168, 170, 172, 174, 176, 178, 180, 182, 184, 186, 188, 190, 192, 194, 196, 198, 200]
--x Menu de Ordenação x--
1 - InsertionSort
2 - SelectionSort
3 - BubbleSort
4 - Sair
```

100-misturado

```
messiasfcm@MessiasFOM: /mnt/c/Users/Messi/OneDrive/Área de Trabalho/roteiro10/roteiro10/atv3$ gcc main.c -o tp3
messiasfcm@MessiasFOM: /mnt/c/Users/Messi/OneDrive/Área de Trabalho/roteiro10/roteiro10/atv3$ ./tp3
-> Leitura do arquivo com sucesso!

--x Menu de Ordenação x--
1 - InsertionSort
2 - SelectionSort
3 - BubbleSort
4 - Sair

-> Qual a sua escolha: 1
-> Informacoes:
- Tempo Execuciao: 0.000006 seconds.
- Comparacoes: 2525
- Movimentacoes: 2426
- Memoria (bytes): 400
- Valores do vetor: [2, 3, 4, 4, 9, 10, 12, 12, 14, 17, 18, 21, 25, 25, 27, 28, 29, 29, 33, 34, 38, 38, 39, 40, 42, 43, 43, 51, 52, 58, 62, 66, 67, 67, 68, 68, 70, 79, 82, 83, 84, 84, 85, 88, 89, 94, 95, 96, 97, 100, 105, 106, 113, 113, 115, 115, 116, 117, 121, 123, 124, 127, 129, 129, 129, 134, 135, 139, 142, 142, 144, 145, 150, 155, 156, 157, 158, 159, 163, 164, 165, 167, 167, 167, 172, 173, 175, 178, 180, 184, 185, 189, 192, 193, 193, 195, 197, 200, 200]
--x Menu de Ordenação x--
1 - InsertionSort
2 - SelectionSort
3 - BubbleSort
4 - Sair

-> Qual a sua escolha: 2
-> Informacoes:
- Tempo Execuciao: 0.000008 seconds.
- Comparacoes: 4950
- Movimentacoes: 0
- Memoria (bytes): 400
- Valores do vetor: [2, 3, 4, 4, 9, 10, 12, 12, 14, 17, 18, 21, 25, 25, 27, 28, 29, 29, 33, 34, 38, 38, 39, 40, 42, 43, 43, 51, 52, 58, 62, 66, 67, 67, 68, 68, 70, 79, 82, 83, 84, 84, 85, 88, 89, 94, 95, 96, 97, 100, 105, 106, 113, 113, 115, 115, 116, 117, 121, 123, 124, 127, 129, 129, 129, 134, 135, 139, 142, 142, 144, 145, 150, 155, 156, 157, 158, 159, 163, 164, 165, 167, 167, 167, 172, 173, 175, 178, 180, 184, 185, 189, 192, 193, 193, 195, 197, 200, 200]
--x Menu de Ordenação x--
1 - InsertionSort
2 - SelectionSort
3 - BubbleSort
4 - Sair

-> Qual a sua escolha: 3
-> Informacoes:
- Tempo Execuciao: 0.000009 seconds.
- Comparacoes: 4950
- Movimentacoes: 0
- Memoria (bytes): 400
- Valores do vetor: [2, 3, 4, 4, 9, 10, 12, 12, 14, 17, 18, 21, 25, 25, 27, 28, 29, 29, 33, 34, 38, 38, 39, 40, 42, 43, 43, 51, 52, 58, 62, 66, 67, 67, 68, 68, 70, 79, 82, 83, 84, 84, 85, 88, 89, 94, 95, 96, 97, 100, 105, 106, 113, 113, 115, 115, 116, 117, 121, 123, 124, 127, 129, 129, 129, 134, 135, 139, 142, 142, 144, 145, 150, 155, 156, 157, 158, 159, 163, 164, 165, 167, 167, 167, 172, 173, 175, 178, 180, 184, 185, 189, 192, 193, 193, 195, 197, 200, 200]
--x Menu de Ordenação x--
1 - InsertionSort
2 - SelectionSort
3 - BubbleSort
4 - Sair
```

100-ordenado

```
messiasfcm@MessiasFCM:/mnt/c/Users/Messi/OneDrive/Área de Trabalho/roteiro10/roteiro10/ativ3$ gcc main.c -o tp3
messiasfcm@MessiasFCM:/mnt/c/Users/Messi/OneDrive/Área de Trabalho/roteiro10/roteiro10/ativ3$ ./tp3
-> Leitura do arquivo com sucesso!

--x Menu de Ordenação x--
1 - InsertionSort
2 - SelectionSort
3 - BubbleSort
4 - Sair

-> Qual a sua escolha: 1
-> Informacoes:
- Tempo Execuciao: 0.000001 seconds.
- Comparacoes: 99
- Movimentacoes: 0
- Memoria (bytes): 400
- Valores do vetor: [1, 3, 5, 7, 9, 11, 13, 15, 17, 19, 21, 23, 25, 27, 29, 31, 33, 35, 37, 39, 41, 43, 45, 47, 49, 51, 53, 55, 57, 59, 61, 63, 65, 67, 69, 71, 73, 75, 77, 79, 81, 83, 85, 87, 89, 91, 93, 95, 97, 99, 101, 103, 105, 107, 109, 111, 113, 115, 117, 119, 121, 123, 125, 127, 129, 131, 133, 135, 137, 139, 141, 143, 145, 147, 149, 151, 153, 155, 157, 159, 161, 163, 165, 167, 169, 171, 173, 175, 177, 179, 181, 183, 185, 187, 189, 191, 193, 195, 197, 199]
--x Menu de Ordenação x--
1 - InsertionSort
2 - SelectionSort
3 - BubbleSort
4 - Sair

-> Qual a sua escolha: 2
-> Informacoes:
- Tempo Execuciao: 0.000007 seconds.
- Comparacoes: 4950
- Movimentacoes: 0
- Memoria (bytes): 400
- Valores do vetor: [1, 3, 5, 7, 9, 11, 13, 15, 17, 19, 21, 23, 25, 27, 29, 31, 33, 35, 37, 39, 41, 43, 45, 47, 49, 51, 53, 55, 57, 59, 61, 63, 65, 67, 69, 71, 73, 75, 77, 79, 81, 83, 85, 87, 89, 91, 93, 95, 97, 99, 101, 103, 105, 107, 109, 111, 113, 115, 117, 119, 121, 123, 125, 127, 129, 131, 133, 135, 137, 139, 141, 143, 145, 147, 149, 151, 153, 155, 157, 159, 161, 163, 165, 167, 169, 171, 173, 175, 177, 179, 181, 183, 185, 187, 189, 191, 193, 195, 197, 199]
--x Menu de Ordenação x--
1 - InsertionSort
2 - SelectionSort
3 - BubbleSort
4 - Sair

-> Qual a sua escolha: 3
-> Informacoes:
- Tempo Execuciao: 0.000010 seconds.
- Comparacoes: 4950
- Movimentacoes: 0
- Memoria (bytes): 400
- Valores do vetor: [1, 3, 5, 7, 9, 11, 13, 15, 17, 19, 21, 23, 25, 27, 29, 31, 33, 35, 37, 39, 41, 43, 45, 47, 49, 51, 53, 55, 57, 59, 61, 63, 65, 67, 69, 71, 73, 75, 77, 79, 81, 83, 85, 87, 89, 91, 93, 95, 97, 99, 101, 103, 105, 107, 109, 111, 113, 115, 117, 119, 121, 123, 125, 127, 129, 131, 133, 135, 137, 139, 141, 143, 145, 147, 149, 151, 153, 155, 157, 159, 161, 163, 165, 167, 169, 171, 173, 175, 177, 179, 181, 183, 185, 187, 189, 191, 193, 195, 197, 199]
--x Menu de Ordenação x--
1 - InsertionSort
2 - SelectionSort
3 - BubbleSort
4 - Sair
```

100-quaseordenado

```
messiasfcm@MessiasFCM:/mnt/c/Users/Messi/OneDrive/Área de Trabalho/roteiro10/roteiro10/ativ3$ gcc main.c -o tp3
messiasfcm@MessiasFCM:/mnt/c/Users/Messi/OneDrive/Área de Trabalho/roteiro10/roteiro10/ativ3$ ./tp3
-> Leitura do arquivo com sucesso!

--x Menu de Ordenação x--
1 - InsertionSort
2 - SelectionSort
3 - BubbleSort
4 - Sair

-> Qual a sua escolha: 1
-> Informacoes:
- Tempo Execuciao: 0.000003 seconds.
- Comparacoes: 460
- Movimentacoes: 361
- Memoria (bytes): 400
- Valores do vetor: [1, 2, 3, 4, 5, 6, 7, 8, 9, 10, 11, 12, 13, 14, 15, 16, 17, 18, 19, 20, 21, 22, 23, 24, 25, 26, 27, 28, 29, 30, 31, 32, 33, 34, 35, 36, 37, 38, 39, 40, 41, 42, 43, 44, 45, 46, 47, 48, 49, 50, 51, 52, 53, 54, 55, 56, 57, 58, 59, 60, 61, 62, 63, 64, 65, 66, 67, 68, 69, 70, 71, 72, 73, 74, 75, 76, 77, 78, 79, 80, 81, 82, 83, 84, 85, 86, 87, 88, 89, 90, 91, 92, 93, 94, 95, 96, 97, 98, 99, 100]
--x Menu de Ordenação x--
1 - InsertionSort
2 - SelectionSort
3 - BubbleSort
4 - Sair

-> Qual a sua escolha: 2
-> Informacoes:
- Tempo Execuciao: 0.000014 seconds.
- Comparacoes: 4950
- Movimentacoes: 0
- Memoria (bytes): 400
- Valores do vetor: [1, 2, 3, 4, 5, 6, 7, 8, 9, 10, 11, 12, 13, 14, 15, 16, 17, 18, 19, 20, 21, 22, 23, 24, 25, 26, 27, 28, 29, 30, 31, 32, 33, 34, 35, 36, 37, 38, 39, 40, 41, 42, 43, 44, 45, 46, 47, 48, 49, 50, 51, 52, 53, 54, 55, 56, 57, 58, 59, 60, 61, 62, 63, 64, 65, 66, 67, 68, 69, 70, 71, 72, 73, 74, 75, 76, 77, 78, 79, 80, 81, 82, 83, 84, 85, 86, 87, 88, 89, 90, 91, 92, 93, 94, 95, 96, 97, 98, 99, 100]
--x Menu de Ordenação x--
1 - InsertionSort
2 - SelectionSort
3 - BubbleSort
4 - Sair

-> Qual a sua escolha: 3
-> Informacoes:
- Tempo Execuciao: 0.000011 seconds.
- Comparacoes: 4950
- Movimentacoes: 0
- Memoria (bytes): 400
- Valores do vetor: [1, 2, 3, 4, 5, 6, 7, 8, 9, 10, 11, 12, 13, 14, 15, 16, 17, 18, 19, 20, 21, 22, 23, 24, 25, 26, 27, 28, 29, 30, 31, 32, 33, 34, 35, 36, 37, 38, 39, 40, 41, 42, 43, 44, 45, 46, 47, 48, 49, 50, 51, 52, 53, 54, 55, 56, 57, 58, 59, 60, 61, 62, 63, 64, 65, 66, 67, 68, 69, 70, 71, 72, 73, 74, 75, 76, 77, 78, 79, 80, 81, 82, 83, 84, 85, 86, 87, 88, 89, 90, 91, 92, 93, 94, 95, 96, 97, 98, 99, 100]
--x Menu de Ordenação x--
1 - InsertionSort
2 - SelectionSort
3 - BubbleSort
4 - Sair
```


1000-ordenado

```
-> Qual a sua escolha: 1
-> Informacoes:
- Tempo Execucao: 0.000003 seconds.
- Comparacoes: 999
- Movimentacoes: 0
- Memoria (bytes): 4000
```

```
-> Qual a sua escolha: 2
-> Informacoes:
- Tempo Execucao: 0.000726 seconds.
- Comparacoes: 499500
- Movimentacoes: 0
- Memoria (bytes): 4000
```

```
-> Qual a sua escolha: 3
-> Informacoes:
- Tempo Execucao: 0.000724 seconds.
- Comparacoes: 499500
- Movimentacoes: 0
- Memoria (bytes): 4000
```

1000-quaseordenado

```
-> Qual a sua escolha: 1
-> Informacoes:
- Tempo Execucao: 0.000018 seconds.
- Comparacoes: 7058
- Movimentacoes: 6059
- Memoria (bytes): 4000
```

```
-> Qual a sua escolha: 2
-> Informacoes:
- Tempo Execucao: 0.000654 seconds.
- Comparacoes: 499500
- Movimentacoes: 0
- Memoria (bytes): 4000
```

```
-> Qual a sua escolha: 3
-> Informacoes:
- Tempo Execucao: 0.000743 seconds.
- Comparacoes: 499500
- Movimentacoes: 0
- Memoria (bytes): 4000
```

10000-contrario

```
-> Qual a sua escolha: 1
-> Informacoes:
- Tempo Execucao: 0.088431 seconds.
- Comparacoes: 50004999
- Movimentacoes: 49995000
- Memoria (bytes): 40000
```

```
-> Qual a sua escolha: 2
-> Informacoes:
- Tempo Execucao: 0.078362 seconds.
- Comparacoes: 49995000
- Movimentacoes: 0
- Memoria (bytes): 40000
```

```
-> Qual a sua escolha: 3
-> Informacoes:
- Tempo Execucao: 0.085671 seconds.
- Comparacoes: 49995000
- Movimentacoes: 0
- Memoria (bytes): 40000
```

10000-misturado

```
-> Qual a sua escolha: 1
-> Informacoes:
- Tempo Execucao: 0.043689 seconds.
- Comparacoes: 24789169
- Movimentacoes: 24779170
- Memoria (bytes): 40000
```

```
-> Qual a sua escolha: 2
-> Informacoes:
- Tempo Execucao: 0.056736 seconds.
- Comparacoes: 49995000
- Movimentacoes: 0
- Memoria (bytes): 40000
```

```
-> Qual a sua escolha: 3
-> Informacoes:
- Tempo Execucao: 0.075127 seconds.
- Comparacoes: 49995000
- Movimentacoes: 0
- Memoria (bytes): 40000
```

10000-ordenado

```
-> Qual a sua escolha: 1
-> Informacoes:
- Tempo Execucao: 0.000024 seconds.
- Comparacoes: 9999
- Movimentacoes: 0
- Memoria (bytes): 40000
```

```
-> Qual a sua escolha: 2
-> Informacoes:
- Tempo Execucao: 0.069230 seconds.
- Comparacoes: 49995000
- Movimentacoes: 0
- Memoria (bytes): 40000
```

```
-> Qual a sua escolha: 3
-> Informacoes:
- Tempo Execucao: 0.081778 seconds.
- Comparacoes: 49995000
- Movimentacoes: 0
- Memoria (bytes): 40000
```

10000-quaseordenado

<pre>-> Qual a sua escolha: 1 -> Informacoes: - Tempo Execucao: 0.000163 seconds. - Comparacoes: 86405 - Movimentacoes: 76406 - Memoria (bytes): 40000</pre>	<pre>-> Qual a sua escolha: 2 -> Informacoes: - Tempo Execucao: 0.059886 seconds. - Comparacoes: 49995000 - Movimentacoes: 0 - Memoria (bytes): 40000</pre>
<pre>-> Qual a sua escolha: 3 -> Informacoes: - Tempo Execucao: 0.077770 seconds. - Comparacoes: 49995000 - Movimentacoes: 0 - Memoria (bytes): 40000</pre>	

100000-contrario

<pre>-> Qual a sua escolha: 1 -> Informacoes: - Tempo Execucao: 10.904682 seconds. - Comparacoes: 705082703 - Movimentacoes: 704982704 - Memoria (bytes): 400000</pre>	<pre>-> Qual a sua escolha: 2 -> Informacoes: - Tempo Execucao: 7.598950 seconds. - Comparacoes: 704982704 - Movimentacoes: 0 - Memoria (bytes): 400000</pre>
<pre>-> Qual a sua escolha: 3 -> Informacoes: - Tempo Execucao: 8.044225 seconds. - Comparacoes: 704982704 - Movimentacoes: 0 - Memoria (bytes): 400000</pre>	

100000-misturado

<pre>-> Qual a sua escolha: 1 -> Informacoes: - Tempo Execucao: 5.735286 seconds. - Comparacoes: -1000173298 - Movimentacoes: -1000273297 - Memoria (bytes): 400000</pre>	<pre>-> Qual a sua escolha: 2 -> Informacoes: - Tempo Execucao: 8.000152 seconds. - Comparacoes: 704982704 - Movimentacoes: 0 - Memoria (bytes): 400000</pre>
<pre>-> Qual a sua escolha: 3 -> Informacoes: - Tempo Execucao: 9.500586 seconds. - Comparacoes: 704982704 - Movimentacoes: 0 - Memoria (bytes): 400000</pre>	

100000-ordenado

<pre>-> Qual a sua escolha: 1 -> Informacoes: - Tempo Execucao: 0.000233 seconds. - Comparacoes: 99999 - Movimentacoes: 0 - Memoria (bytes): 400000</pre>	<pre>-> Qual a sua escolha: 2 -> Informacoes: - Tempo Execucao: 7.038920 seconds. - Comparacoes: 704982704 - Movimentacoes: 0 - Memoria (bytes): 400000</pre>
<pre>-> Qual a sua escolha: 3 -> Informacoes: - Tempo Execucao: 8.514745 seconds. - Comparacoes: 704982704 - Movimentacoes: 0 - Memoria (bytes): 400000</pre>	

100000-quaseordenado

<pre>-> Qual a sua escolha: 1 -> Informacoes: - Tempo Execucao: 0.000005 seconds. - Comparacoes: 416425 - Movimentacoes: 316426 - Memoria (bytes): 400000</pre>	<pre>-> Qual a sua escolha: 2 -> Informacoes: - Tempo Execucao: 7.915503 seconds. - Comparacoes: 704982704 - Movimentacoes: 0 - Memoria (bytes): 400000</pre>
---	---

```
-> Qual a sua escolha: 3
-> Informacoes:
- Tempo Execucao: 10.068680 seconds.
- Comparacoes: 704982704
- Movimentacoes: 0
- Memoria (bytes): 400000
```

1.4

main.c

```
atv4 > C main.c > main()
1  #include <stdio.h>
2  #include <stdlib.h>
3  #include <time.h>
4  #include <string.h>
5
6  typedef struct sPessoa{
7      char nome[50];
8      int idade;
9  }Pessoa;
10
11 int comp;
12 int mov;
13
14 void preencheVetorPessoa(Pessoa* v, int n){
15     int i;
16     for(i=0; i<n; i++){
17         while(getchar() != '\n');
18         printf("Digite o nome: ");
19         scanf("%s", v[i].nome);
20         printf("Digite a idade: ");
21         scanf("%d", &v[i].idade);
22     }
23 }
24
25 void imprimeVetorPessoa(Pessoa* v, int n){
26     int i, prim = 1;
27     printf("[");
28     for(i=0; i<n; i++)
29         if(prim){ printf("(%s, %d)", v[i].nome, v[i].idade); prim = 0; }
30         else printf(", (%s, %d)", v[i].nome, v[i].idade);
31     printf("]\n");
32 }
33
34 void trocaPessoa(Pessoa* a, Pessoa *b){
35     char auxNome[50];
36     int auxIdade;
37     strcpy(auxNome, a->nome);
38     auxIdade = a->idade;
39     strcpy(a->nome, b->nome);
```



```

40     a->idade = b->idade;
41     strcpy(b->nome, auxNome);
42     b->idade = auxIdade;
43 }
44
45 int comparaPessoa(Pessoa* a, Pessoa* b){
46     if(strcmp(a->nome, b->nome) < 0) return 1;
47     else if(strcmp(a->nome, b->nome) > 0) return 0;
48     else{
49         if(a->idade < b->idade) return 1;
50         else return 0;
51     }
52 }
53
54 void SelectionSortPessoa(Pessoa *v, int n, int crescente){
55     int i, j, min_idx;
56     for(i = 0; i < n-1; i++){
57         min_idx = i;
58         for (j = i+1; j < n; j++){
59             comp++;
60             if (crescente ? comparaPessoa(&v[j], &v[min_idx]) : !comparaPessoa(&v[j], &v[min_idx]))
61                 min_idx = j;
62         }
63         trocaPessoa(&v[min_idx], &v[i]);
64         mov++;
65     }
66 }
67
68 void InsertionSortPessoa(Pessoa *v, int n, int crescente){
69     int i, j;
70     Pessoa key;
71
72     for(i = 1; i < n; i++){
73         key = v[i];
74         j = i - 1;
75         while(j >= 0 && (crescente ? comparaPessoa(&v[j], &key) : !comparaPessoa(&v[j], &key))){
76             comp++;

```



```

77         v[j + 1] = v[j];
78         j = j - 1;
79         mov++;
80     }
81     v[j + 1] = key;
82     mov++;
83 }
84 }
85
86 int main(){
87     clock_t t;
88     Pessoa *v;
89     int n, escolha;
90
91     printf("-> Digite o tamanho do vetor: ");
92     scanf("%d", &n);
93     v = (Pessoa*) malloc (n*sizeof(Pessoa));
94
95     preencheVetorPessoa(v, n);
96
97     do{
98         printf("-> Menu de Ordenação x=-\n");
99         printf("1 - InsertionSort (Crescente)\n");
100        printf("2 - InsertionSort (Decrescente)\n");
101        printf("3 - SelectionSort (Crescente)\n");
102        printf("4 - SelectionSort (Decrescente)\n");
103        printf("5 - Sair\n\n");
104        printf("-> Qual a sua escolha: ");
105        scanf("%d", &escolha);
106
107        comp = 0; mov = 0;
108        if(escolha == 1){
109            t = clock();
110            InsertionSortPessoa(v, n, 0);
111            t = clock() - t;
112        }else if(escolha == 2){
113            t = clock();
114            InsertionSortPessoa(v, n, 1);
115            t = clock() - t;
116        }else if(escolha == 3){
117            t = clock();
118            SelectionSortPessoa(v, n, 1);
119            t = clock() - t;
120        }else if(escolha == 4){
121            t = clock();
122            SelectionSortPessoa(v, n, 0);
123            t = clock() - t;
124        }else{
125            printf("-> Escolha Inválida!");
126            break;
127        }
128
129        printf("-> Informacoes:\n");
130        printf("- Tempo Execucao: %f seconds.\n", ((float)t)/CLOCKS_PER_SEC);
131        printf("- Comparacoes: %d\n", comp);
132        printf("- Movimentacoes: %d\n", mov);
133        printf("- Memoria (bytes): %ld\n", n*sizeof(int));
134        printf("- Valores do vetor: ");
135        imprimeVetorPessoa(v, n);
136    }while(escolha!=5);
137
138    free(v);
139    return 0;
140 }
141

```

Console

```
messiasfcm@MessiasFCM:/mnt/c/Users/Messi/OneDrive/Área de Trabalho/roteiro10/roteiro10/atv4$ gcc main.c -o tp4
messiasfcm@MessiasFCM:/mnt/c/Users/Messi/OneDrive/Área de Trabalho/roteiro10/roteiro10/atv4$ ./tp4
-> Digite o tamanho do vetor: 4
Digite o nome: Arthur
Digite a idade: 20
Digite o nome: Arthur
Digite a idade: 21
Digite o nome: Bianca
Digite a idade: 19
Digite o nome: Carlos
Digite a idade: 10
-=x Menu de Ordenação x=-
1 - InsertionSort (Crescente)
2 - InsertionSort (Decrescente)
3 - SelectionSort (Crescente)
4 - SelectionSort (Decrescente)
5 - Sair

-> Qual a sua escolha: 1
-> Informacoes:
- Tempo Execucao: 0.000003 seconds.
- Comparacoes: 0
- Movimentacoes: 3
- Memoria (bytes): 16
- Valores do vetor: [(Arthur, 20), (Arthur, 21), (Bianca, 19), (Carlos, 10)]
-=x Menu de Ordenação x=-
1 - InsertionSort (Crescente)
2 - InsertionSort (Decrescente)
3 - SelectionSort (Crescente)
4 - SelectionSort (Decrescente)
5 - Sair

-> Qual a sua escolha: 2
-> Informacoes:
- Tempo Execucao: 0.000001 seconds.
- Comparacoes: 6
- Movimentacoes: 9
- Memoria (bytes): 16
- Valores do vetor: [(Carlos, 10), (Bianca, 19), (Arthur, 21), (Arthur, 20)]
-=x Menu de Ordenação x=-
1 - InsertionSort (Crescente)
2 - InsertionSort (Decrescente)
```

```
3 - SelectionSort (Crescente)
4 - SelectionSort (Decrescente)
5 - Sair

-> Qual a sua escolha: 3
-> Informacoes:
- Tempo Execucao: 0.000001 seconds.
- Comparacoes: 6
- Movimentacoes: 3
- Memoria (bytes): 16
- Valores do vetor: [(Arthur, 20), (Arthur, 21), (Bianca, 19), (Carlos, 10)]
-=x Menu de Ordenação x=-
1 - InsertionSort (Crescente)
2 - InsertionSort (Decrescente)
3 - SelectionSort (Crescente)
4 - SelectionSort (Decrescente)
5 - Sair

-> Qual a sua escolha: 4
-> Informacoes:
- Tempo Execucao: 0.000002 seconds.
- Comparacoes: 6
- Movimentacoes: 3
- Memoria (bytes): 16
- Valores do vetor: [(Carlos, 10), (Bianca, 19), (Arthur, 21), (Arthur, 20)]
-=x Menu de Ordenação x=-
1 - InsertionSort (Crescente)
2 - InsertionSort (Decrescente)
3 - SelectionSort (Crescente)
4 - SelectionSort (Decrescente)
5 - Sair
```

```
-> Qual a sua escolha: 5
```

```
messiasfcm@MessiasFCM:/mnt/c/Users/Messi/OneDrive/Área de Trabalho/roteiro10/roteiro10/atv4$
```