

Victor Hugo Martins de Oliveira

Selection e Bubble Sort.

Unicentro – CEDETEG
Guarapuava/Paraná

I – Selection Sort.

Iteration	Current State of Array	Min Index	Swapped Elements (if any)
Initial	{7, 2, 5, 4, 9}	-	-
1	{7, 2, 5, 4, 9}	1	7 ↔ 2
After 1	{2, 7, 5, 4, 9}	-	-
2	{2, 7, 5, 4, 9}	3	7 ↔ 4
After 2	{2, 4, 5, 7, 9}	-	-
3	{2, 4, 5, 7, 9}	2	No swap
After 3	{2, 4, 5, 7, 9}	-	-
4	{2, 4, 5, 7, 9}	3	No swap
After 4	{2, 4, 5, 7, 9}	-	-

Iteration	Current State of Array	Min Index	Swapped Elements (if any)
Initial	{7, 5, 5, 3, 9}	-	-
1	{7, 5, 5, 3, 9}	3	7 ↔ 3
After 1	{3, 5, 5, 7, 9}	-	-
2	{3, 5, 5, 7, 9}	1	No swap
After 2	{3, 5, 5, 7, 9}	-	-
3	{3, 5, 5, 7, 9}	2	No swap
After 3	{3, 5, 5, 7, 9}	-	-
4	{3, 5, 5, 7, 9}	3	No swap
After 4	{3, 5, 5, 7, 9}	-	-

Iteration	Current State of Array	Min Index	Swapped Elements (if any)
Initial	{1, 2, 3, 4}	-	-
1	{1, 2, 3, 4}	0	No swap
After 1	{1, 2, 3, 4}	-	-
2	{1, 2, 3, 4}	1	No swap
After 2	{1, 2, 3, 4}	-	-
3	{1, 2, 3, 4}	2	No swap
After 3	{1, 2, 3, 4}	-	-
4	{1, 2, 3, 4}	3	No swap
After 4	{1, 2, 3, 4}	-	-

Iteration	Current State of Array	Min Index	Swapped Elements (if any)
Initial	{4, 3, 2, 1}	-	-
1	{4, 3, 2, 1}	3	4 ↔ 1
After 1	{1, 3, 2, 4}	-	-
2	{1, 3, 2, 4}	2	3 ↔ 2
After 2	{1, 2, 3, 4}	-	-
3	{1, 2, 3, 4}	2	No swap
After 3	{1, 2, 3, 4}	-	-
4	{1, 2, 3, 4}	3	No swap
After 4	{1, 2, 3, 4}	-	-

II – Caso a gente troque a linha por “ $i < n$ ”, haverá uma posição a mais no vetor que não está em seu escopo podendo ocorrer comparação com lixo de memória.

III – O elemento será mantido na posição original, pois, a comparação é apenas com “ $<$ ” e não “ $<=$ ”.

IV – Em v3 nenhuma e em v4 duas.

V -

```
void selection(int *v, int n) {
    int i = 0, j = 0, maior = 0;
    for(i = 0; i < n - 1; i++) {
        maior = i;
        for(j = i + 1; j < n; j++) {
            if(v[j] > v[maior]) {
                maior = j;
            }
        }
        if(v[maior] != v[i]) {
            int aux = v[maior];
            v[maior] = v[i];
            v[i] = aux;
        }
    }
}
```

I – Bubble Sort.

Pass	Current State of Array	Swaps Made
Initial	{7, 2, 5, 4, 9}	-
1	{2, 7, 5, 4, 9}	7 ↔ 2
	{2, 5, 7, 4, 9}	7 ↔ 5
	{2, 5, 4, 7, 9}	7 ↔ 4
	{2, 5, 4, 7, 9}	-
2	{2, 5, 4, 7, 9}	-
	{2, 4, 5, 7, 9}	5 ↔ 4
	{2, 4, 5, 7, 9}	-
3	{2, 4, 5, 7, 9}	-
	{2, 4, 5, 7, 9}	-
4	{2, 4, 5, 7, 9}	-

Pass	Current State of Array	Swaps Made
Initial	{7, 5, 5, 3, 9}	-
1	{5, 7, 5, 3, 9}	7 ↔ 5
	{5, 5, 7, 3, 9}	7 ↔ 5
	{5, 5, 3, 7, 9}	7 ↔ 3
	{5, 5, 3, 7, 9}	-
2	{5, 5, 3, 7, 9}	-
	{5, 3, 5, 7, 9}	5 ↔ 3
	{5, 3, 5, 7, 9}	-
3	{3, 5, 5, 7, 9}	5 ↔ 3
4	{3, 5, 5, 7, 9}	-

Pass	Current State of Array	Swaps Made
Initial	{1, 2, 3, 4}	-
1	{1, 2, 3, 4}	-
2	{1, 2, 3, 4}	-
3	{1, 2, 3, 4}	-
4	{1, 2, 3, 4}	-

Pass	Current State of Array	Swaps Made
Initial	{4, 3, 2, 1}	-
1	{3, 4, 2, 1}	4 ↔ 3
	{3, 2, 4, 1}	4 ↔ 2
	{3, 2, 1, 4}	4 ↔ 1
2	{2, 3, 1, 4}	3 ↔ 2
	{2, 1, 3, 4}	3 ↔ 1
3	{1, 2, 3, 4}	2 ↔ 1
4	{1, 2, 3, 4}	-

II - 5 trocas ocorrem.

III – Há uma maior comparação e trocas no bubble sort, do que no Selection Sort.

IV -

```
void bubble_sort(int *v, int n) {
    if(n == 1) {
        return;
    }
    for (int i = 0; i < n - 1; i++) {
        if (v[i] > v[i + 1]) {
            int aux = v[i];
            v[i] = v[i + 1];
            v[i + 1] = aux;
        }
    }
    bubble_sort(v, n - 1);
}
```


