

Algorithm

Selection Sort

Code Walkthrough





Selection Sort

```
def selection_sort(lst):  
    for i in range(len(lst)):  
  
        min_index = i  
  
        for curr_index in range(i+1, len(lst)):  
            if lst[min_index] > lst[curr_index]:  
                min_index = curr_index  
  
        lst[i], lst[min_index] = lst[min_index], lst[i]
```



Selection Sort



[6, 1, 8, 2, 3]

```
def selection_sort(lst):  
    for i in range(len(lst)):  
  
        min_index = i  
  
        for curr_index in range(i+1, len(lst)):  
            if lst[min_index] > lst[curr_index]:  
                min_index = curr_index  
  
        lst[i], lst[min_index] = lst[min_index], lst[i]
```

=====> Starting Selection Sort <=====

=====> Outer Loop iteration #1

List: [6, 1, 8, 2, 3]
Sorted portion: []
Unsorted portion: [6, 1, 8, 2, 3]
The unsorted portion starts at index: 0

--> Inner Loop iteration

Current element: 1

Min element so far: 6

Is the current element smaller than the min element? Yes
1 is now the new min element. It is located at index: 1

--> Inner Loop iteration

Current element: 8

Min element so far: 1

Is the current element smaller than the min element? No
No need to change the min element

--> Inner Loop iteration

Current element: 2

Min element so far: 1

Is the current element smaller than the min element? No
No need to change the min element

[6, 1, 8, 2, 3]

```
def selection_sort(lst):  
    for i in range(len(lst)):  
  
        min_index = i  
  
        for curr_index in range(i+1, len(lst)):  
            if lst[min_index] > lst[curr_index]:  
                min_index = curr_index  
  
        lst[i], lst[min_index] = lst[min_index], lst[i]
```

Min: 6

min_index: 0

=====> Starting Selection Sort <=====

=====> Outer Loop iteration #1

List: [6, 1, 8, 2, 3]
Sorted portion: []
Unsorted portion: [6, 1, 8, 2, 3]
The unsorted portion starts at index: 0

--> Inner Loop iteration
Current element: 1
Min element so far: 6
Is the current element smaller than the min element? Yes
1 is now the new min element. It is located at index: 1

--> Inner Loop iteration
Current element: 8
Min element so far: 1
Is the current element smaller than the min element? No
No need to change the min element

--> Inner Loop iteration
Current element: 2
Min element so far: 1
Is the current element smaller than the min element? No
No need to change the min element

[6, 1, 8, 2, 3]

```
def selection_sort(lst):  
    for i in range(len(lst)):  
  
        min_index = i  
  
        for curr_index in range(i+1, len(lst)):  
            if lst[min_index] > lst[curr_index]:  
                min_index = curr_index  
  
        lst[i], lst[min_index] = lst[min_index], lst[i]
```

Min: 1

min_index: 1

=====> Starting Selection Sort <=====

=====> Outer Loop iteration #1

List: [6, 1, 8, 2, 3]
Sorted portion: []
Unsorted portion: [6, 1, 8, 2, 3]
The unsorted portion starts at index: 0

--> Inner Loop iteration
Current element: 1
Min element so far: 6
Is the current element smaller than the min element? Yes
1 is now the new min element. It is located at index: 1

--> Inner Loop iteration
Current element: 8
Min element so far: 1
Is the current element smaller than the min element? No
No need to change the min element

--> Inner Loop iteration
Current element: 2
Min element so far: 1
Is the current element smaller than the min element? No
No need to change the min element

[6, 1, 8, 2, 3]

```
def selection_sort(lst):  
    for i in range(len(lst)):  
  
        min_index = i  
  
        for curr_index in range(i+1, len(lst)):  
            if lst[min_index] > lst[curr_index]:  
                min_index = curr_index  
  
        lst[i], lst[min_index] = lst[min_index], lst[i]
```

Min: 1

min_index: 1

--> Inner Loop iteration

Current element: 3

Min element so far: 1

Is the current element smaller than the min element? No

No need to change the min element

-> Out of inner loop

Previous list: [6, 1, 8, 2, 3]

Swapping the first element in the unsorted portion: 6

With the min element found: 1

New list: [1, 6, 8, 2, 3]

[1, 6, 8, 2, 3]

```
def selection_sort(lst):  
    for i in range(len(lst)):  
  
        min_index = i  
  
        for curr_index in range(i+1, len(lst)):  
            if lst[min_index] > lst[curr_index]:  
                min_index = curr_index  
  
        lst[i], lst[min_index] = lst[min_index], lst[i]
```

--> Inner Loop iteration

Current element: 3

Min element so far: 1

Is the current element smaller than the min element? No

No need to change the min element

-> Out of inner loop

Previous list: [6, 1, 8, 2, 3]

Swapping the first element in the unsorted portion: 6

With the min element found: 1

New list: [1, 6, 8, 2, 3]

[1, 6, 8, 2, 3]

```
def selection_sort(lst):  
    for i in range(len(lst)):  
  
        min_index = i  
  
        for curr_index in range(i+1, len(lst)):  
            if lst[min_index] > lst[curr_index]:  
                min_index = curr_index  
  
        lst[i], lst[min_index] = lst[min_index], lst[i]
```

=====> Outer Loop iteration #2

List: [1, 6, 8, 2, 3]
Sorted portion: [1]
Unsorted portion: [6, 8, 2, 3]
The unsorted portion starts at index: 1

--> Inner Loop iteration
Current element: 8
Min element so far: 6
Is the current element smaller than the min element? No
No need to change the min element

--> Inner Loop iteration
Current element: 2
Min element so far: 6
Is the current element smaller than the min element? Yes
2 is now the new min element. It is located at index: 3

--> Inner Loop iteration
Current element: 3
Min element so far: 2
Is the current element smaller than the min element? No
No need to change the min element

-> Out of inner loop
Previous list: [1, 6, 8, 2, 3]

Swapping the first element in the unsorted portion: 6
With the min element found: 2
New list: [1, 2, 8, 6, 3]



[1, 6, 8, 2, 3]

```
def selection_sort(lst):  
    for i in range(len(lst)):  
  
        min_index = i  
  
        for curr_index in range(i+1, len(lst)):  
            if lst[min_index] > lst[curr_index]:  
                min_index = curr_index  
  
        lst[i], lst[min_index] = lst[min_index], lst[i]
```

Min: 6

min_index: 1

=====> Outer Loop iteration #2

List: [1, 6, 8, 2, 3]
Sorted portion: [1]
Unsorted portion: [6, 8, 2, 3]
The unsorted portion starts at index: 1

--> Inner Loop iteration
Current element: 8
Min element so far: 6
Is the current element smaller than the min element? No
No need to change the min element

--> Inner Loop iteration
Current element: 2
Min element so far: 6
Is the current element smaller than the min element? Yes
2 is now the new min element. It is located at index: 3

--> Inner Loop iteration
Current element: 3
Min element so far: 2
Is the current element smaller than the min element? No
No need to change the min element

-> Out of inner loop
Previous list: [1, 6, 8, 2, 3]

Swapping the first element in the unsorted portion: 6
With the min element found: 2
New list: [1, 2, 8, 6, 3]



[1, 6, 8, 2, 3]

```
def selection_sort(lst):  
    for i in range(len(lst)):  
  
        min_index = i  
  
        for curr_index in range(i+1, len(lst)):  
            if lst[min_index] > lst[curr_index]:  
                min_index = curr_index  
  
        lst[i], lst[min_index] = lst[min_index], lst[i]
```

Min: 2

min_index: 3

=====> Outer Loop iteration #2

List: [1, 6, 8, 2, 3]
Sorted portion: [1]
Unsorted portion: [6, 8, 2, 3]
The unsorted portion starts at index: 1

--> Inner Loop iteration
Current element: 8
Min element so far: 6
Is the current element smaller than the min element? No
No need to change the min element

--> Inner Loop iteration
Current element: 2
Min element so far: 6
Is the current element smaller than the min element? Yes
2 is now the new min element. It is located at index: 3

--> Inner Loop iteration
Current element: 3
Min element so far: 2
Is the current element smaller than the min element? No
No need to change the min element

-> Out of inner loop
Previous list: [1, 6, 8, 2, 3]

Swapping the first element in the unsorted portion: 6
With the min element found: 2
New list: [1, 2, 8, 6, 3]

[1, 2, 8, 6, 3]

```
def selection_sort(lst):
    for i in range(len(lst)):

        min_index = i

        for curr_index in range(i+1, len(lst)):
            if lst[min_index] > lst[curr_index]:
                min_index = curr_index

        lst[i], lst[min_index] = lst[min_index], lst[i]
```

=====> Outer Loop iteration #2

List: [1, 6, 8, 2, 3]
Sorted portion: [1]
Unsorted portion: [6, 8, 2, 3]
The unsorted portion starts at index: 1

--> Inner Loop iteration
Current element: 8
Min element so far: 6
Is the current element smaller than the min element? No
No need to change the min element

--> Inner Loop iteration
Current element: 2
Min element so far: 6
Is the current element smaller than the min element? Yes
2 is now the new min element. It is located at index: 3

--> Inner Loop iteration
Current element: 3
Min element so far: 2
Is the current element smaller than the min element? No
No need to change the min element

-> Out of inner loop
Previous list: [1, 6, 8, 2, 3]

Swapping the first element in the unsorted portion: 6
With the min element found: 2
New list: [1, 2, 8, 6, 3]

[1, 2, 8, 6, 3]

```
def selection_sort(lst):  
    for i in range(len(lst)):  
  
        min_index = i  
  
        for curr_index in range(i+1, len(lst)):  
            if lst[min_index] > lst[curr_index]:  
                min_index = curr_index  
  
        lst[i], lst[min_index] = lst[min_index], lst[i]
```

=====> Outer Loop iteration #3

List: [1, 2, 8, 6, 3]

Sorted portion: [1, 2]

Unsorted portion: [8, 6, 3]

The unsorted portion starts at index: 2

--> Inner Loop iteration

Current element: 6

Min element so far: 8

Is the current element smaller than the min element? Yes
6 is now the new min element. It is located at index: 3

--> Inner Loop iteration

Current element: 3

Min element so far: 6

Is the current element smaller than the min element? Yes
3 is now the new min element. It is located at index: 4

-> Out of inner loop

Previous list: [1, 2, 8, 6, 3]

Swapping the first element in the unsorted portion: 8

With the min element found: 3

New list: [1, 2, 3, 6, 8]

[1, 2, 8, 6, 3]

```
def selection_sort(lst):  
    for i in range(len(lst)):  
  
        min_index = i  
  
        for curr_index in range(i+1, len(lst)):  
            if lst[min_index] > lst[curr_index]:  
                min_index = curr_index  
  
        lst[i], lst[min_index] = lst[min_index], lst[i]
```

Min: 8

min_index: 2

=====> Outer Loop iteration #3

List: [1, 2, 8, 6, 3]

Sorted portion: [1, 2]

Unsorted portion: [8, 6, 3]

The unsorted portion starts at index: 2

--> Inner Loop iteration

Current element: 6

Min element so far: 8

Is the current element smaller than the min element? Yes
6 is now the new min element. It is located at index: 3

--> Inner Loop iteration

Current element: 3

Min element so far: 6

Is the current element smaller than the min element? Yes
3 is now the new min element. It is located at index: 4

-> Out of inner loop

Previous list: [1, 2, 8, 6, 3]

Swapping the first element in the unsorted portion: 8

With the min element found: 3

New list: [1, 2, 3, 6, 8]

[1, 2, 8, 6, 3]

```
def selection_sort(lst):  
    for i in range(len(lst)):  
  
        min_index = i  
  
        for curr_index in range(i+1, len(lst)):  
            if lst[min_index] > lst[curr_index]:  
                min_index = curr_index  
  
        lst[i], lst[min_index] = lst[min_index], lst[i]
```

Min: 6

min_index: 3

=====> Outer Loop iteration #3

List: [1, 2, 8, 6, 3]
Sorted portion: [1, 2]
Unsorted portion: [8, 6, 3]
The unsorted portion starts at index: 2

--> Inner Loop iteration
Current element: 6
Min element so far: 8
Is the current element smaller than the min element? Yes
6 is now the new min element. It is located at index: 3

--> Inner Loop iteration
Current element: 3
Min element so far: 6
Is the current element smaller than the min element? Yes
3 is now the new min element. It is located at index: 4

-> Out of inner loop
Previous list: [1, 2, 8, 6, 3]

Swapping the first element in the unsorted portion: 8
With the min element found: 3
New list: [1, 2, 3, 6, 8]

[1, 2, 8, 6, 3]

```
def selection_sort(lst):  
    for i in range(len(lst)):  
  
        min_index = i  
  
        for curr_index in range(i+1, len(lst)):  
            if lst[min_index] > lst[curr_index]:  
                min_index = curr_index  
  
        lst[i], lst[min_index] = lst[min_index], lst[i]
```

Min: 3

min_index: 4

=====> Outer Loop iteration #3

List: [1, 2, 8, 6, 3]
Sorted portion: [1, 2]
Unsorted portion: [8, 6, 3]
The unsorted portion starts at index: 2

--> Inner Loop iteration
Current element: 6
Min element so far: 8
Is the current element smaller than the min element? Yes
6 is now the new min element. It is located at index: 3

--> Inner Loop iteration
Current element: 3
Min element so far: 6
Is the current element smaller than the min element? Yes
3 is now the new min element. It is located at index: 4

-> Out of inner loop
Previous list: [1, 2, 8, 6, 3]

Swapping the first element in the unsorted portion: 8
With the min element found: 3
New list: [1, 2, 3, 6, 8]

[1, 2, 3, 6, 8]

```
def selection_sort(lst):  
    for i in range(len(lst)):  
  
        min_index = i  
  
        for curr_index in range(i+1, len(lst)):  
            if lst[min_index] > lst[curr_index]:  
                min_index = curr_index  
  
        lst[i], lst[min_index] = lst[min_index], lst[i]
```

=====> Outer Loop iteration #4

List: [1, 2, 3, 6, 8]

Sorted portion: [1, 2, 3]

Unsorted portion: [6, 8]

The unsorted portion starts at index: 3

--> Inner Loop iteration

Current element: 8

Min element so far: 6

Is the current element smaller than the min element? No

No need to change the min element

-> Out of inner loop

Previous list: [1, 2, 3, 6, 8]

Swapping the first element in the unsorted portion: 6

With the min element found: 6

New list: [1, 2, 3, 6, 8]

=====> Outer Loop iteration #5

The list is now sorted!

[1, 2, 3, 6, 8]

[1, 2, 3, 6, 8]

```
def selection_sort(lst):  
    for i in range(len(lst)):  
  
        min_index = i  
  
        for curr_index in range(i+1, len(lst)):  
            if lst[min_index] > lst[curr_index]:  
                min_index = curr_index  
  
        lst[i], lst[min_index] = lst[min_index], lst[i]
```

Min: 6

min_index: 3

=====> Outer Loop iteration #4

List: [1, 2, 3, 6, 8]

Sorted portion: [1, 2, 3]

Unsorted portion: [6, 8]

The unsorted portion starts at index: 3

--> Inner Loop iteration

Current element: 8

Min element so far: 6

Is the current element smaller than the min element? No

No need to change the min element

-> Out of inner loop

Previous list: [1, 2, 3, 6, 8]

Swapping the first element in the unsorted portion: 6

With the min element found: 6

New list: [1, 2, 3, 6, 8]

=====> Outer Loop iteration #5

The list is now sorted!

[1, 2, 3, 6, 8]

[1, 2, 3, 6, 8]

```
def selection_sort(lst):  
    for i in range(len(lst)):  
  
        min_index = i  
  
        for curr_index in range(i+1, len(lst)):  
            if lst[min_index] > lst[curr_index]:  
                min_index = curr_index  
  
        lst[i], lst[min_index] = lst[min_index], lst[i]
```

=====> Outer Loop iteration #4

List: [1, 2, 3, 6, 8]
Sorted portion: [1, 2, 3]
Unsorted portion: [6, 8]
The unsorted portion starts at index: 3

--> Inner Loop iteration

Current element: 8
Min element so far: 6
Is the current element smaller than the min element? No
No need to change the min element

-> Out of inner loop

Previous list: [1, 2, 3, 6, 8]

Swapping the first element in the unsorted portion: 6
With the min element found: 6
New list: [1, 2, 3, 6, 8]

=====> Outer Loop iteration #5

The list is now sorted!
[1, 2, 3, 6, 8]

[1, 2, 3, 6, 8]

```
def selection_sort(lst):  
    for i in range(len(lst)):  
  
        min_index = i  
  
        for curr_index in range(i+1, len(lst)):  
            if lst[min_index] > lst[curr_index]:  
                min_index = curr_index  
  
        lst[i], lst[min_index] = lst[min_index], lst[i]
```

=====> Outer Loop iteration #4

List: [1, 2, 3, 6, 8]

Sorted portion: [1, 2, 3]

Unsorted portion: [6, 8]

The unsorted portion starts at index: 3

--> Inner Loop iteration

Current element: 8

Min element so far: 6

Is the current element smaller than the min element? No

No need to change the min element

-> Out of inner loop

Previous list: [1, 2, 3, 6, 8]

Swapping the first element in the unsorted portion: 6

With the min element found: 6

New list: [1, 2, 3, 6, 8]

=====> Outer Loop iteration #5

The list is now sorted!

[1, 2, 3, 6, 8]



Time to Practice!

