Algorithm

# Selection Sort

## Example | Visual

# Selection Sort

```
[5, 2, 8, 1, 10]
```

# Selection Sort

```
[5, 2, 8, 1, 10]
```

min: 5

# Selection Sort

```
[5, 2, 8, 1, 10]
```

min: 2

# Selection Sort

```
[5, 2, 8, 1, 10]
```

min: 1

# Selection Sort


min: 2

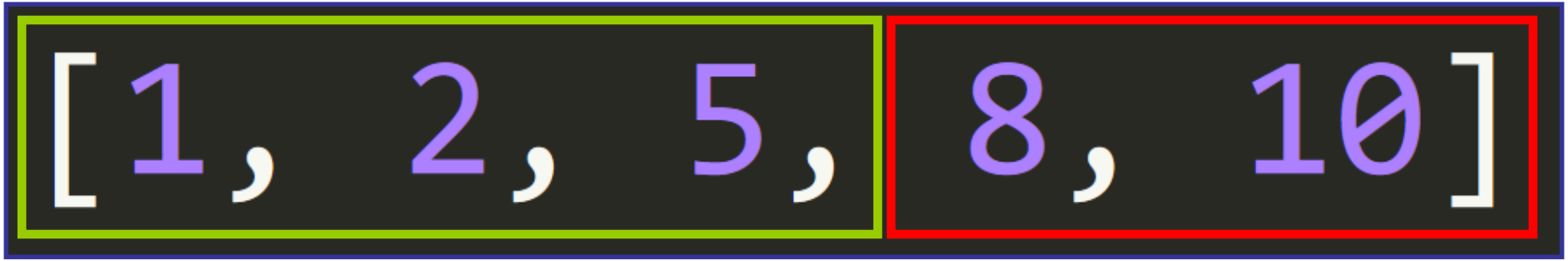
# Selection Sort


min: 8

# Selection Sort

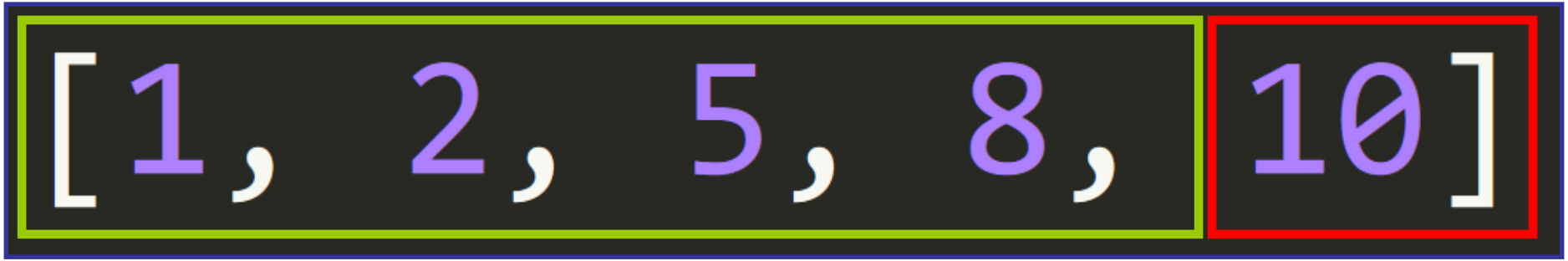

min: 5

# Selection Sort




min: 8

# Selection Sort




min: 8

# Selection Sort

```
[1, 2, 5, 8, 10]
```

# To the Code!

Algorithm

# Selection Sort

## Example | Code

```
[5, 2, 8, 1, 10]
```

```python
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: [5, 2, 8, 1, 10]
Sorted portion: []
Unsorted portion: [5, 2, 8, 1, 10]
The unsorted portion starts at index: 0

--> Inner Loop iteration
Current element: 2
Min element so far: 5
Is the current element smaller than the min element? Yes
2 is now the new min element. It is located at index: 1

--> Inner Loop iteration
Current element: 8
Min element so far: 2
Is the current element smaller than the min element? No
No need to change the min element
```

[5, 2, 8, 1, 10]

```python
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: 5**

**min_index: 0**

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


=========> Outer Loop iteration #1

List: [5, 2, 8, 1, 10]
Sorted portion: []
Unsorted portion: [5, 2, 8, 1, 10]
The unsorted portion starts at index: 0

--> Inner Loop iteration
Current element: 2
Min element so far: 5
Is the current element smaller than the min element? Yes
2 is now the new min element. It is located at index: 1

--> Inner Loop iteration
Current element: 8
Min element so far: 2
Is the current element smaller than the min element? No
No need to change the min element
```

[5, 2, 8, 1, 10]

```python
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: 1**

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


=========> Outer Loop iteration #1

List: [5, 2, 8, 1, 10]
Sorted portion: []
Unsorted portion: [5, 2, 8, 1, 10]
The unsorted portion starts at index: 0

--> Inner Loop iteration
Current element: 2
Min element so far: 5
Is the current element smaller than the min element? Yes
2 is now the new min element. It is located at index: 1

--> Inner Loop iteration
Current element: 8
Min element so far: 2
Is the current element smaller than the min element? No
No need to change the min element
```

[5, 2, 8, 1, 10]

```python
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: 1**

```
--> Inner Loop iteration
Current element: 1
Min element so far: 2
Is the current element smaller than the min element? Yes
1 is now the new min element. It is located at index: 3

--> Inner Loop iteration
Current element: 10
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: [5, 2, 8, 1, 10]

Swapping the first element in the unsorted portion: 5
With the min element found: 1
New list: [1, 2, 8, 5, 10]
```

[5, 2, 8, 1, 10]

```python
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: 3**

```
--> Inner Loop iteration
Current element: 1
Min element so far: 2
Is the current element smaller than the min element? Yes
1 is now the new min element. It is located at index: 3

--> Inner Loop iteration
Current element: 10
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: [5, 2, 8, 1, 10]

Swapping the first element in the unsorted portion: 5
With the min element found: 1
New list: [1, 2, 8, 5, 10]
```


```python
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: 1
Min element so far: 2
Is the current element smaller than the min element? Yes
1 is now the new min element. It is located at index: 3

--> Inner Loop iteration
Current element: 10
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: [5, 2, 8, 1, 10]

Swapping the first element in the unsorted portion: 5
With the min element found: 1
New list: [1, 2, 8, 5, 10]
```

[1, 2, 8, 5, 10]

```
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: 1**

```
=========> Outer Loop iteration #2

List: [1, 2, 8, 5, 10]
Sorted portion: [1]
Unsorted portion: [2, 8, 5, 10]
The unsorted portion starts at index: 1

--> Inner Loop iteration
Current element: 8
Min element so far: 2
Is the current element smaller than the min element? No
No need to change the min element

--> Inner Loop iteration
Current element: 5
Min element so far: 2
Is the current element smaller than the min element? No
No need to change the min element
```

[1, 2, 8, 5, 10]

```python
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: 10
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, 2, 8, 5, 10]

Swapping the first element in the unsorted portion: 2
With the min element found: 2
New list: [1, 2, 8, 5, 10]
```

[1, 2, 8, 5, 10]

```python
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: 10
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, 2, 8, 5, 10]

Swapping the first element in the unsorted portion: 2
With the min element found: 2
New list: [1, 2, 8, 5, 10]
```


```python
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: 5**

**min_index: 2**

```
=========> Outer Loop iteration #3

List: [1, 2, 8, 5, 10]
Sorted portion: [1, 2]
Unsorted portion: [8, 5, 10]
The unsorted portion starts at index: 2

--> Inner Loop iteration
Current element: 5
Min element so far: 8
Is the current element smaller than the min element? Yes
5 is now the new min element. It is located at index: 3

--> Inner Loop iteration
Current element: 10
Min element so far: 5
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, 8, 5, 10]

Swapping the first element in the unsorted portion: 8
With the min element found: 5
New list: [1, 2, 5, 8, 10]
```


```python
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]
```

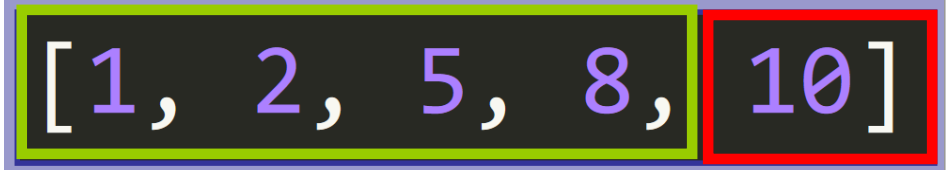
```
=========> Outer Loop iteration #3

List: [1, 2, 8, 5, 10]
Sorted portion: [1, 2]
Unsorted portion: [8, 5, 10]
The unsorted portion starts at index: 2

--> Inner Loop iteration
Current element: 5
Min element so far: 8
Is the current element smaller than the min element? Yes
5 is now the new min element. It is located at index: 3

--> Inner Loop iteration
Current element: 10
Min element so far: 5
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, 8, 5, 10]

Swapping the first element in the unsorted portion: 8
With the min element found: 5
New list: [1, 2, 5, 8, 10]
```


```python
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: 3**

```
=========> Outer Loop iteration #4

List: [1, 2, 5, 8, 10]
Sorted portion: [1, 2, 5]
Unsorted portion: [8, 10]
The unsorted portion starts at index: 3

--> Inner Loop iteration
Current element: 10
Min element so far: 8
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, 5, 8, 10]

Swapping the first element in the unsorted portion: 8
With the min element found: 8
New list: [1, 2, 5, 8, 10]

=========> Outer Loop iteration #5

The list is now sorted!
[1, 2, 5, 8, 10]
```




```python
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, 5, 8, 10]
Sorted portion: [1, 2, 5]
Unsorted portion: [8, 10]
The unsorted portion starts at index: 3

--> Inner Loop iteration
Current element: 10
Min element so far: 8
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, 5, 8, 10]

Swapping the first element in the unsorted portion: 8
With the min element found: 8
New list: [1, 2, 5, 8, 10]

=========> Outer Loop iteration #5

The list is now sorted!
[1, 2, 5, 8, 10]
```

```
[1, 2, 5, 8, 10]
```

```python
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, 5, 8, 10]
Sorted portion: [1, 2, 5]
Unsorted portion: [8, 10]
The unsorted portion starts at index: 3

--> Inner Loop iteration
Current element: 10
Min element so far: 8
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, 5, 8, 10]

Swapping the first element in the unsorted portion: 8
With the min element found: 8
New list: [1, 2, 5, 8, 10]

=========> Outer Loop iteration #5

The list is now sorted!
[1, 2, 5, 8, 10]
```

# Time to Practice!