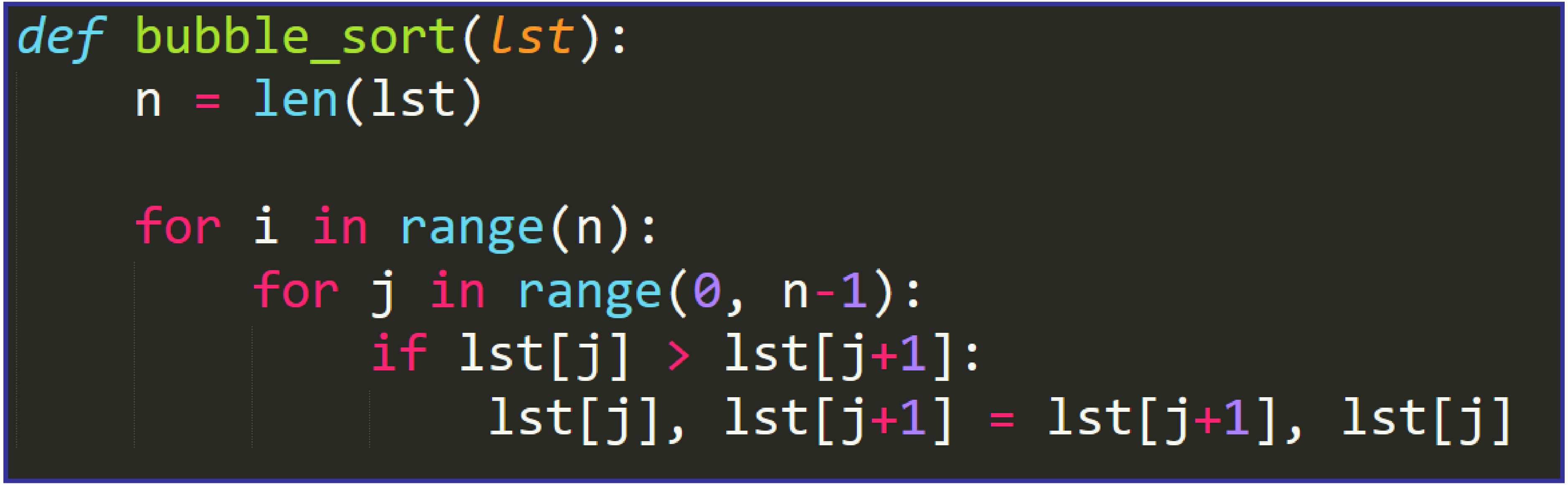
Algorithm

Bubble Sort

Code Walkthrough

Bubble Sort



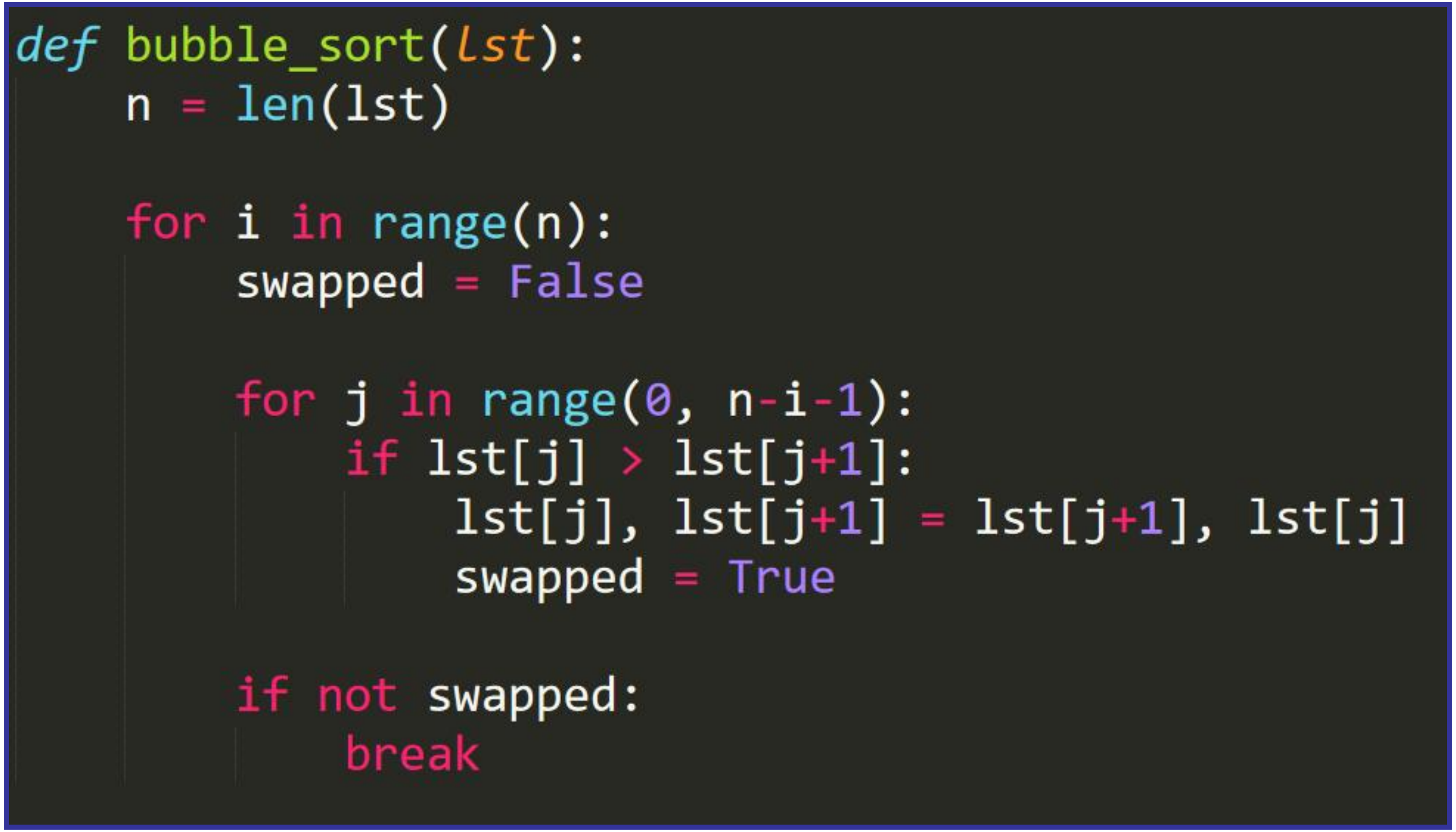
```python
def bubble_sort(lst):
    n = len(lst)

    for i in range(n):
        for j in range(0, n-1):
            if lst[j] > lst[j+1]:
                lst[j], lst[j+1] = lst[j+1], lst[j]
```

Binary Search

Python Searching and Sorting Algorithms: A Practical Approach

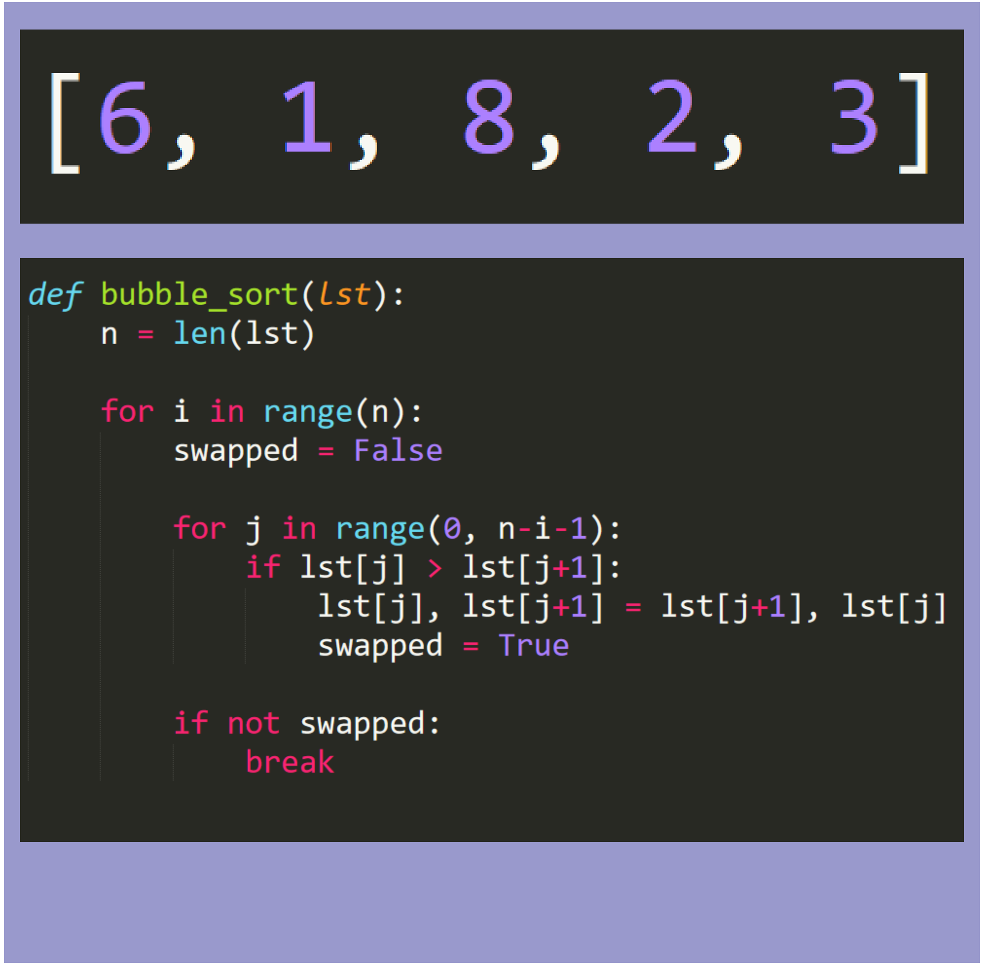
Let’s think about…
How to optimize?

Bubble Sort



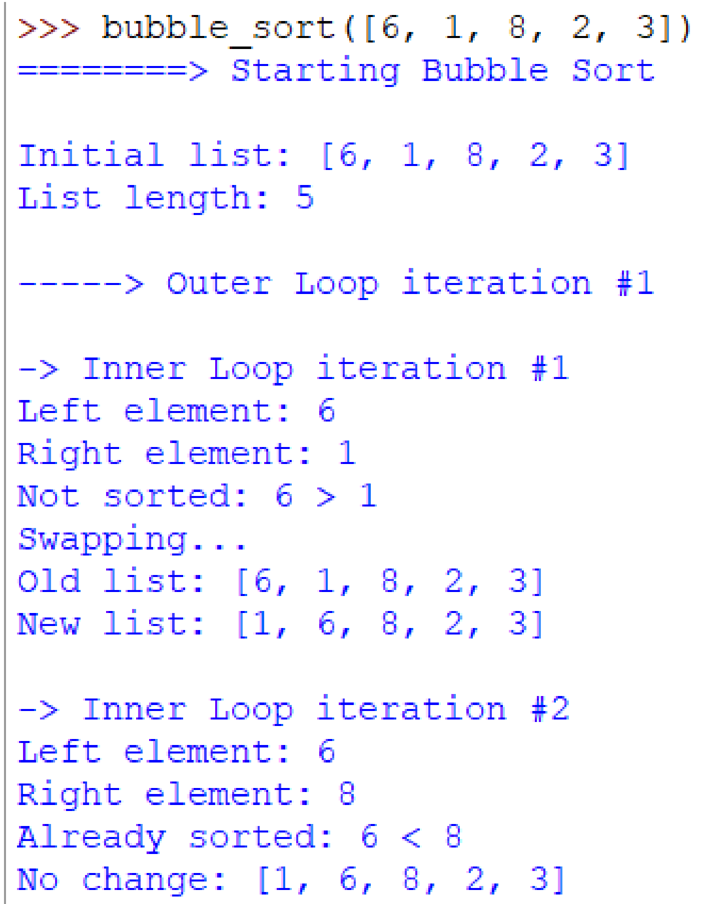
```python
def bubble_sort(lst):
    n = len(lst)

    for i in range(n):
        swapped = False

        for j in range(0, n-i-1):
            if lst[j] > lst[j+1]:
                lst[j], lst[j+1] = lst[j+1], lst[j]
                swapped = True

        if not swapped:
            break
```

Binary Search

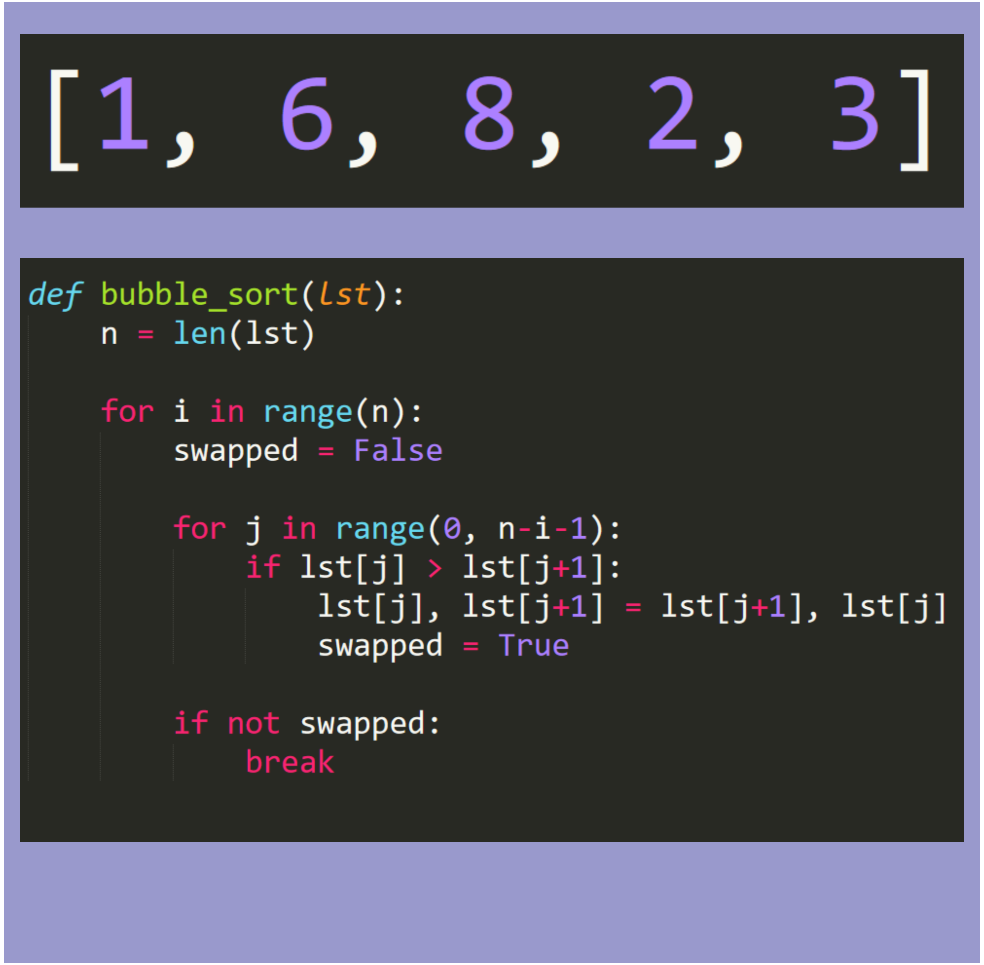


```
[6, 1, 8, 2, 3]
```

```python
def bubble_sort(lst):
    n = len(lst)

    for i in range(n):
        swapped = False

        for j in range(0, n-i-1):
            if lst[j] > lst[j+1]:
                lst[j], lst[j+1] = lst[j+1], lst[j]
                swapped = True

        if not swapped:
            break
```



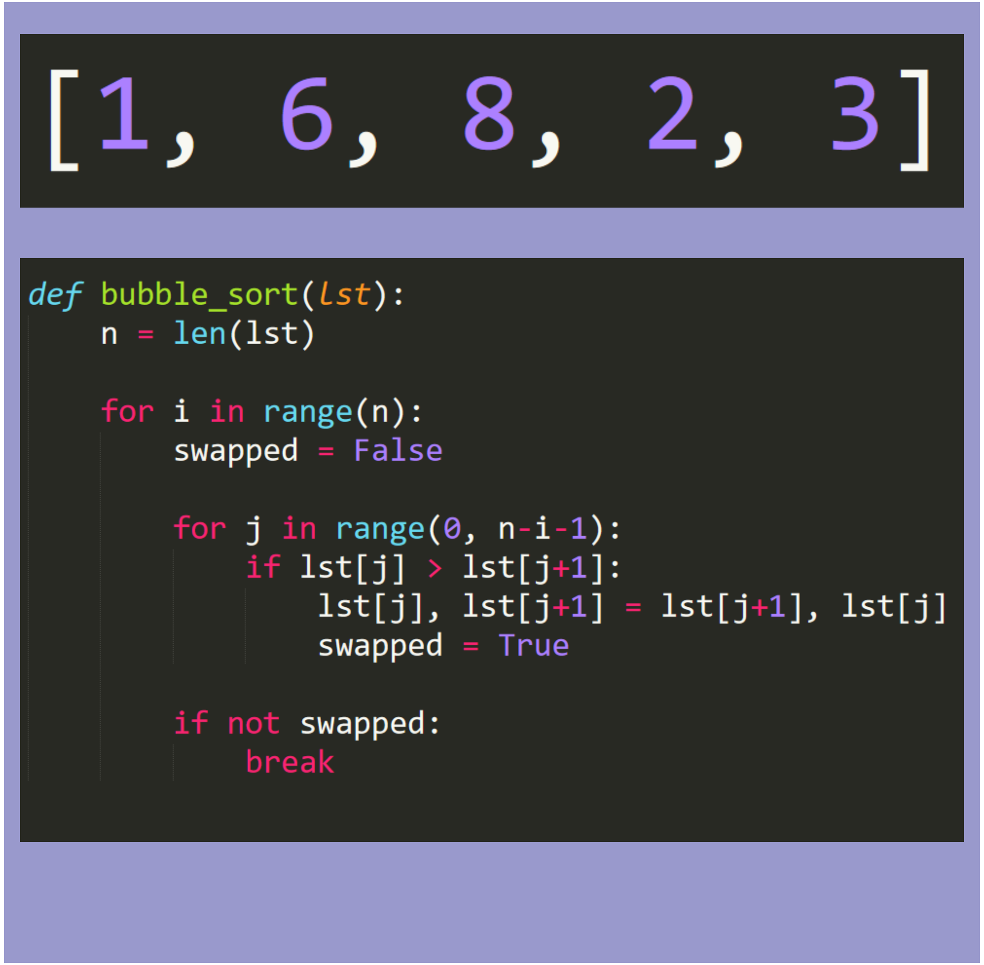
```
>>> bubble_sort([6, 1, 8, 2, 3])
=========> Starting Bubble Sort

Initial list: [6, 1, 8, 2, 3]
List length: 5

-----> Outer Loop iteration #1

-> Inner Loop iteration #1
Left element: 6
Right element: 1
Not sorted: 6 > 1
Swapping...
Old list: [6, 1, 8, 2, 3]
New list: [1, 6, 8, 2, 3]

-> Inner Loop iteration #2
Left element: 6
Right element: 8
Already sorted: 6 < 8
No change: [1, 6, 8, 2, 3]
```



```
[1, 6, 8, 2, 3]
```

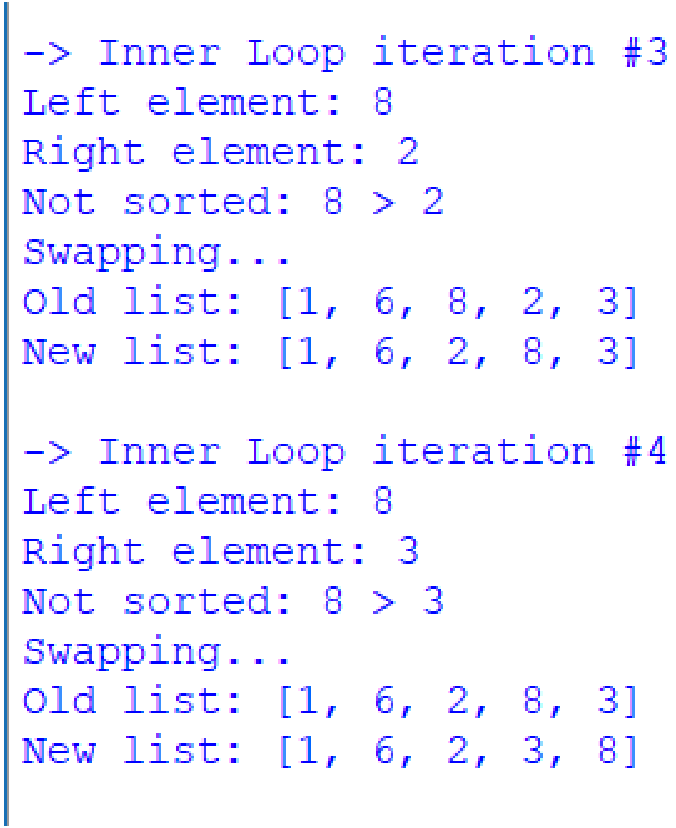
```python
def bubble_sort(lst):
    n = len(lst)

    for i in range(n):
        swapped = False

        for j in range(0, n-i-1):
            if lst[j] > lst[j+1]:
                lst[j], lst[j+1] = lst[j+1], lst[j]
                swapped = True

        if not swapped:
            break
```

```
>>> bubble_sort([6, 1, 8, 2, 3])
========> Starting Bubble Sort

Initial list: [6, 1, 8, 2, 3]
List length: 5

-----> Outer Loop iteration #1

-> Inner Loop iteration #1
Left element: 6
Right element: 1
Not sorted: 6 > 1
Swapping...
Old list: [6, 1, 8, 2, 3]
New list: [1, 6, 8, 2, 3]

-> Inner Loop iteration #2
Left element: 6
Right element: 8
Already sorted: 6 < 8
No change: [1, 6, 8, 2, 3]
```

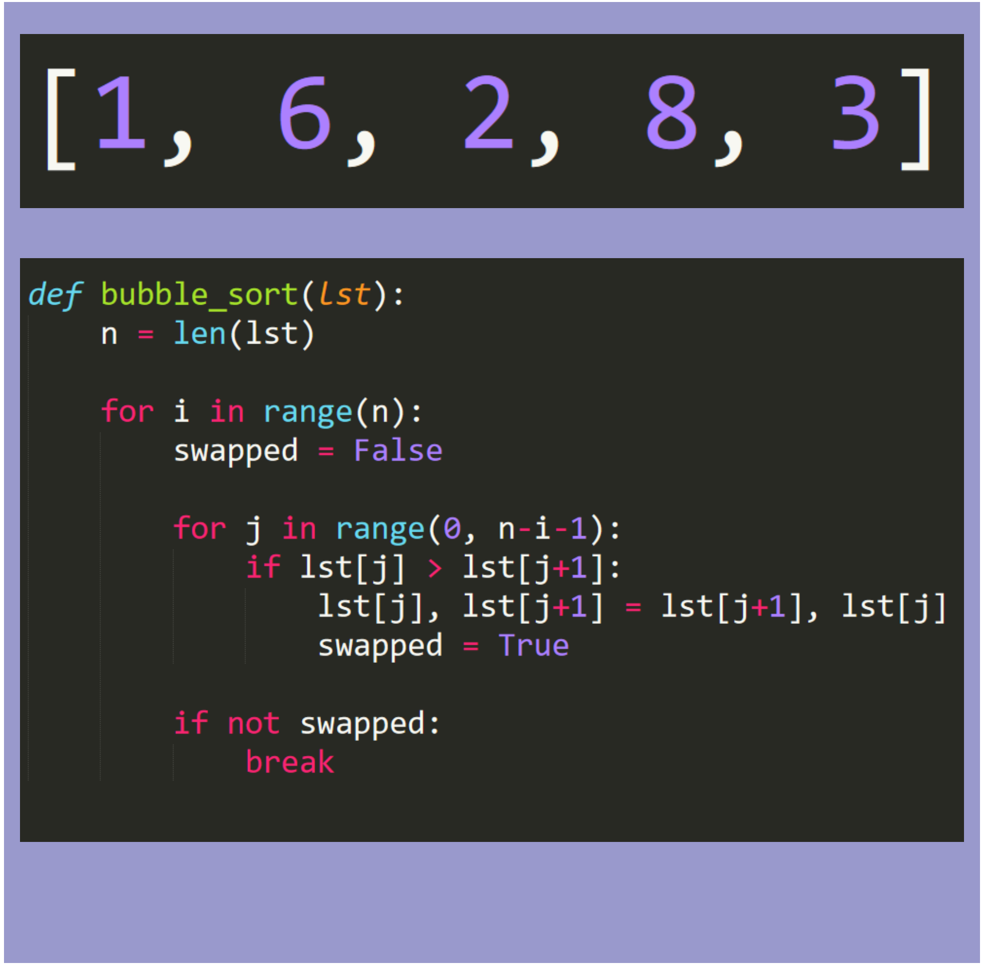


```
[1, 6, 8, 2, 3]
```

```python
def bubble_sort(lst):
    n = len(lst)

    for i in range(n):
        swapped = False

        for j in range(0, n-i-1):
            if lst[j] > lst[j+1]:
                lst[j], lst[j+1] = lst[j+1], lst[j]
                swapped = True

        if not swapped:
            break
```

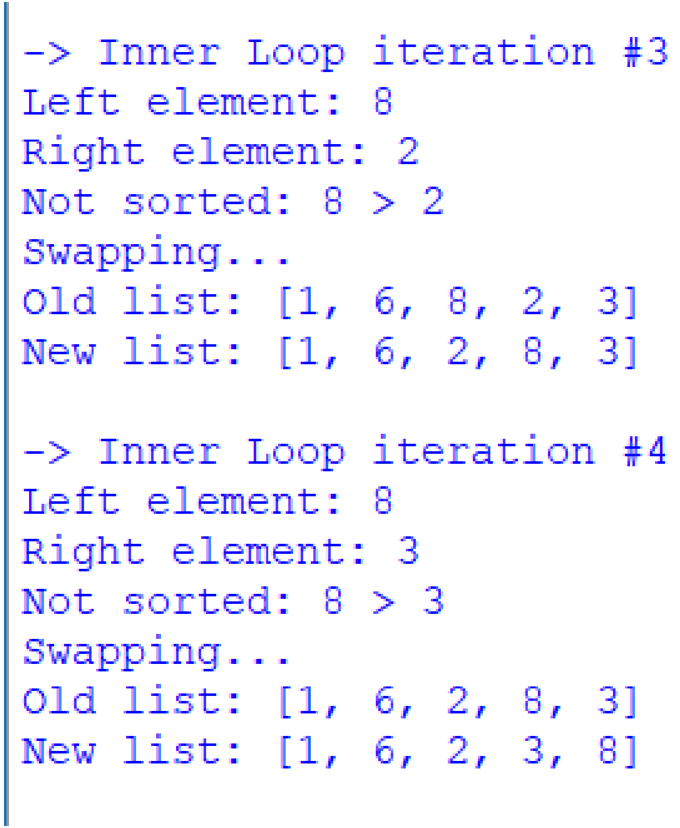


```
-> Inner Loop iteration #3
Left element: 8
Right element: 2
Not sorted: 8 > 2
Swapping...
Old list: [1, 6, 8, 2, 3]
New list: [1, 6, 2, 8, 3]

-> Inner Loop iteration #4
Left element: 8
Right element: 3
Not sorted: 8 > 3
Swapping...
Old list: [1, 6, 2, 8, 3]
New list: [1, 6, 2, 3, 8]
```

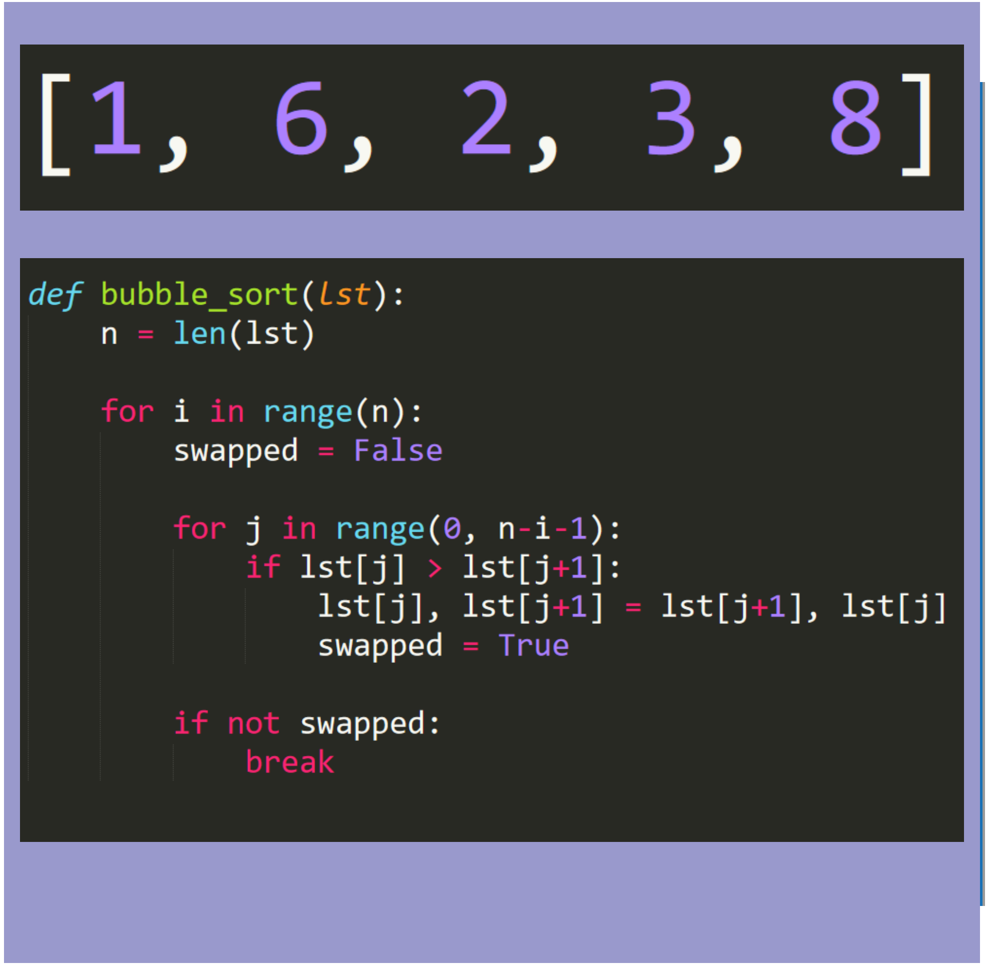


```
[1, 6, 2, 8, 3]
```

```python
def bubble_sort(lst):
    n = len(lst)

    for i in range(n):
        swapped = False

        for j in range(0, n-i-1):
            if lst[j] > lst[j+1]:
                lst[j], lst[j+1] = lst[j+1], lst[j]
                swapped = True

        if not swapped:
            break
```

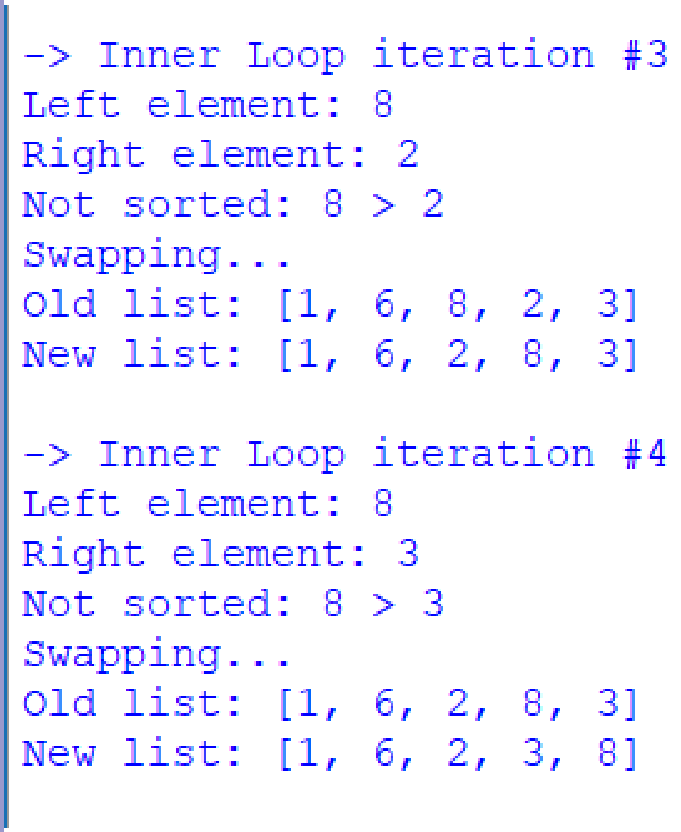


```
-> Inner Loop iteration #3
Left element: 8
Right element: 2
Not sorted: 8 > 2
Swapping...
Old list: [1, 6, 8, 2, 3]
New list: [1, 6, 2, 8, 3]

-> Inner Loop iteration #4
Left element: 8
Right element: 3
Not sorted: 8 > 3
Swapping...
Old list: [1, 6, 2, 8, 3]
New list: [1, 6, 2, 3, 8]
```

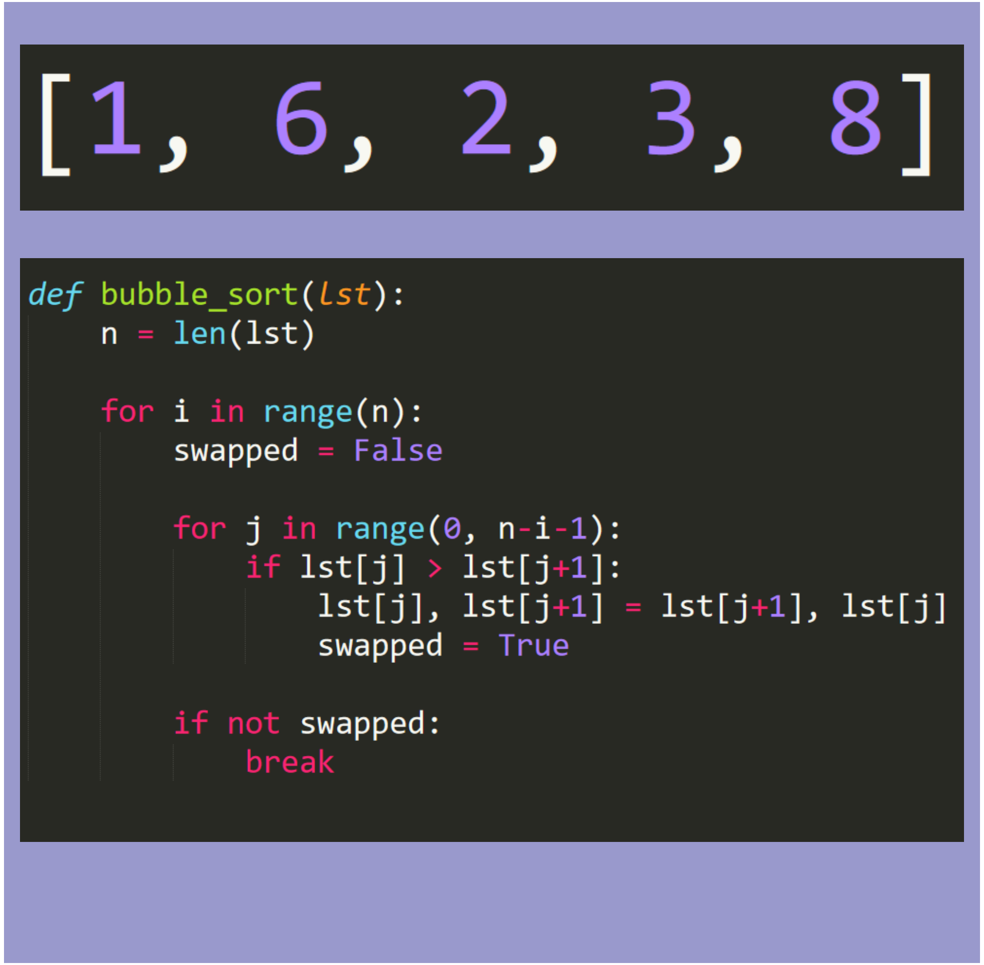


```
[1, 6, 2, 3, 8]
```

```python
def bubble_sort(lst):
    n = len(lst)

    for i in range(n):
        swapped = False

        for j in range(0, n-i-1):
            if lst[j] > lst[j+1]:
                lst[j], lst[j+1] = lst[j+1], lst[j]
                swapped = True

        if not swapped:
            break
```



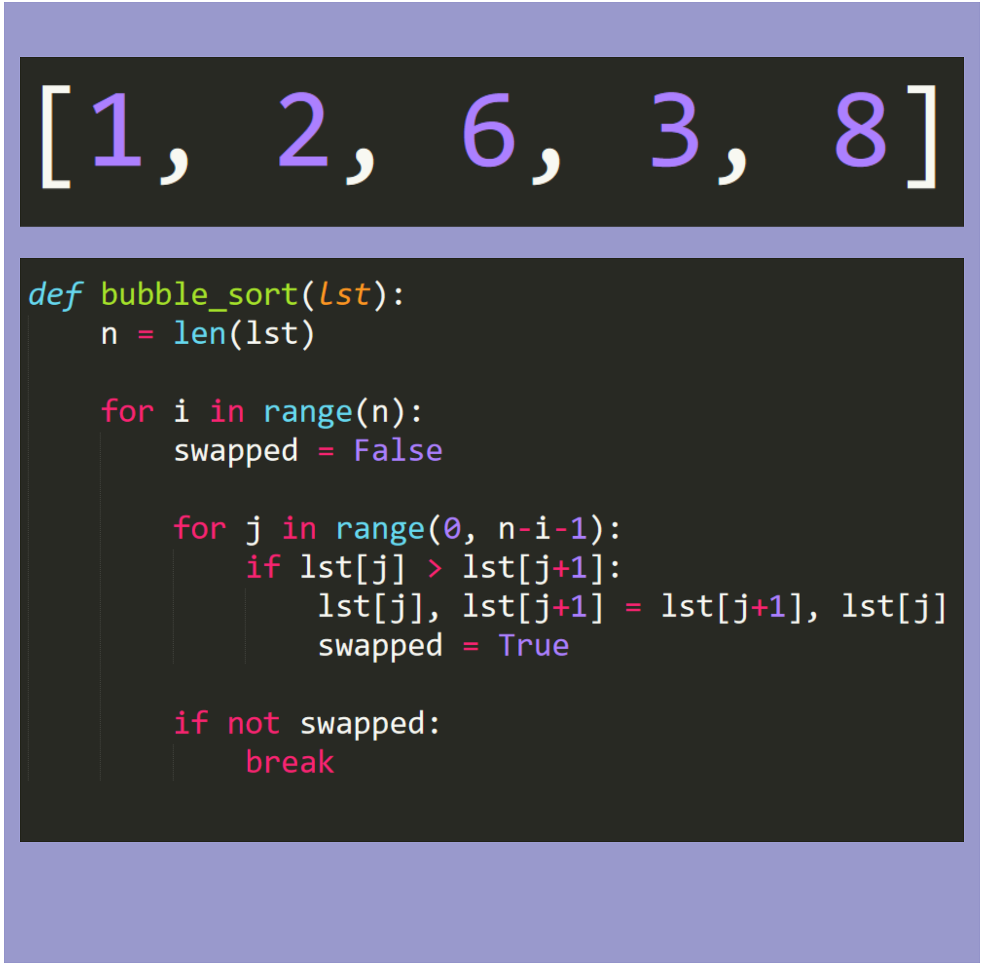
```
-> Inner Loop iteration #3
Left element: 8
Right element: 2
Not sorted: 8 > 2
Swapping...
Old list: [1, 6, 8, 2, 3]
New list: [1, 6, 2, 8, 3]

-> Inner Loop iteration #4
Left element: 8
Right element: 3
Not sorted: 8 > 3
Swapping...
Old list: [1, 6, 2, 8, 3]
New list: [1, 6, 2, 3, 8]
```



```
[1, 6, 2, 3, 8]
```

```python
def bubble_sort(lst):
    n = len(lst)

    for i in range(n):
        swapped = False

        for j in range(0, n-i-1):
            if lst[j] > lst[j+1]:
                lst[j], lst[j+1] = lst[j+1], lst[j]
                swapped = True

        if not swapped:
            break
```

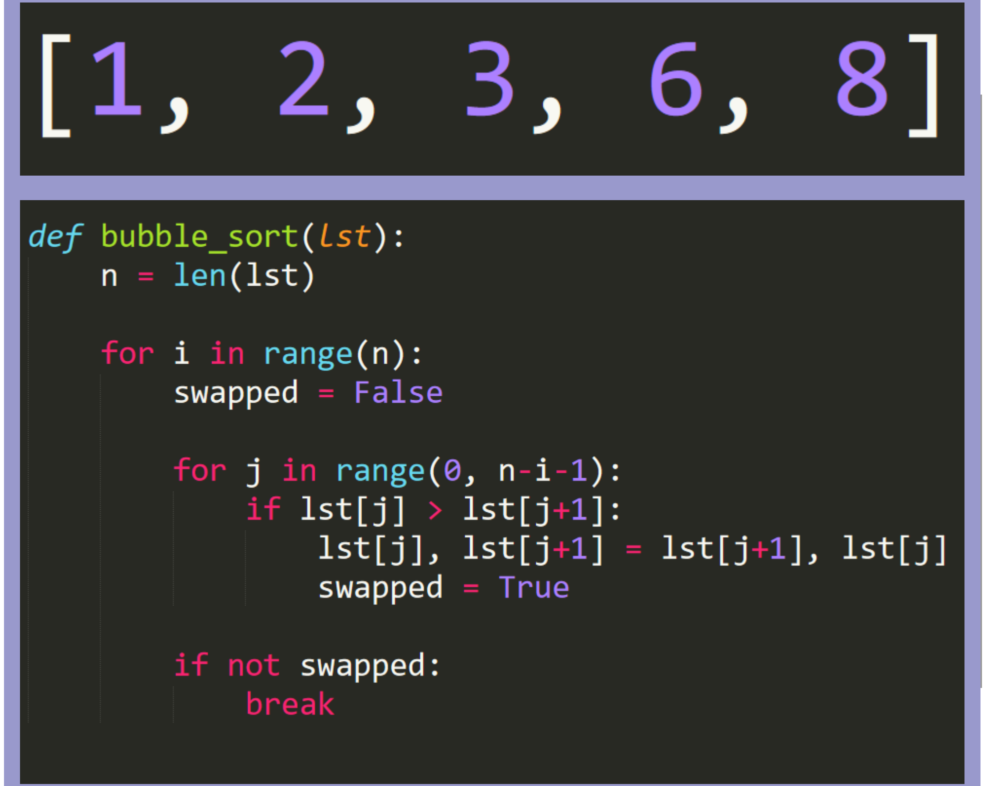
```
-----> Outer Loop iteration #2

-> Inner Loop iteration #1
Left element: 1
Right element: 6
Already sorted: 1 < 6
No change: [1, 6, 2, 3, 8]

-> Inner Loop iteration #2
Left element: 6
Right element: 2
Not sorted: 6 > 2
Swapping...
Old list: [1, 6, 2, 3, 8]
New list: [1, 2, 6, 3, 8]

-> Inner Loop iteration #3
Left element: 6
Right element: 3
Not sorted: 6 > 3
Swapping...
Old list: [1, 2, 6, 3, 8]
New list: [1, 2, 3, 6, 8]
```



```
[1, 2, 6, 3, 8]
```

```python
def bubble_sort(lst):
    n = len(lst)

    for i in range(n):
        swapped = False

        for j in range(0, n-i-1):
            if lst[j] > lst[j+1]:
                lst[j], lst[j+1] = lst[j+1], lst[j]
                swapped = True

        if not swapped:
            break
```

```
-----> Outer Loop iteration #2

-> Inner Loop iteration #1
Left element: 1
Right element: 6
Already sorted: 1 < 6
No change: [1, 6, 2, 3, 8]

-> Inner Loop iteration #2
Left element: 6
Right element: 2
Not sorted: 6 > 2
Swapping...
Old list: [1, 6, 2, 3, 8]
New list: [1, 2, 6, 3, 8]

-> Inner Loop iteration #3
Left element: 6
Right element: 3
Not sorted: 6 > 3
Swapping...
Old list: [1, 2, 6, 3, 8]
New list: [1, 2, 3, 6, 8]
```



```
[1, 2, 3, 6, 8]
```

```python
def bubble_sort(lst):
    n = len(lst)

    for i in range(n):
        swapped = False

        for j in range(0, n-i-1):
            if lst[j] > lst[j+1]:
                lst[j], lst[j+1] = lst[j+1], lst[j]
                swapped = True

        if not swapped:
            break
```

```
-----> Outer Loop iteration #3

-> Inner Loop iteration #1
Left element: 1
Right element: 2
Already sorted: 1 < 2
No change: [1, 2, 3, 6, 8]

-> Inner Loop iteration #2
Left element: 2
Right element: 3
Already sorted: 2 < 3
No change: [1, 2, 3, 6, 8]

There was no need to swap! The list is now sorted
[1, 2, 3, 6, 8]
```

Time to Practice!