Bubble Sort

Bubble Sort Algorithm

The Bubble Sort algorithm is a simple algorithm to sort a list of N numbers in ascending order. Bubble sort works by iterating through a list and checking whether the current element is larger or smaller than the next element. This algorithm consists of an outer iteration and an inner iteration. In the inner iteration, the first and second elements are first compared and swapped so that the second element has a higher value than the first. This is repeated for the subsequent second and third element pairs and so forth until the last pair of (N-2, N-1) elements is compared. At the end of the inner iteration, the largest element appears last. This is repeated for all elements of the list in the outer iteration.

3 5 4 6 2 1

Bubble Sort Big-O Runtime

The Bubble Sort algorithm utilizes two loops: an outer loop to iterate over each element in the input list, and an inner loop to iterate, compare and exchange a pair of values in the list. The inner loop takes (N-1) iterations while the outer loop takes N iterations. Hence, the Big-O runtime for the algorithm is the product of $O(N)$ and $O(N-1)$, which is $O(N^2)$.
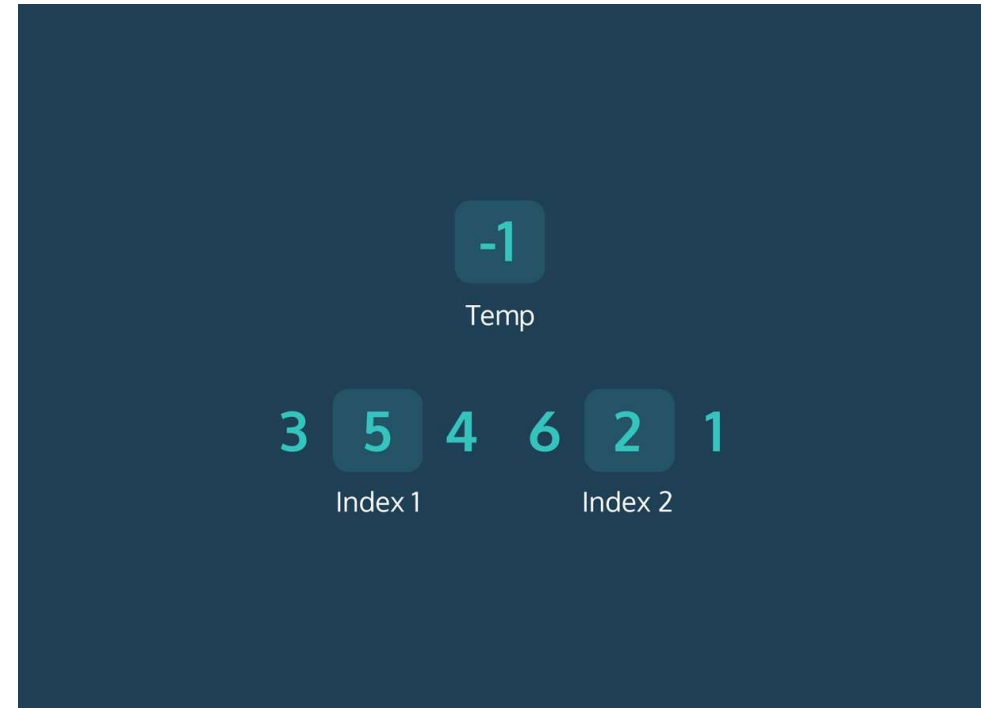
Bubble Sort Swapping Variables

The Bubble Sort algorithm requires swapping of variables in order to sort them.
The swapping algorithm is dependent on the programming language. For most
languages, a temporary variable is needed to hold one of the values being
swapped:

```
temp_variable = number_1
number_1 = number_2
number_2 = temp_variable
```

For others, the swapping can be done in a single assignment:

```
number_1, number_2 = number_2, number_1
```


Swapping Variables in Bubble Sort

In the Bubble Sort algorithm, the swap function that swaps two elements in a
 list can be called in a Bubble Sort function to iteratively swap an element with
its adjacent neighbor whose value is smaller until all the elements are sorted in
ascending order.

```python
def swap(arr, left_pos, right_pos):
    temp = arr[left_pos]
    arr[left_pos] = arr[right_pos]
    arr[right_pos] = temp


def bubble_sort(arr):
    for itm in arr:
        for idx in range(len(arr) - 1):
            if arr[idx] > arr[idx + 1]:
```

Python Swap Function

A Python function that swaps two adjacent values in a list can be written as follows:

```python
def swap(arr, pos_1, pos_2):
  tmp = arr[pos_1]
  arr[pos_1] = arr[pos_2]
  arr[pos_2] = tmp
```