Computer Science
UNIVERSITY OF TORONTO

Bubble Sort

Let's look at an example:

Example

We'll start at index `i` and work towards end of list `e`

Pass 1

```
  i           e
| 7 | 3 | 5 | 2 |
```

Compare the item at index `i` with the item at `i + 1`. Since `7` is greater than `3`, they are swapped.

```
      i       e
| 3 | 7 | 5 | 2 |
```

Increase `i` by 1. Compare the item at index `i` with the item at `i + 1`. Since `7` is greater than `5`, they are swapped.

```
          i   e
| 3 | 5 | 7 | 2 |
```

Increase `i` by 1. Compare the item at index `i` with the item at `i + 1`. Since `7` is greater than `2`, they are swapped.

```
              e
| 3 | 5 | 2 | 7 |
```

This pass is complete. The largest item in the list, `7`, is in its correct location.

Pass 2

```
  i       e |
| 3 | 5 | 2 |   7   |
unsorted    | sorted
```

`i` initially refers to `0`. Compare the item at index `i` with the item at `i + 1`. Since `3` is less than `5`, nothing needs to be swapped.

```
      i   e |
| 3 | 5 | 2 |   7   |
unsorted    | sorted
```

Increase `i` by 1. Compare the item at index `i` with the item at `i + 1`. Since `5` is greater than `2`, they are swapped.

```
          e |
| 3 | 2 | 5 |   7   |
unsorted    | sorted
```

This pass is complete. The second largest item in the list, `5`, is in its correct location.

Pass 3

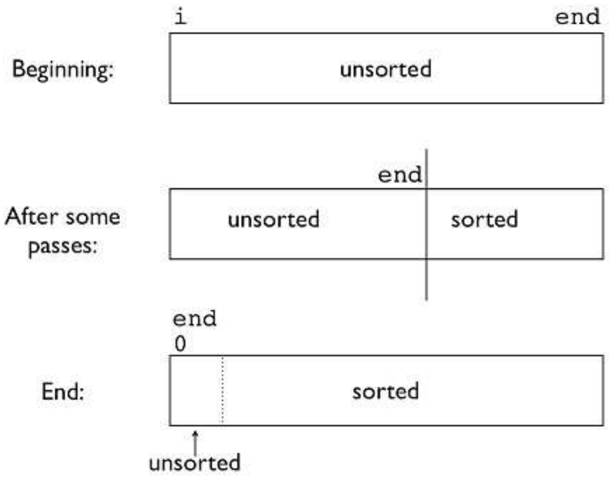
```
     i    e |
   | 3 | 2 | 5 | 7 |
unsorted    | sorted
```

`i` initially refers to `0`. Compare the item at index `i` with the item at `i + 1`. Since `3` is greater than `2`, they are swapped.

```
  | 2 | 3 | 5 | 7 |
```

This pass is complete. The list is now sorted!

Generalized List States



Implementation

```
def bubble_sort(L):
    """ (list) -> NoneType

    Sort the items of L from smallest to largest.

    >>> L = [7, 3, 5, 2]
    >>> bubble_sort(L)
    >>> L
    [2, 3, 5, 7]
    """

    # The index of the last unsorted item.
    end = len(L) - 1

    while end != 0:

        # Bubble once through the unsorted section to move the largest item
        # to index end.
        for i in range(end):
            if L[i] > L[i + 1]:
                L[i], L[i + 1] = L[i + 1], L[i]

        end = end - 1


if __name__ == '__main__':
    import doctest
    doctest.testmod()
```

Jennifer Campbell • Paul Gries
University of Toronto