

BUBBLE SORT

CS302 – Introduction to Programming
University of Wisconsin – Madison
Lecture 12

By Matthew Bernstein – matthewb@cs.wisc.edu

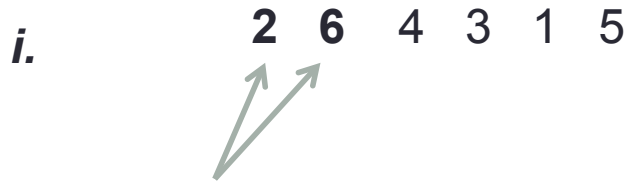
Bubble Sort

- The idea of bubble sort is that we iterate through our array repeatedly. For each iteration, every time we see a pair of elements that are out of order (i.e. a_2 precedes a_1 when $a_1 < a_2$), then we swap the two elements. If we ever iterate through the array and we never have to swap, this means that the array is in order and we can terminate the algorithm.

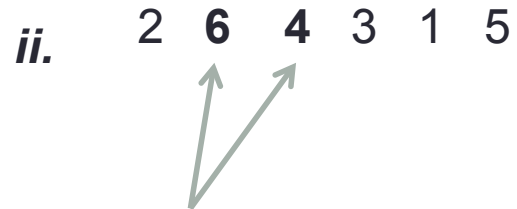
Suppose we have the following array:

2 6 4 3 1 5

1st Iteration



Look at this pair of elements.
Are the two elements in order?
Yes. Because 2 precedes 6.
So we do not swap them



Look at this pair of elements.
Are the two elements in order?
No. Because 6 precedes 4.
So we DO swap them.




Look at this pair of elements.
Are the two elements in order?
No. Because 6 precedes 3.
So we DO swap them.



Look at this pair of elements.
Are the two elements in order?
No. Because 6 precedes 1.
So we DO swap them.

1st Iteration Continued

v. 2 4 3 1 **6** **5**



Look at this pair of elements.
Are the two elements in order?
No. Because 6 precedes 5.
So we DO swap them

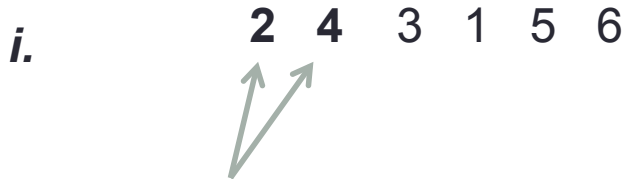
vi. 2 4 3 1 5 6

This is the array we are left with after the first iteration.
Now we ask ourselves:
During this iteration, did we perform any swaps?

Answer:

Yes, we performed many swaps. This means we go on to the next iteration

2nd Iteration



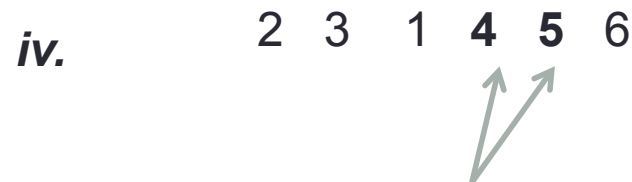
Look at this pair of elements.
Are the two elements in order?
Yes. We don't swap them.



Look at this pair of elements.
Are the two elements in order?
No.. So we swap them.




Look at this pair of elements.
Are the two elements in order?
No. So we swap them.



Look at this pair of elements.
Are the two elements in order?
Yes. We don't swap them.

2nd Iteration Continued

v. 2 3 1 4 **5 6**



Look at this pair of elements.
Are the two elements in order?
Yes. We don't swap them.

vi. 2 3 1 4 5 6

This is the array we are left with after the second iteration.

Did we perform any swaps during this iteration?

Yes, so we go on to another iteration.

3rd Iteration

i. 2 3 1 4 5 6

ii. 2 3 1 4 5 6

 Swap 3 and 1

iii. 2 1 3 4 5 6

iv. 2 1 3 4 5 6

v. 2 1 3 4 5 6

vi. 2 1 3 4 5 6

4th Iteration

i. 2 1 3 4 5 6

 Swap 2 and 1

ii. 1 2 3 4 5 6

iii. 1 2 3 4 5 6

iv. 1 2 3 4 5 6

v. 1 2 3 4 5 6

vi. 1 2 3 4 5 6

5th Iteration

i. 1 2 3 4 5 6

ii. 1 2 3 4 5 6

iii. 1 2 3 4 5 6

iv. 1 2 3 4 5 6

v. 1 2 3 4 5 6

vi. 1 2 3 4 5 6

During the 5th iteration, we performed no swaps

So the algorithm terminates. We're finished and we see that the array is now ordered.

Pseudocode for Bubble Sort

```
procedure bubbleSort( A : list of sortable items )  
  
    repeat  
        swapped = false  
        for i = 1 to length(A) - 1 inclusive do:  
            if A[i-1] > A[i] then  
                swap( A[i-1], A[i] )  
                swapped = true  
            end if  
        end for  
    until not swapped  
  
end procedure
```

Cool CS Link of the Day

- Website for Android Development
- <http://developer.android.com/index.html>

