**The Bubble Sort**

**Swapping Values**

Before we get into the sort algorithms, we need to learn how to swap two values in a list. Imagine we have two integer variables:

x = 7

y = 3

print (x,y)

Now imagine that we wish to swap the values. Would this work?

x = 7

y = 3

print (x,y)

x = y # make x, y

y = x # make y, x

print (x,y) # is x and y reversed?

The answer is “no”. All this does is make them both the same value (y’s value). To swap the values, we need a third temporary variable. Replace the last three lines of code with these four lines:

temp = x # temp gets x value

x = y # x gets y value

y = temp # y gets temp value (which is x old value)

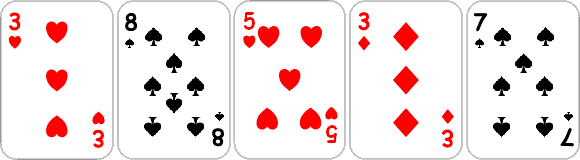
print (x,y) # is x and y reversed?

Now the answer is “yes”. This creates a “daisy chain” which allows the values to be swapped.

**Sort Algorithms: The Bubble Sort**

Imagine I give you a set of cards and I ask you to sort them in order from ace to king. You could do this easily, but describing how to do it is not so easy. One simple sorting method is called the **bubble** **sort.** This is how it works (note this is not a program):

Suppose we begin with five cards unsorted in a row:



1. Beginning with the first card:
2. Compare the card with the card next to it. If the card on the left is bigger, switch the cards.
3. Go to the next card.
4. Repeat steps 2 and 3 three times (four times in total).
5. Repeat steps 1 to 4 until no cards get switched (i.e. they are in order).

Done!

Notice this algorithm has **two loops**. The **inner loop** is steps 2 and 3. This is done 4 times. The **outer loop** is steps 1 to 4. They are repeated until no cards are moved.

The inner loop switches pairs of cards until we get to the end of the line. Notice that if we have 5 cards, we only make 4 switches.

The outer loop goes back to the first card and does the process over again.

This animation shows what this looks like:

<https://www.youtube.com/watch?v=EuZvwpcpMdo>

Now that we have a sort algorithm, we try to program this in Python.

First let’s start with a list of 5 numbers:

numbers = [3,2,7,1,4]

Now we apply the algorithm. We build the inner loop:

for i in range (4):

if numbers[0] > numbers[1]: # if numbers are in wrong order

# switch the numbers

This is where we add your switch code that we wrote in the first part of this lesson.

This code isn’t quite right, since it will only compare the first two values. How do you get it to say “compare this guy with the guy next to it?”

We want to repeat everything over and over until no switches are made. You’ll need a while loop and a flag to do this.

Exercises

1. Examine the data file named sort1.txt. Create a program that reads the values and sorts them. Print the sorted values on the screen.
2. Examine the data file named sort2.txt. Create a program that reads the values and sorts them. Print the sorted values on the screen. Hint: letters are really numbers. Hint #2: Your code is almost exactly the same as for exercise 1.
3. Examine the data file named sort3.txt. Create a program that reads the values and sorts them by last name. Print the sorted values on the screen.
4. Examine the data file named sort4.txt. Create a program that reads the values and sorts them by score. Print the sorted values on the screen.