3.1 Sorting

Ordering the elements of a list is a problem that occurs in many contexts.

Sorting is putting elements into a list in which the elements are in increasing (or decreasing) order.

Example 1:
Given a list {1, 5, 2, 7, 3, 4}, the sorted list will be {1, 2, 3, 4, 5, 7}

Given a list {a, g, s, d, f, p} the sorted list will be {a, d, f, g, p, s}

There are many sorting algorithms. Some algorithms are easy to implements, some more efficient, some take advantage of particular computer architecture, and so on.

Some of the names (we will discuss first two):
Bubble sort
Insertion sort
Merge sort
Selection sort
Quicksort

3.1 Bubble sort

Let's consider Bubble sort.
It is a simplest one, but not an efficient algorithm

Idea: compares adjacent elements and interchanges them if necessary

procedure *bubblesort*(a_1,...,a_n:real numbers with $n \geq 2$)
for $i := 1$ **to** $n-1$
    for $j := 1$ **to** $n-i$
        if $a_j > a_{j+1}$ **then** interchange a_j and a_{j+1}
{$a_1, a_2, ..., a_n$ is in increasing order}

Summary: the bubble sort is done in n-1 passes.
During *each pass* we start at the beginning of the list and compare first and second elements: if the first element is larger than the second – we interchange them, and do nothing otherwise. Then we compare the second and the third elements (and interchange them if the second element is larger than the third one). And so on – till we reach the end of the list.

Example: Let's see the work of the Bubble sort on the list {3, 1, 7, 5, 0}

First pass (i=1):

a_5	0	0	0	0	7
a_4	5	5	5	7	0
a_3	7	7	7	5	5
a_2	1	3	3	3	3
a_1	3	1	1	1	1
	j=1	j=2	j=3	j=4=n-i	
	$a_1 > a_2$?	$a_2 > a_3$?	$a_3 > a_4$?	$a_3 > a_4$?	
	3 > 1 ?	3 > 7 ?	7 > 5 ?	7 > 5 ?	

Second pass (i=2):

a_5	7	7	7	7
a_4	0	0	0	5
a_3	5	5	5	0
a_2	3	3	3	3
a_1	1	1	1	1
	j=1	j=2	j=3=n-i	
	$a_1 > a_2$?	$a_2 > a_3$?	$a_3 > a_4$?	
	1 > 3 ?	3 > 5 ?	5 > 0 ?	

3.1 Bubble sort

Example continued: Bubble sort on list {3, 1, 7, 5, 0}

Third pass (i=3):

a_5	7	7	7
a_4	5	5	5
a_3	0	0	3
a_2	3	3	0
a_1	1	1	1
	j=1	j=2=n-i	
	$a_1 > a_2$?	$a_2 > a_3$?	
	1 > 3 ?	3 > 0 ?	

Fourth pass (i=4):

a_5	7	7	7
a_4	5	5	5
a_3	3	3	3
a_2	0	1	1
a_1	1	0	0
	j=1=n-i		
	$a_1 > a_2$?	Stop	
	1 > 0 ?		

procedure *bubblesort*(a_1,...,a_n:real numbers with $n \geq 2$)
for *i := 1* **to** *n-1*
  for *j := 1* **to** *n-i*
    if $a_j > a_{j+1}$ **then** interchange a_j and a_{j+1}
{a_1, a_2, ..., a_n is in increasing order}

3.1 Bubble sort

procedure *bubblesort*(a_1,...,a_n:real numbers with $n \geq 2$)
for *i := 1* **to** *n-1*
    for *j := 1* **to** *n-i*
        if $a_j > a_{j+1}$ **then** interchange a_j and a_{j+1}
{a_1, a_2, ..., a_n is in increasing order}

How many iterations (comparisons) are performed on an n-element list?

for i=1 n-1
for i=2 n-2
for i=3 n-3
....
for i=n-1 n-(n-1)

Therefore we have the following sum:

(n-1) + (n-2) + (n-3) + (n-4) + ... + (n-(n-1)) = (1+2+3+4+...(n-1))=

i=1 i=2 i=3 i=4 i=n-1

n(n-1)/2 = $n^2/2 - n/2$ (quadratic)

3.1 Insertion sort

Insertion sort is a simple algorithm, but still not usually efficient

procedure *insertionsort* ($a_1,...,a_n$:real numbers with $n \geq 2$)

for $j := 2$ **to** n

    $i := 1$

    while $a_j > a_i$

        $i := i+1$

    $m := a_j$

    for $k := 0$ **to** $j-i-1$

        $a_{j-k} := a_{j-k-1}$

    $a_i := m$

{$a_1, a_2, ..., a_n$ is in increasing order}

Summary:

- There are n-1 passes (passes 2,3, …,n).
- On the start of pass j, the first j-1 elements are in order.
- The correct position (i) for element j is found among the first j-1 elements.
- The elements in positions i to j-1 are shifted up one.
- Element j is inserted at position i.

Example 3:
Let's see how insertion sort works on the list {7, 0, 3, 2, 6}

j=2:
i=1 7, 0, 3, 2, 6 → 7, 0, 3, 2, 6 → 7, 7, 3, 2, 6 → 0, 7, 3, 2, 6
m=0 $a_2=a_1$ $a_1=m=0$

j=3:
i=1 0, 7, 3, 2, 6 → 0, 7, 3, 2, 6 → 0, 7, 7, 2, 6 → 0, 3, 7, 2, 6
i=2 m=3 $a_3=a_2$ $a_2=m=3$

j=4:
2nd 1st
i=1 0, 3, 7, 2, 6 → 0, 3, 7, 2, 6 → 0, 3, 3, 7, 6 → 0, 2, 3, 7, 6
i=2 m=2 $a_4=a_3, a_3=a_2$ $a_2=m=2$

j=5:
i=1 0, 2, 3, 7, 6 → 0, 2, 3, 7, 6 → 0, 2, 3, 7, 7 → 0, 2, 3, 6, 7
i=4 m=6 $a_5=a_4$ $a_4=m=6$

Problems from book (Section 3.1):

35, 37, 39, 45, 47, 48, 49