MAT2440, Classwork29, Spring2025

ID:______________________ Name:______________________

1. Algorithms of **Sorting Problems.**

   Problem: Given a sequence. Then sort the elements in ascending (or descending) order.

   Algorithms: (1) **The Bubble Sort**. (2) **The Insertion Sort**.

   For example: input is $\{3, 2, 4, 1, 5\}$ and the output of ascending order is $\{1, 2, 3, 4, 5\}$.

2. Sorting algorithm I: Algorithm and Pseudocode of the **Bubble Sort**.

   The smaller elements "bubble" to the top as they are interchanged with larger elements. Here we use an example to explain the algorithm:

compare a_1, a_2

Given a list $\{3, 2, 4, 1, 5\}$ *and get this list in ascending order* $\{1, 2, 3, 4, 5\}$.

a_j, a_{j+1}

Algorithm:

First round $i=1$

	$j=1$	$j=2$	$j=3$	$j=4$
3	2	2	2	2
2	3	3	3	3
5	5	5	1	1
1	1	1	5	4
4	4	4	4	$\boxed{5}$

Second round $i=2$

	$j=1$	$j=2$	$j=3$
2	2	2	2
3	3	1	1
1	1	3	3
4	4	4	$\boxed{4}$
5	5	5	$\boxed{5}$

Third round $i=3$

	$j=1$	$j=2$
2	1	1
1	2	2
3	3	3
4	4	4
5	5	5

Fourth round $i=4$

	$j=1$
1	1
2	2
3	3
4	4
5	5

length of the list

where $i = 1$ to $\underline{5-1}$: how many rounds you have to go through, and

$j = 1$ to $\underline{5-i}$: in each round, how many times you need to check and swap.

Pseudocode:

procedure *bubblesort*($a_1, a_2, \cdots, a_n$: real numbers with $n \geq 2$)
$n :=$ the length of $\{a_i\}$
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}
$\{a_1, a_2, \cdots, a_n$ is in increasing order$\}$

3. Sorting algorithm II: Algorithm and Pseudocode of the **Insertion Sort**.

Insertion sort compares the i^{th} element to its previous $i-1$ sorted elements and <u>insert</u> the i^{th} element in the right location among the previous $i-1$ sorted ones.

Here we use the same example to explain the algorithm:

Given a list $\{3, 2, 5, 1, 4\}$ *and get this list in ascending order* $\{1, 2, 3, 4, 5\}$.

Algorithm:

First Round: $i=2$ 2^{nd} element "<u>2</u>", $3, 2, 5, 1, 4 \Rightarrow \boxed{2\,3}\,5\,1\,4$

Second Round: $i=3$ 3^{rd} element "<u>5</u>", $2, 3, 5, 1, 4 \Rightarrow \boxed{2\,3\,5}\,1\,4$

Third Round: $i=4$ 4^{th} element "<u>1</u>", $2, 3, 5, 1, 4 \Rightarrow \boxed{1\,2\,3\,5}\,4$

Fourth Round: $i=5$ 5^{th} element "<u>4</u>", $1, 2, 3, 5, 4 \Rightarrow 1\,2\,3\,4\,5$

In "$\Rightarrow$" parts, we do (1) compare a_i to a_j where $1 \le j \le$ <u>i</u> until a_i <u>$<$</u> a_j, then (2) insert a_i to the position of a_j.

Pseudocode:

procedure *insertionsort*$(a_1, a_2, \cdots, a_n$: real numbers with $n \ge 2)$
$n :=$ the <u>length</u> of $\{a_i\}$
for $i :=$ <u>2</u> **to** <u>n</u> ← pick i^{th} element
    $j :=$ <u>1</u>
    while $(a_i > a_j$ and $i > j)$
        $j := j + 1$
    } search for index j
    $m := a_i$
    for $k :=$ <u>0</u> **to** <u>$i-j-1$</u>
        $a_{i-k} := a_{i-k-1}$
    $a_j := m$
    } move $(a_j, \cdots, a_{i-1})$ to $(a_{j+1}, \cdots, a_i)$
    insert a_i to the position a_j
$\{a_1, a_2, \cdots, a_n$ is in increasing order$\}$