Adam Mukharil Bachtiar
English Class
Informatics Engineering 2011

Algorithms and Programming

Sorting

Steps of the Day

Let’s Start

Definition of Sorting

What is Sorting

Process that **arranges random data** into **sorted data**. Data can be sorted into **ascending** or **descending**.

Algorithms of Sorting

- Bubble Sort
- Selection Sort

Bubble Sort

Definition and Structures of Bubble Sort

What is Bubble Sort

- Sorting algorithm which was inspired by **bubble soap**.
- Comparing **element of array (i)** with **next element of it (i+1)**.
- If i is **bigger than** i+1 then **swap value** of each element.

Illustration of Bubble Sort

Array:	5	3	7	9	2	3	6	4	3	1

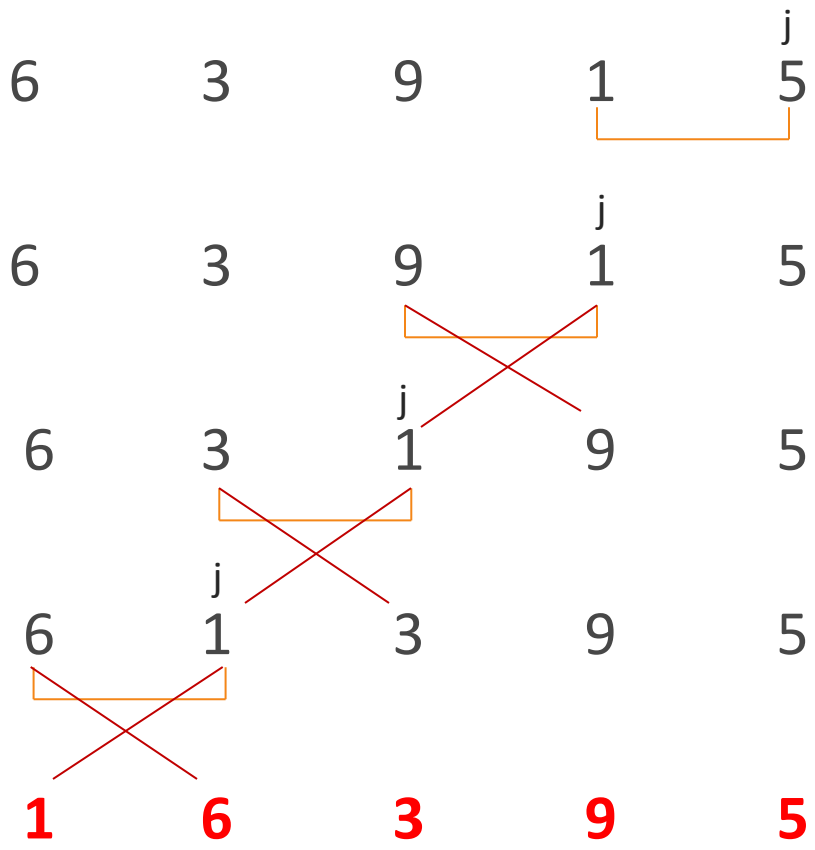
L. 1	3	5	7	2	3	6	4	3	1	9
L. 2	3	5	2	3	6	4	3	1	7	9
L. 3	3	2	3	5	4	3	1	6	7	9
L. 4	2	3	3	4	3	1	5	6	7	9
L. 5	2	3	3	3	1	4	5	6	7	9
L. 6	2	3	3	1	3	4	5	6	7	9
L. 7	2	3	1	3	3	4	5	6	7	9
L. 8	2	1	3	3	3	4	5	6	7	9
L. 9	1	2	3	3	3	4	5	6	7	9

Process of Bubble Sort (Ascending)

This is an array that will be sorted in Ascending way:

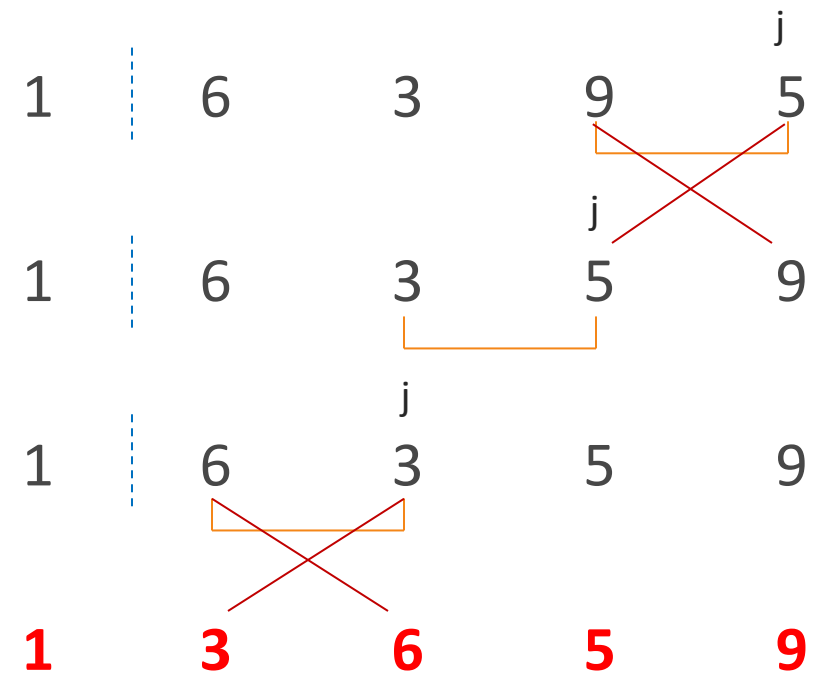
6 3 9 1 5

Step 1 :

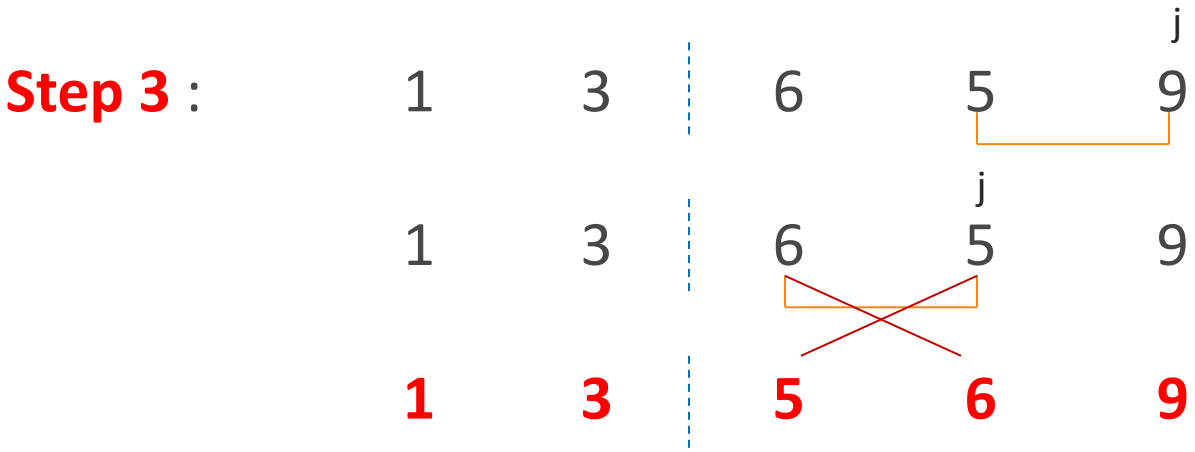


Process of Bubble Sort (Ascending)

Step 2 :



Process of Bubble Sort (Ascending)



Array after sorted in ascending ways:

1 3 5 6 9

General Format for Bubble Sort Ascending

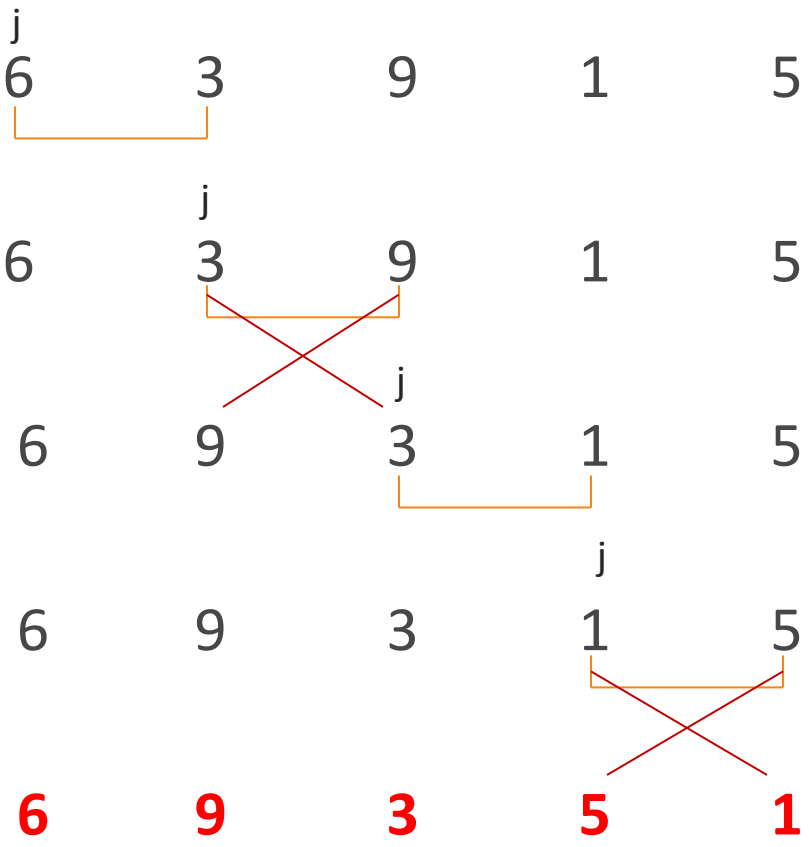
```
1   Procedure BubbleSortAsc(I/O  nama_var_array : nama_tipe_array,
                            Input  N : integer)
2   {I.S. : array[1..N] sudah terdefinisi}
3   {F.S. : menghasilkan array[1..N] yang tersusun secara ascending}
4   Kamus:
5       i, j : integer
6       temp : tipedata
7   Algoritma:
8       for  i  ← 1  to  N-1   do
9         for   j   ← n   downto   i+1  do
10            if(nama_var_array[j] < nama_var_array[j-1])
11            then
12               temp ← nama_var_array[j]
13               nama_var_array[j] ← nama_var_array[j-1]
14               nama_var_array[j-1]  ←  temp
15            endif
16        endfor
17      endfor
18  EndProcedure
```

Process of Bubble Sort (Descending)

This is an array that will be sorted in Descending way :

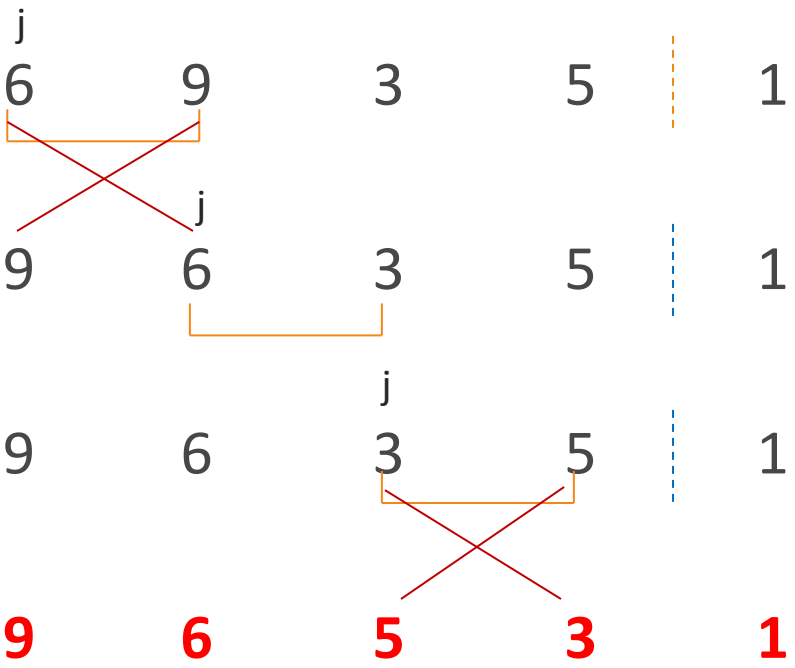
6 3 9 1 5

Step 1 :



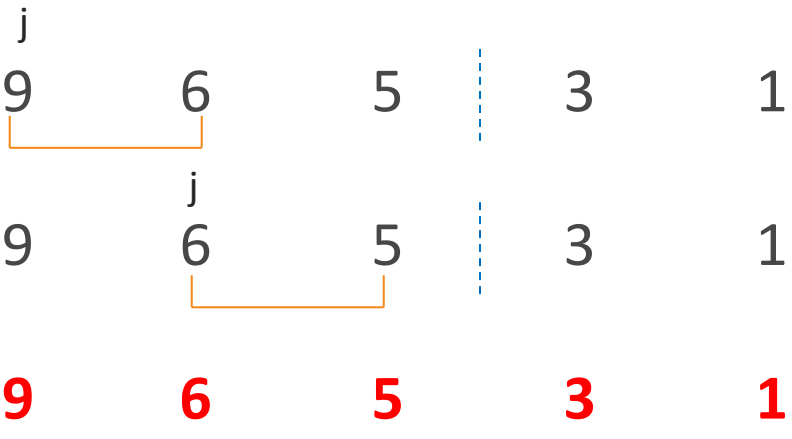
Process of Bubble Sort (Descending)

Step 2 :

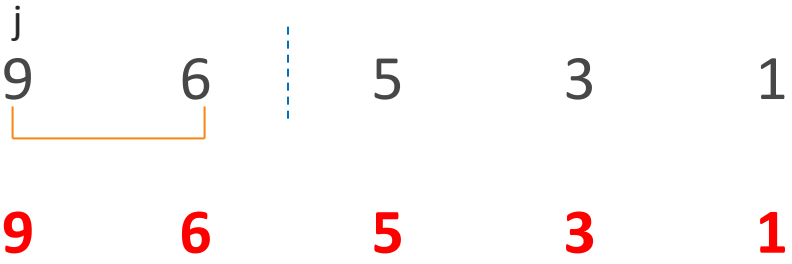


Process of Bubble Sort (Descending)

Step 3 :



Step 4 :



Array after sorted in descending ways:

9 6 5 3 1

General Format for Bubble Sort Descending

```
1   Procedure BubbleSortDesc (I/O nama_var_array : nama_tipe_array,
                              Input  N : integer)
2   {I.S. : array[1..N] sudah terdefinisi}
3   {F.S. : menghasilkan array[1..N] yang tersusun secara descending}
4   Kamus:
5       i,j : integer
6       temp : tipedata
7   Algoritma:
8       for  i  ←  1  to  N-1   do
9           for   j   ←  1   to   (N - i)  do
10            if(nama_var_array[j] < nama_var_array[j+1])
11            then
12              temp ← nama_var_array[j]
13              nama_var_array[j] ← nama_var_array[j+1]
14              nama_var_array[j+1] ← temp
15            endif
16          endfor
17      endfor
18  EndProcedure
```

Selection Sort

Definition and Structures of Selection Sort

What is Selection Sort

Sorting algorithm that arranges random data by **selecting the biggest data** or **the smallest one**.

Methods in Selection Sort

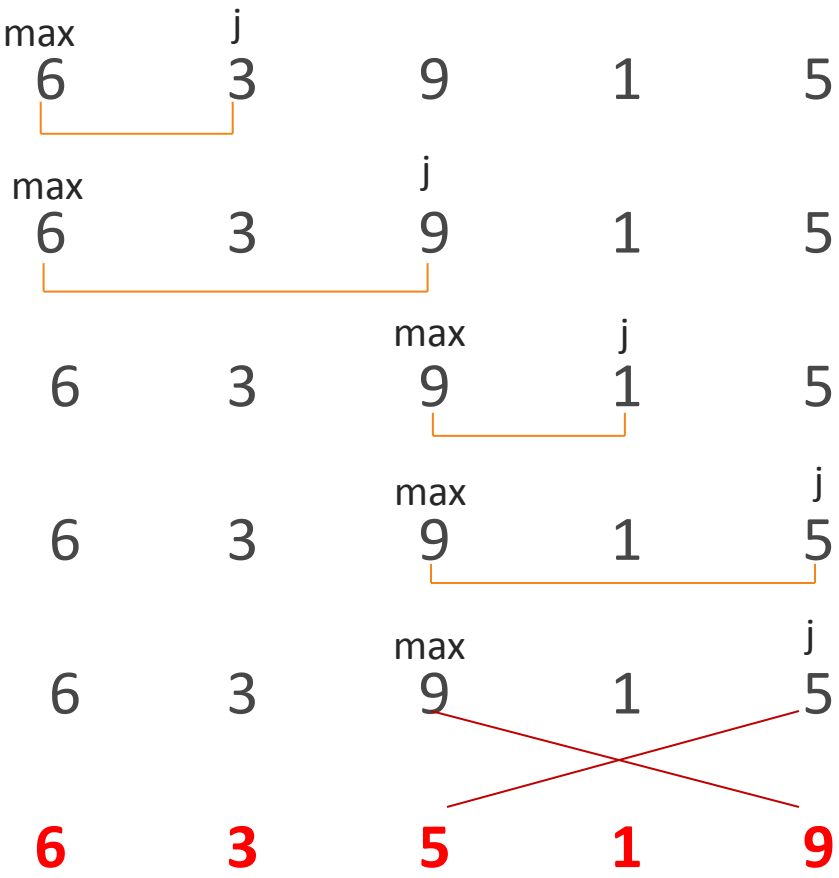
- Maximum Sort
- Minimum Sort

Process of Maximum Sort (Ascending)

This is an array that will be sorted in Ascending way :

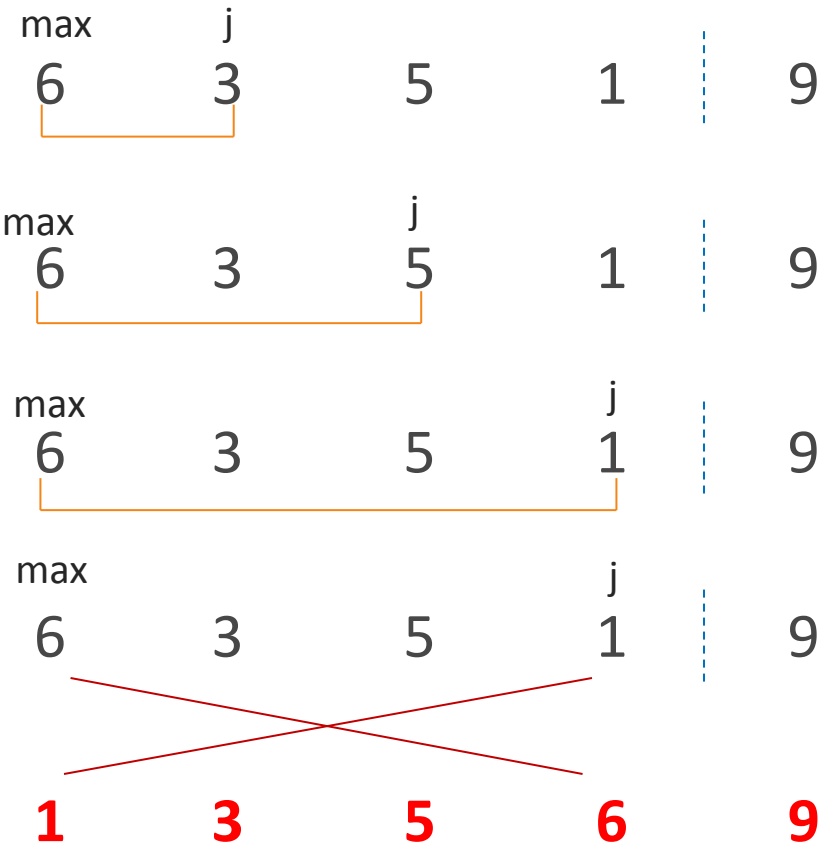
6 3 9 1 5

Step 1 :



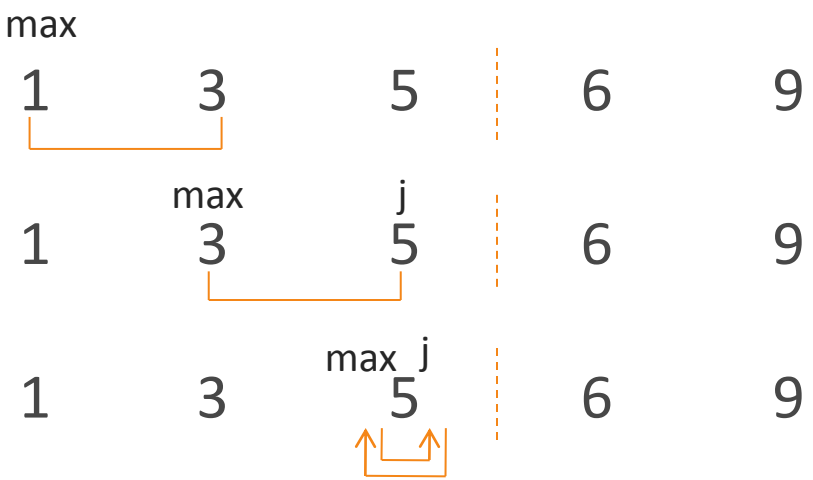
Process of Maximum Sort (Ascending)

Step 2 :



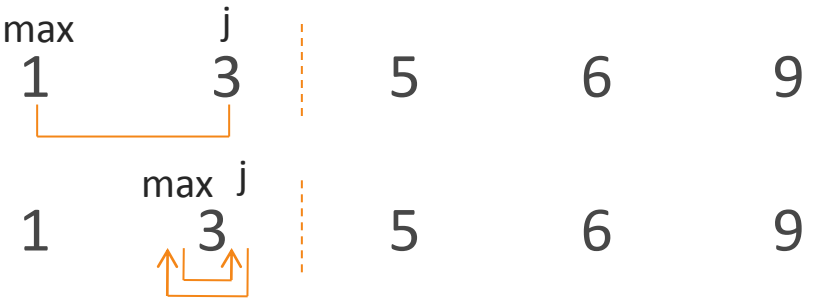
Process of Maximum Sort (Ascending)

Step 3 :



1 3 5 6 9

Step 4 :



Array after sorted in descending way:

1 3 5 6 9

General Format for Maximum Sort Ascending

```
1   Procedure MaximumSortAsc(I/O  nama_var_array : nama_tipe_array,
                             Input N : integer)
2   {I.S. : array [1..N] sudah terdefinisi}
3   {F.S. : menghasilkan array [1..N] yang tersusun secara ascending}
4   Kamus:
5       i, j, max, x : integer
6       temp : tipedata
7   Algoritma:
8       x ← n
9       for i ← 1 to N-1 do
10          max ← 1
11          for j ← 2 to x do
12              if(nama_var_array[j] > nama_var_array[max])
13              then
14                  max ← j
15              endif
16          endfor
17          temp ← nama_var_array[max]
18          nama_var_array[max] ← nama_var_array[j]
19          nama_var_array[j] ← temp
20          x ← x - 1
21      endfor
22  EndProcedure
```

General Format for Minimum Sort Ascending

```
1   Procedure MinimumSortAsc(I/O nama_var_array : nama_tipe_array,
                             Input  N : integer)
2   {I.S. : array[1..n] sudah terdefinisi}
3   {F.S. : menghasilkan array [1..n] yang tersusun secara ascending}
4   Kamus:
5       i, j, min : integer
6       temp  :  tipedata
7   Algoritma:
8       for  i  ←  1  to  (N - 1)   do
9            min ← i
10           for   j   ←  i+1   to   N  do
11              if(nama_var_array[j] < nama_var_array[min])
12              then
13                  min ← j
14              endif
15           endfor
16          temp ← nama_var_array[min]
17          nama_var_array[min] ← nama_var_array[i]
18          nama_var_array[i]  ←  temp
19      endfor
20  EndProcedure
```

THANK YOU

GRACIAS

Contact Person:
Adam Mukharil Bachtiar
Informatics Engineering UNIKOM
Jalan Dipati Ukur Nomor. 112-114 Bandung 40132
Email: adfbipotter@gmail.com
Blog: http://adfbipotter.wordpress.com