**SORTING**

**Algoritma Bubble Sort Per Tahap**

Susun Algoritma (flowchart/pseudocode) tahap 0 skema pengurutan ascending sesuai ilustrasi

procedure BubbleSort(data: array)

n = length(data)

for i from 0 to n-1 do

for j from 0 to n-i-1 do

if data[j] > data[j+1] then

swap(data[j], data[j+1])

end if

end for

end for

end procedure

**Algoritma Bubble Sort**

Susun Algoritma (flowchart) keseluruhan tahap skema pengurutan ascending Bubble Sort sesuai ilustrasi dengan size data n

procedure BubbleSort(data: array)

n = length(data)

for i from 0 to n-1 do

swapped = false

for j from 0 to n-i-1 do

if data[j] > data[j+1] then

swap(data[j], data[j+1])

swapped = true

end if

end for

if swapped = false then

break

end if

end for

end procedure

**Algoritma Selection Sort**

procedure SelectionSort(data: array)

n = length(data)

for i from 0 to n-1 do

minIndex = i

for j from i+1 to n-1 do

if data[j] < data[minIndex] then

minIndex = j

end if

end for

if minIndex != i then

swap(data[minIndex], data[i])

end if

end for

end procedure

**Algoritma Insertion Sort**

Susun Algoritma (flowchart/pseudo code) keseluruhan tahap skema pengurutan ascending Insertion Sort sesuai ilustrasi dengan size data n

procedure InsertionSort(data: array)

n = length(data)

for i from 1 to n-1 do

key = data[i]

j = i - 1

while j >= 0 and data[j] > key do

data[j+1] = data[j]

j = j - 1

end while

data[j+1] = key

end for

end procedure

**Latihan**

Data = {23,35,7,14,67}

Gambarkan proses penyelesaian kasus pengurutan descending data di

atas dengan menggunakan algoritma

**a. Bubble Sort:**

Iterasi 1:

- Membandingkan 23 dan 35 → Tidak perlu pertukaran.

- Membandingkan 35 dan 7 → Pertukaran (7, 35).

- Membandingkan 35 dan 14 → Pertukaran (14, 35).

- Membandingkan 35 dan 67 → Tidak perlu pertukaran.

Iterasi 2:

- Membandingkan 23 dan 7 → Pertukaran (7, 23).

- Membandingkan 23 dan 14 → Pertukaran (14, 23).

- Membandingkan 23 dan 35 → Tidak perlu pertukaran.

Iterasi 3:

- Membandingkan 7 dan 14 → Tidak perlu pertukaran.

- Membandingkan 14 dan 23 → Tidak perlu pertukaran.

Iterasi 4:

- Membandingkan 7 dan 14 → Tidak perlu pertukaran.

Iterasi 5:

- Selesai, array terurut: {67, 35, 23, 14, 7}.

**b. Selection Sort:**

Iterasi 1:

- Memilih 67 sebagai elemen terbesar.

- Pertukaran 67 dengan 7.

Iterasi 2:

- Memilih 35 sebagai elemen terbesar.

- Pertukaran 35 dengan 14.

Iterasi 3:

- Memilih 23 sebagai elemen terbesar.

- Pertukaran 23 dengan 7.

Iterasi 4:

- Memilih 14 sebagai elemen terbesar.

- Pertukaran 14 dengan 7.

Iterasi 5:

- Selesai, array terurut: {67, 35, 23, 14, 7}.

**c. Insertion Sort:**

Iterasi 1:

- Memindahkan 35 ke posisi pertama (35, 23, 7, 14, 67).

Iterasi 2:

- Memindahkan 23 ke posisi kedua (35, 23, 7, 14, 67).

Iterasi 3:

- Memindahkan 7 ke posisi ketiga (35, 23, 7, 14, 67).

Iterasi 4:

- Memindahkan 14 ke posisi keempat (35, 23, 14, 7, 67).

Iterasi 5:

- Selesai, array terurut: {67, 35, 23, 14, 7}.