

LAPORAN HASIL TEORI
MATA KULIAH ALGORITMA DAN STRUKTUR DATA

**PERTEMUAN 6 : BUBBLE SORT, SELECTION SORT, INSERTION
SORT**



KAYLA RACHMAUDINA SATITI PUTRI

2341760103

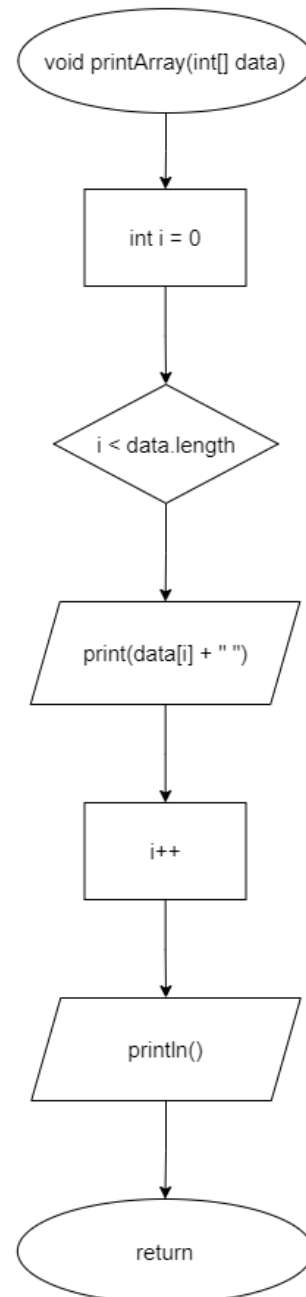
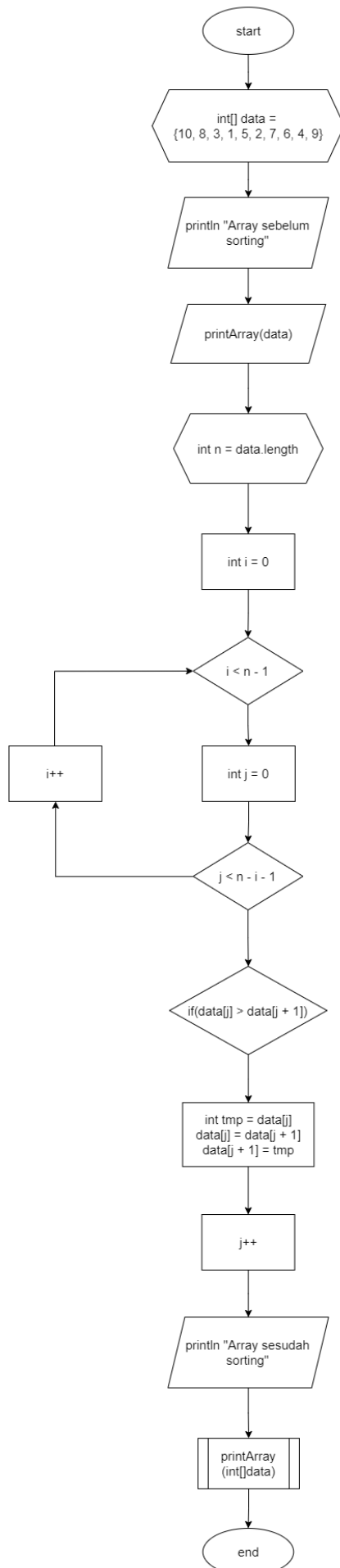
D-IV SISTEM INFORMASI BISNIS

**JURUSAN TEKNOLOGI INFORMASI POLITEKNIK
NEGERI MALANG**

2024

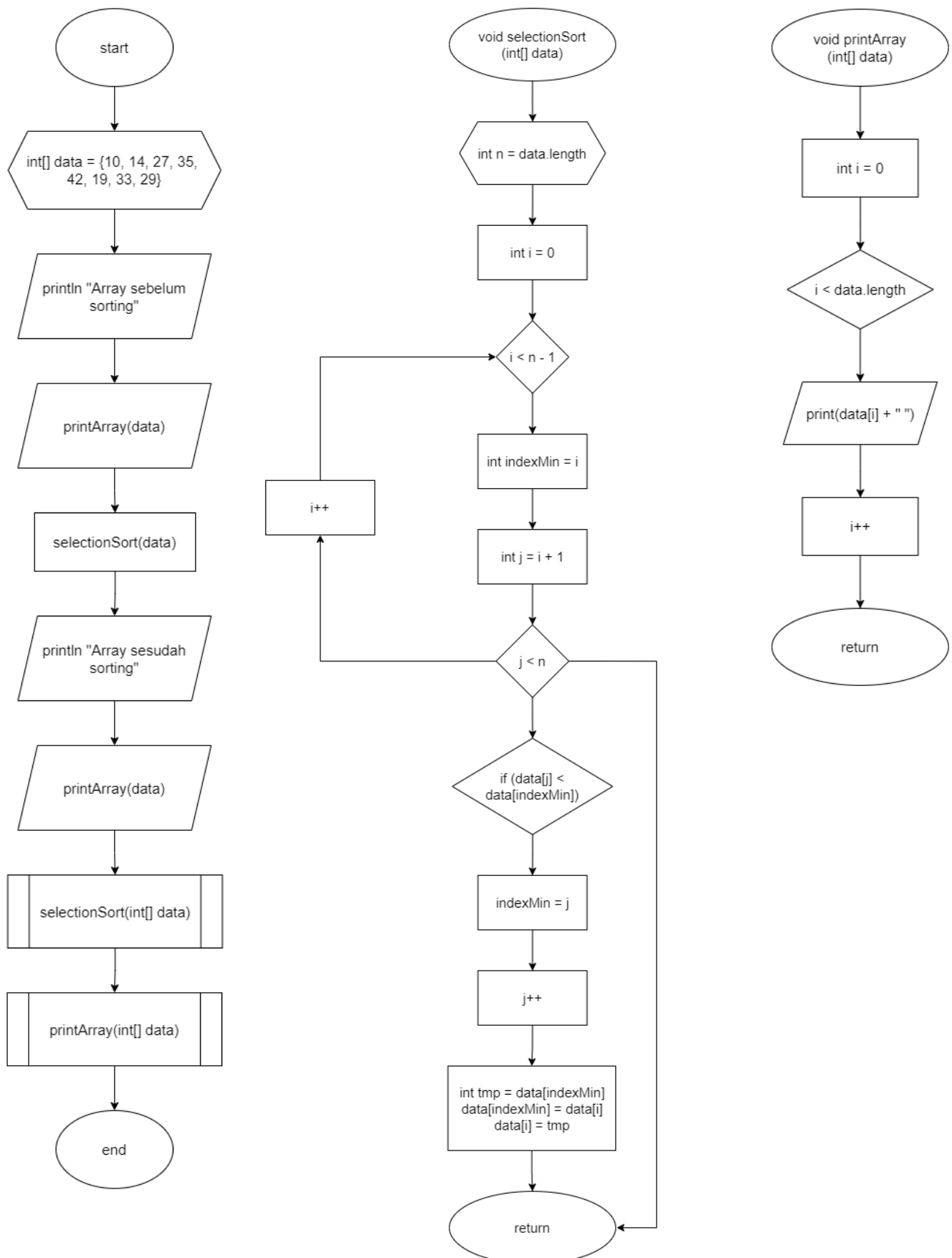
1. Algoritma Bubble Sort

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



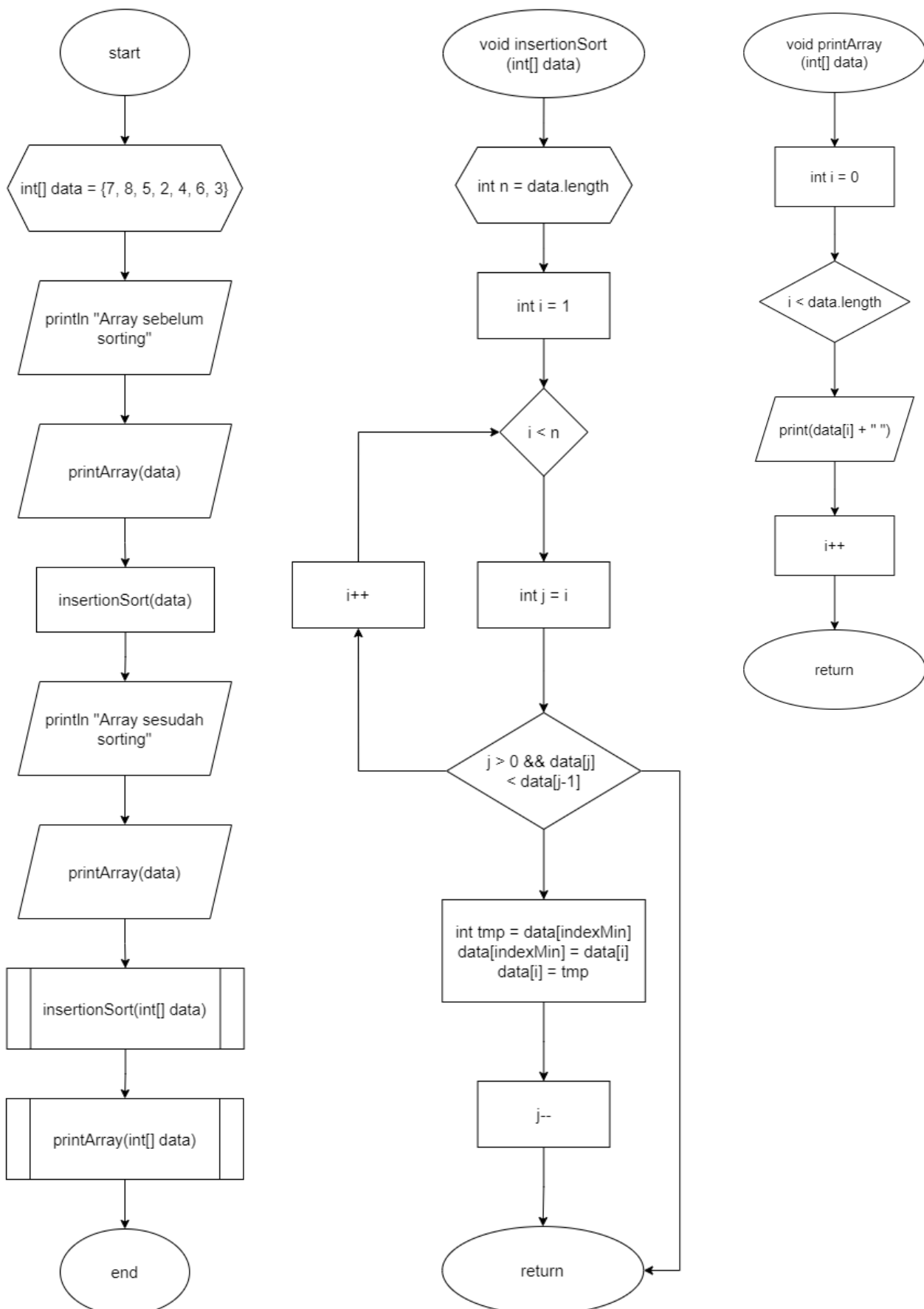
2. Algoritma Selection Sort

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



3. Algoritma Insertion Sort

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



LATIHAN

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

Gambarkan proses penyelesaian kasus pengurutan descending data di atas dengan menggunakan algoritma

a. Bubble sort

{23, 35, 7, 14, 67}

- Tahap 0 (4 langkah)
23 -> 35 swap : {35, 23, 7, 14, 67}
23 -> 7 no swap : {35, 23, 7, 14, 67}
7 -> 14 swap : {35, 23, 14, 7, 67}
7 -> 67 swap : {35, 23, 14, 67, 7}
- Tahap 1 (3 langkah)
35 -> 23 no swap : {35, 23, 14, 67, 7}
23 -> 14 no swap : {35, 23, 14, 67, 7}
14 -> 67 swap : {35, 23, 67, 14, 7}
- Tahap 2 (2 langkah)
35 -> 23 no swap : {35, 23, 67, 14, 7}
23 -> 67 swap : {35, 67, 23, 14, 7}
- Tahap 3 (1 langkah)
35 -> 67 swap : {67, 35, 23, 14, 7}

b. Selection sort

{23, 35, 7, 14, 67}

- Tahap 0
Index = 0 ; id = 0
Max awal = 23
35 > 23 (max = 35, id = 1)
7 > 35
14 > 35
67 > 35 (max = 67, id = 4)
Swap index 0 dan id 4 {67, 35, 7, 14, 23}
- Tahap 1
Index = 1 ; id = 1
Max awal = 35
7 > 35

14 > 35

23 > 35

Swap index 1 dan id 1 {67, 35, 7, 14, 23}

- Tahap 2

Index = 2 ; id = 2

Max awal = 7

14 > 7 (max = 14, id = 3)

23 > 14 (max = 23, id = 4)

Swap index 2 dan id 4 {67, 35, 23, 14, 7}

- Tahap 3

Index = 3 ; id = 3

Max awal = 14

7 > 14

Swap index 3 dan id 3 {67, 35, 23, 14, 7}

Hasil akhir pengurutan = {67, 35, 23, 14, 7}

c. Insertion sort

{23, 35, 7, 14, 67}

- Langkah 1



Data terdiri dari 2 bagian, sorted dan unsorted Pada langkah pertama item index pertama dari data langsung menjadi bagian sorted Sisanya menjadi bagian unsorted

- Langkah 2



Dimulai dari nilai ke 2, dibandingkan dengan nilai di kirinya (bagian sorted)

Jika nilai di kirinya lebih besar, maka posisi tetap dan nilai ke 2 menjadi bagian sorted

- Langkah 3



Kemudian nilai ke 3, dibandingkan dengan kirinya. Posisinya masih tetap karena nilai di kirinya lebih besar. Proses ini akan terus berulang. Jika ditemukan nilai sorted lebih besar dari nilai yang disimpan, maka berhenti dan nilai yang disimpan disisipkan di posisi terakhir bergeser

- Langkah 4



{ 35, 23, 7, 7, 67 }

{ 35, 23, 14, 7, 67 }



{ 35, 23, 14, 7, 7 }

{ 35, 23, 14, 14, 7 }

{ 35, 23, 23, 14, 7 }

{ 35, 35, 23, 14, 7 }

{ 67, 35, 23, 14, 7 }