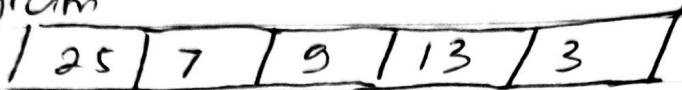


## TUGAS 4

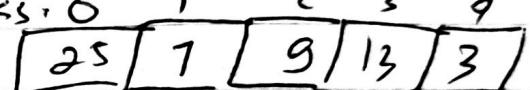
- 1.) Gambaran ilustrasi algoritma insertion sort  
→ kode program



→ array sebelum terurut



indeks: 0 1 2 3 4

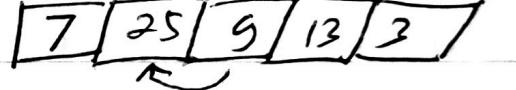


→ Cek bilangan indeks-1 (7).

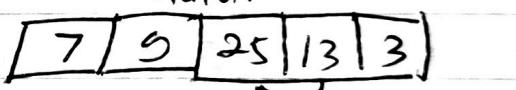
apakah lebih kecil dr indeks-0 (25)



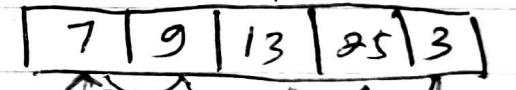
→ indeks-2 (9) lebih ~~lebih~~ kecil dr indeks-1 (25), maka tukar



→ indeks-3 (13) lebih kecil dr dari indeks-2 (25), maka tukar



→ indeks-4 (3) lebih kecil dari indeks-3 (-2), -1, -0, maka tukar



→ semua angka berurut.



⇒ pada gambaran diatas terlihat pergeseran array dilakukan dari  $i = 1$  yang kemudian dibandingkan dengan array yg di sebelah kiri. Apabila array kedua lebih kecil dari array pertama, maka dilakukan pertukaran hingga tersusun secara berurut.

2) Gambaran ilustrasi algoritma bubble sort.  
→ kode program

25	7	9	13	3
----	---	---	----	---

$i = 1$

25	7	9	13	3
----	---	---	----	---

⇒ pada saat  $i = 1$ ,

nilai  $j$  diulang

25	7	9	3	13
----	---	---	---	----

↑ tukar

dari 4 sampai 1,

25	7	3	9	13
----	---	---	---	----

↑ tukar

Pada pengulangan pertama, data [4] dibanding data [3],

25	3	7	9	13
----	---	---	---	----

↑ tukar

karena  $3 < 13$ , maka ditukar

$i = 2$

$j = 4$

3	25	7	9	13
---	----	---	---	----

↑  
 $j$

⇒ pada saat  $i = 2$ , nilai  $j$  diulang dari 4 sampai 2.

3	25	7	9	13
---	----	---	---	----

↑  
 $j$

Pada pengulangan pertama data [4]

dibandingkan data [3]

3	25	7	9	13
---	----	---	---	----

↑  
tukar

karena  $9 < 13$ , maka tidak ditukar

\* Dan seterusnya  $i = 9$

$i = 3$        $j = 4$  = 

3	7	25	9	13
---	---	----	---	----

  
4)

$i = 3$        $j = 3$ 

3	7	25	9	13
---	---	----	---	----

  
↑  
tukar

$i = 4$        $j = 4$ 

3	7	9	25	13
---	---	---	----	----

  
↑  
tukar  
Akhir 

3	7	9	13	25
---	---	---	----	----

### 3) Selection sort :

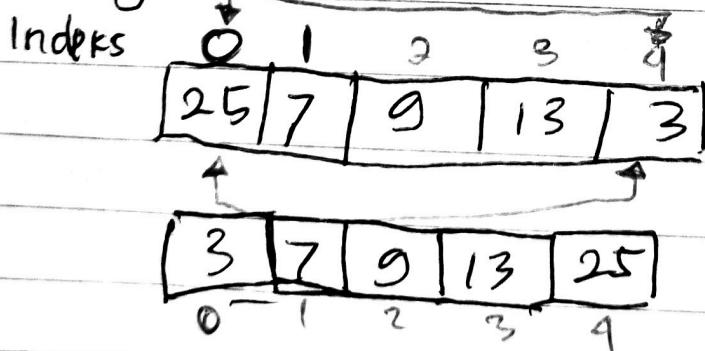
→ Berikut data yang akan diurutkan -

25	7	0	13	3
----	---	---	----	---

→ pointer pertama adalah indeks ke-0.

Cek apakah ada data sesudahnya yang lebih kecil dari data indeks ke-0.

Jika ada, maka ditukar



→ pointer kedua adalah indeks ke-1 cek apakah ada data sesudahnya yang lebih kecil dari data indeks ke-1. jika ada, maka tulis. jika tidak ada, maka posisinya tetap sama.

	3	7	9	13	25
indeks	0	1	2	3	4

\* Langkah ini dicelang sampai pointernya berada di indeks ke-3.

pointer =	3	7	9	13	25
i = 2	0	1	2	3	4

pointer =	3	7	9	13	25
i = 3	0	1	2	3	4

Akhir →	3	7	9	13	25
---------	---	---	---	----	----

#### 4) shell sort

→ Kode yang harus drurutkan

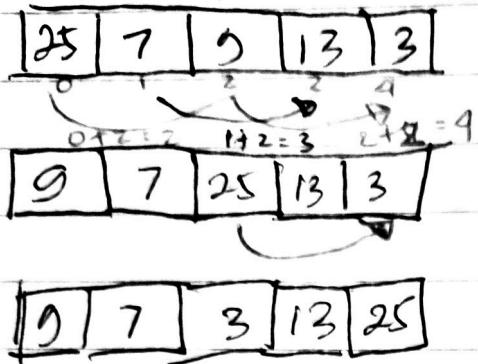
25	7	9	13	3
----	---	---	----	---

→ pertama, menentukan jarak pertama dengan cara  $N/2$ .  $N$  = jumlah data

$$N = 5 \rightarrow \frac{N}{2} = \frac{5}{2} = 2,5 \text{ (dibulatkan kebawah)}$$

Jadi ↴

Jarak Pertama = 2



\* penjelasan

Cek, apakah data yang dibandingkan lebih kecil, jika ya, maka tukar contohnya = indeks[0] dg indeks(1)  
 $9 < 25$ , maka perlu ditukar. Jika tdk, maka tdk ditukar. contohnya = indeks[1] dg indeks(3)  
 $13 > 7$ , maka tdk ditukar

→ kedua, menentukan jarak kedua dg cara yang sama  $N/2$ . Tetapi,  $N$  disini mengambil pada jarak pertama yaitu 2. Jadi  $N=2$ .

$$N/2 \rightarrow 2/2 = 1 \text{ (jarak kedua)}$$



\* step kedua ini sama dg metode insertion sort

Akhir → [3 | 7 | 9 | 13 | 25]

7) program menggunakan satu algoritma sorting dan satu algoritma searching

→ insertion sort

→ Data yg akan diurutka

25 | 7 | 9 | 13 | 3 |

↑  
titik

7 | 25 | 9 | 13 | 3 |

↑  
titik

7 | 9 | 25 | 13 | 3 |

↑  
titik

7 | 9 | 13 | 25 | 3 |

↑  
titik

Akhir → 3 | 7 | 9 | 13 | 25 |

yang kedua melakukan pencarian dengan menggunakan algoritma Sequential Search.

→ Data yang urut

3 | 7 | 9 | 13 | 25 |

→ Data array B

→ indeks

Misalkan dari data diatas yang akan dicari adalah angka 13 dalam array B, maka proses yang akan terjadi pada proses pencarian adalah Sbg berikut :

1. pencarian dimulai pada indeks 0 yaitu angka 3, kemudian dicocokan dg angka yg akan dicari yaitu 13. Jika tdk sama, maka pencarian akan dilanjutkan ke indek selanjutnya.
2. pada indeks ke-3 , yaitu angka 13, ternya angka 13 merupakan angka yang dicari. Pencarian angka telah ditemukan, maka pencarian ditentukan dan dihentikan keluar dari looping pencarian .