

Jurnal Praktikum CII1F4

Modul 11 – Sorting

Catatan: Setiap solusi untuk tugas praktikum ini sudah menggunakan subprogram dan struktur kontrol, seperti perulangan dan percabangan.

1. **[Kompetisi]** Kompetisi pemrograman yang baru saja berlalu diikuti oleh 17 tim dari berbagai perguruan tinggi ternama. Dalam kompetisi tersebut, setiap tim berlomba untuk menyelesaikan sebanyak mungkin problem yang diberikan. Dari 13 problem yang diberikan, ada satu problem yang menarik. Problem tersebut mudah dipahami, hampir semua tim mencoba untuk menyelesaikannya, tetapi hanya 3 tim yang berhasil. Apa sih problemnya?

"Median adalah nilai tengah dari suatu koleksi data yang sudah terurut. Jika jumlah data genap, maka nilai median adalah rerata dari kedua nilai tengahnya. Pada problem ini, semua data merupakan bilangan bulat positif, dan karenanya rerata nilai tengah dibulatkan kebawah."

Buatlah program median yang mencetak nilai median terhadap seluruh data yang sudah terbaca, jika data yang dibaca saat itu adalah 0.

Masukan berbentuk rangkaian bilangan bulat. Masukan tidak akan berisi lebih dari **1000000** data, tidak termasuk bilangan **0**. Data **0** merupakan tanda bahwa median harus dicetak, tidak termasuk data yang dicari mediannya. Data masukan diakhiri dengan bilangan bulat **-5313541**.

Keluaran adalah median yang diminta, satu data perbaris.

Petunjuk:

- a. Untuk setiap data bukan 0 (dan bukan marker -5313541) simpan ke dalam array.
- b. dan setiap kali menemukan bilangan 0, urutkanlah data yang sudah tersimpan dengan menggunakan metoda selection sort dan ambil mediannya.

Contoh

1	Masukan	7 23 11 <u>0</u> 5 19 2 29 3 13 17 <u>0</u> -5313541
	Keluaran	11 12
	Penjelasan	Sampai bilangan 0 yang pertama, data terbaca adalah 7 23 11, setelah tersusun: 7 11 23, maka median saat itu adalah 11. Sampai bilangan 0 yang kedua, data adalah 7 23 11 5 19 2 29 3 13 17, setelah tersusun diperoleh: 2 3 5 7 11 13 17 19 23 29. Karena ada 10 data, genap, maka median adalah $(11+13)/2=12$.
2	Masukan	23 12 24 26 20 10 25 8 <u>0</u> -5313541
	Keluaran	21.5
	Penjelasan	Sampai bilangan 0 yang pertama, data terbaca adalah 23 12 24 26 20 10 25 8, setelah tersusun: 8 10 12 20 23 24 25 26, maka median saat itu adalah $(20+23)/2=21.5$.

Jurnal Praktikum CII1F4

Modul 11 – Sorting

```
package main
import "fmt"

const NMAX = ...
type arrInt [...]int

func sorting(T *arrInt, n int){
    /* I.S. terdefinisi array T yang berisi sejumlah n bilangan bulat
       F.S. array T terurut secara membesar berdasarkan algoritma selection sort
    */
    ...
}

func median(T arrInt, n int) float64 {
    /* mengembalikan median dari array T yang berisi sejumlah n bilangan bulat yang
       telah terurut membesar */
    ...
}

func main(){
    ...
}
```

2. **[tie-break]** Di masa pandemi covid-19 ini English Premier League sudah memasuki minggu ke-11. Tanpa kehadiran langsung fans fanatik di stadion, gairah pemain menunjukkan kemampuan mereka secara optimal jauh berkurang. Tidak ada gol-gol spektakular atau permainan menggocek bola yang cantik.

Untuk mendorong gairah pemain, penyelenggara menampilkan peringkat pencetak gol terbanyak sementara. Data itu ingin disusun **secara menurun terhadap perolehan gol dan assist**. Atribut data adalah **nama, jumlah gol, dan jumlah assist**. Jika ada dua orang pemain atau lebih memiliki jumlah gol yang sama, maka assist yang lebih banyak akan meletakkan pemain tersebut pada peringkat lebih atas.

Masukan terdiri atas **n** (tipe data integer) yang menyatakan jumlah pencetak gol, diikuti sebanyak **n** baris berikutnya berupa nama (dua tipe data string), jumlah gol (tipe data integer), dan jumlah assist (tipe data integer).

Keluaran terdiri atas data pemain (nama, jumlah goal, jumlah assist) yang tersusun menurun sesuai dengan kriteria di atas. Untuk kebutuhan pembuatan program, algoritma sorting yang boleh digunakan adalah Selection sort atau Insertion sort. Selain itu anggap bahwa jumlah pemain yang diperhitungkan dalam peringkat tidak lebih dari 1001 orang pemain saja.

Contoh:

No	Masukan	Keluaran
1	9 Bruno Fernandes 7 3 Jamie Vardy 8 1 Dominic Calvert-Lewin 10 2	Dominic Calvert-Lewin 10 2 Son Heung-Min 9 2 Mohamed Salah 8 2 Jamie Vardy 8 1

Jurnal Praktikum CII1F4

Modul 11 – Sorting

	Son Heung-Min 9 2 Harry Kane 7 2 Ollie Watkins 6 1 Patrick Bamford 7 1 Callum Wilson 7 0 Mohamed Salah 8 2	Bruno Fernandes 7 3 Harry Kane 7 2 Patrick Bamford 7 1 Callum Wilson 7 0 Ollie Watkins 6 1
2	7 Bruno Fernandes 7 3 Dominic Calvert-Lewin 7 12 Son Heung-Min 7 2 Harry Kane 5 2 Patrick Bamford 5 1 Callum Wilson 5 0 Mohamed Salah 3 2	Dominic Calvert-Lewin 7 12 Bruno Fernandes 7 3 Son Heung-Min 7 2 Harry Kane 5 2 Patrick Bamford 5 1 Callum Wilson 5 0 Mohamed Salah 3 2

3. **[PilLeg]** Sebuah program digunakan untuk menghitung perolehan suara dari berbagai partai politik dalam sebuah pemilihan umum calon legislatif. Program akan menampilkan data partai terurut berdasarkan perolehan suara terurut. Nama partai hanya disimbolkan dari angka 1 hingga N, di mana $1 \leq N \leq 1000000$.

Masukan berupa beberapa nilai yang dipisahkan oleh spasi. Masing-masing nilai menyatakan nama partai (1 hingga N) yang dipilih. Proses input ini diakhiri dengan nilai -1.

Keluaran berupa daftar partai dan peroleh suaranya yang terurut descending atau mengecil dengan format <partai><suara>. Perhatikan contoh masukan dan keluaran yang diberikan. **Petunjuk** : gunakan tipe bentukan partai yang berisi nama dan suara. Data perolehan suara disimpan pada array of partai (kapasitas 1000000). Array tersebutlah yang diurutkan.

Contoh:

No	Masukan	Keluaran
1	5 1 1 1 1 1 1 1 3 3 3 3 3 2 2 5 5 5 5 5 4 3 2 2 2 2 -1	1(7) 5(6) 3(6) 2(6) 4(1)
2	5 8 8 5 6 8 8 7 6 5 8 7 5 6 7 5 8 6 7 8 8 7 7 8 6 7 7 6 8 6 8 8 5 5 6 6 6 7 7 6 7 8 8 8 5 7 6 6 8 6 5 5 8 7 5 5 6 8 7 6 5 5 8 6 6 7 8 8 8 6 7 6 6 5 7 8 7 6 6 6 8 7 7 8 6 5 5 7 7 6 5 7 8 8 6 8 8 6 7 8 -1	8(30) 6(28) 7(24) 5(18)
3	8 8 8 8 8 8 8 8 8 8 8 8 8 8 -1	8(15)
4	10 1 7 8 10 1 4 8 8 5 -1	8(3) 10(2) 1(2) 7(1) 4(1) 5(1)
5	14 10 13 13 14 10 11 13 13 12 15 11 10 -1	13(4) 10(3) 14(2) 11(2) 12(1) 15(1)
6	-1	

Lengkapi program berikut ini

```
package main
import "fmt"
const NMAX = 1000000
// tipe bentukan partai
```

Jurnal Praktikum CII1F4

Modul 11 – Sorting

```
...  
// tipe tabPartai: array of partai dengan kapasitas NMAX  
...  
func main(){  
    // deklarasi variabel  
    ...  
    // lakukan proses input suara secara berulang di sini, simpan ke dalam array  
    p, sehingga terdapat array p yang berisi hasil peroleh suara n partai.  
    ...  
    // lakukan proses pengurutan dengan insertion sort descending berdasarkan  
    jumlah suara yang diperoleh  
    ...  
    // tampilkan array p  
    ...  
}  
  
func posisi(t tabPartai, n int, nama int) int {  
    /* mengembalikan indeks partai yang memiliki nama yang dicari pada array t yang  
    berisi n partai atau -1 apabila tidak ditemukan , gunakan pencarian sekuensial  
    */  
    ...  
}
```