

Test Alpro

Nama : Rakha Arbiyandanu

NIM : 2311102263

1. Sebuah program yang digunakan untuk mencari sebuah irisan himpunan. Masukan terdiri dari dua baris. Setiap barisnya yang berisi sekumpulan bilangan. Masukan disetiap barisnya akan berakhir apabila bilangan yang diberikan sudah pernah diberikan pada baris tersebut (atau duplikat), Catatan: anggota suatu himpunan tidak boleh duplikat. Keluaran adalah sekumpulan bilangan yang menyatakan irisan dari himpunan pada baris pertama dan baris kedua pada masukan.

Jawab :

Source Code

```
package main

import "fmt"

func bacaHimpunan_2311102263() map[int]struct{} { //Untuk menyimpan data pada setiap bilangan yang diinputkan dan membaca data bilangan duplikat atau tidaknya
    himpunan := make(map[int]struct{})
    for {
        var bilangan int
        fmt.Println("Masukkan bilangan : ")
        _, err := fmt.Scan(&bilangan)
        if err != nil {
            fmt.Println("Masukkan bilangan bulat.")
            continue
        }

        if _, double := himpunan[bilangan]; double {
            fmt.Printf("%d duplikat, masukan berhenti.\n", bilangan)
            break
        }
        himpunan[bilangan] = struct{}{}
    }

    return himpunan
}

func main() { //fungsi ini adalah fungsi pertama untuk menjalankan program seperti membaca data, cek data dll
    fmt.Println("Masukkan bilangan untuk himpunan pertama:")
    himpunan1 := bacaHimpunan_2311102263()

    fmt.Println("\nMasukkan bilangan untuk himpunan kedua:")
    himpunan2 := bacaHimpunan_2311102263()

    irisan := make(map[int]struct{})
    for bilangan := range himpunan1 {
        if _, exists := himpunan2[bilangan]; exists {
            irisan[bilangan] = struct{}{}
        }
    }

    fmt.Println("\nHimpunan pertama:", keys(himpunan1))
    fmt.Println("Himpunan kedua:", keys(himpunan2))
    fmt.Println("Irisan kedua himpunan:", keys(irisan))
}

func keys(m map[int]struct{}) []int { //fungsi ini digunakan untuk membuat struct digubakan untuk membagi bilangan dalam setiap baris
    result := []int{}
    for k := range m {
        result = append(result, k)
    }
    return result
}
```

Output :

```
PS D:\SEMESTER 3\Algoritma Pemrograman 2\Praktikum\TestAlpro> go run "d:\SEMESTER 3\Algoritma Pemrograman 2\Praktikum\TestAlpro\TestAlpro1.go"
Masukkan bilangan untuk himpunan pertama:
Masukkan bilangan : 1 1
Masukkan bilangan : 1 duplikat, masukan berhenti.

Masukkan bilangan untuk himpunan kedua:
Masukkan bilangan : 1 1
Masukkan bilangan : 1 duplikat, masukan berhenti.

Himpunan pertama: [1]
Himpunan kedua: [1]
```

2. Suatu tabel digunakan untuk mencatat data mahasiswa. Mahasiswa memiliki atribut NIM, nama, dan nilai. Setiap data baru akan selalu ditambahkan ke dalam tabel di indeks N+1. N adalah jumlah data saat ini di dalam array. Sehingga pada tabel mungkin terdapat beberapa data untuk seorang mahasiswa

Jawab :

Source Code

```
1 package main
2
3 import "fmt"
4
5 const nMax = 51 // sebagai batas data mahasiswa yang dapat di inputkan
6
7 type mahasiswa struct { // menggunakan struct untuk mengatur fungsi fungsi yang ada
8     NIM string
9     nama string
10    nilai int
11 }
12
13 // fungsi untuk membaca data mahasiswa dan menggunakan fungsi slice agar data agar sesuai dengan data yang di input
14 func inputMahasiswa_2311180263(n int) []mahasiswa {
15     data := make([]mahasiswa, n)
16     for i := 0; i < n; i++ {
17         fmt.Printf("Masukkan data mahasiswa ke-%d (NIM nama nilai): ", i+1)
18         fmt.Scan(&data[i].NIM, &data[i].nama, &data[i].nilai)
19     }
20     return data
21 }
22
23 // fungsi mencari data mahasiswa dengan nim
24 func cariNilaiPertama(data []mahasiswa, nim string) (int, bool) {
25     for _, mhs := range data {
26         if mhs.NIM == nim {
27             return mhs.nilai, true
28         }
29     }
30     return 0, false
31 }
32
33 // fungsi untuk mencari nilai terbesar dalam dengan bantuan slice
34 func cariNilaiTerbesar(data []mahasiswa, nim string) (int, bool) {
35     max := -1
36     found := false
37     for _, mhs := range data {
38         if mhs.NIM == nim {
39             found = true
40             if mhs.nilai > max {
41                 max = mhs.nilai
42             }
43         }
44     }
45     return max, found
46 }
47
48 func main() { // fungsi untuk menjalankan seluruh program main redclared in this block
49     var n int
50     fmt.Printf("Masukkan jumlah mahasiswa: ")
51     fmt.Scan(&n)
52
53     if n <= 0 || n > nMax {
54         fmt.Printf("Jumlah mahasiswa harus antara 1 hingga %d.\n", nMax)
55         return
56     }
57
58     data := inputMahasiswa_2311180263(n)
59
60     var nim string
61     fmt.Printf("Masukkan NIM untuk pencarian: ")
62     fmt.Scan(&nim)
63
64     // Cari nilai pertama
65     if nilaiPertama, found := cariNilaiPertama(data, nim); found {
66         fmt.Printf("Nilai pertama mahasiswa dengan NIM %s adalah %d\n", nim, nilaiPertama)
67     } else {
68         fmt.Printf("Mahasiswa dengan NIM %s tidak ditemukan\n", nim)
69     }
70
71     // Cari nilai terbesar
72     if nilaiTerbesar, found := cariNilaiTerbesar(data, nim); found {
73         fmt.Printf("Nilai terbesar mahasiswa dengan NIM %s adalah %d\n", nim, nilaiTerbesar)
74     } else {
75         fmt.Printf("Mahasiswa dengan NIM %s tidak ditemukan\n", nim)
76     }
77 }
78 }
```

Output :

```
PS D:\SEMESTER 3\Algoritma Pemrograman 2\praktikum\testAlpro> go run "d:\SEMESTER 3\Algoritma Pemrograman 2\praktikum\testAlpro\testAlpro2.go"
Masukkan jumlah mahasiswa: 5
Masukkan data mahasiswa ke-1 (NIM nama nilai): 2311902341 Fahrul 80
Masukkan data mahasiswa ke-2 (NIM nama nilai): 2311902354 Alfin 90
Masukkan data mahasiswa ke-3 (NIM nama nilai): 2311902357 Dzul 95
Masukkan data mahasiswa ke-4 (NIM nama nilai): 2311902351 Yudi 70
Masukkan data mahasiswa ke-5 (NIM nama nilai): 2311902380 Bryan 95
Masukkan NIM untuk pencarian: 2311902341
Nilai pertama mahasiswa dengan NIM 2311902341 adalah 80
Nilai terbesar mahasiswa dengan NIM 2311902341 adalah 80
```

- Sebuah program digunakan untuk mengolah data nama provinsi, populasi, dan angka pertumbuhan penduduk provinsi di Indonesia pada tahun 2018. Halaman 93 | Modul Praktikum Algoritma dan Pemrograman 2 Masukan terdiri dari 35 baris, yang mana masing-masing barisnya terdiri dari tiga nilai yang menyatakan nama provinsi, jumlah populasi provinsi (bilangan bulat), dan angka pertumbuhan (riil) provinsi tersebut. Pada baris terakhir hanya sebuah string yang menyatakan nama provinsi yang akan dicari. Keluaran terdiri dari 36 baris. Baris pertama adalah nama provinsi dengan angka pertumbuhan tercepat. Baris kedua adalah indeks provinsi yang dicari sesuai dengan nama provinsi yang ditulis pada masukan baris terakhir. Terakhir terdiri dari 34 baris yang menampilkan nama provinsi beserta prediksi jumlah penduduk pada provinsi tersebut di tahun depannya, khusus yang memiliki pertumbuhan di atas 2%.

Source Code

```
1 package main
2
3 import "fmt"
4
5 const nProv = 35 //digunakan untuk membaca data
6
7 type Provinsi struct {
8     nama      string
9     populasi  int
10    pertumbuhan float
11 }
12
13 type arrayProvint [nProv]Provinsi
14
15 func input_20180208(a int) arrayProvint { //inputan untuk provinsi
16     var data arrayProvint
17     for i := 0; i < a; i++ {
18         fmt.Printf("Masukkan data provinsi ke-%d (Nama Provinsi Pertumbuhan): ", i+1)
19         data[i] = Provinsi{nama: "", populasi: 0, pertumbuhan: 0}
20     }
21     return data
22 }
23
24 func ProvinsiTerlengkap(data arrayProvint, n int) int { //mencari provinsi dengan pertumbuhan tertinggi
25     maxIndex := 0
26     for i := 0; i < n; i++ {
27         if data[i].pertumbuhan > data[maxIndex].pertumbuhan {
28             maxIndex = i
29         }
30     }
31     return maxIndex
32 }
33
34 func IndeksProvinsi(data arrayProvint, n int, nama string) int { //mencari provinsi berdasarkan nama
35     for i := 0; i < n; i++ {
36         if data[i].nama == nama {
37             return i
38         }
39     }
40     return -1
41 }
42
43 func HitungPopulasi(data arrayProvint, index int) int { //fungsi untuk menghitung populasi tahun berikutnya
44     return int(float(data[index].populasi) * (1 + data[index].pertumbuhan/100))
45 }
46
47 func main() { //fungsi untuk menjalankan seluruh program... note redclared in this block
48     var a int
49     fmt.Println("Masukkan jumlah provinsi: ")
50     fmt.Scan(&a)
51     if a > nProv {
52         fmt.Printf("Jumlah provinsi tidak boleh lebih dari %d\n", nProv)
53         return
54     }
55     data := input_20180208(a)
56
57     // Menampilkan provinsi dengan pertumbuhan tertinggi
58     tinggiIndex := ProvinsiTerlengkap(data, a)
59     fmt.Printf("Provinsi dengan pertumbuhan tertinggi adalah %s\n", data[tinggiIndex].nama)
60
61     // Mencari provinsi berdasarkan nama
62     var namaCari string
63     fmt.Println("Masukkan nama provinsi yang ingin dicari: ")
64     fmt.Scan(&namaCari)
65     indexCari := IndeksProvinsi(data, a, namaCari)
66
67     if indexCari != -1 {
68         fmt.Printf("Provinsi ditemukan: %s dengan populasi %d dan pertumbuhan %.2f\n", data[indexCari].nama, data[indexCari].populasi, data[indexCari].pertumbuhan)
69
70         // Menghitung populasi di tahun berikutnya
71         populasiBaru := HitungPopulasi(data, indexCari)
72         fmt.Printf("Populasi provinsi %s di tahun depan %d\n", data[indexCari].nama, populasiBaru)
73
74         // Menampilkan data jika pertumbuhan di atas 2%
75         if data[indexCari].pertumbuhan > 2 {
76             fmt.Printf("Provinsi %s memiliki pertumbuhan di atas 2%\n", data[indexCari].nama)
77         }
78     } else {
79         fmt.Printf("Provinsi dengan nama %s tidak ditemukan.\n", namaCari)
80     }
81 }
```

Output :

```
PS D:\SEMESTER 3\Algoritma Pemrograman 2\Praktikum\TestAlpro> go run "d:\SEMESTER 3\Algoritma Pemrograman 2\Praktikum\TestAlpro\TestAlpro3.go"
# command-line-arguments
.\TestAlpro3.go:51:14: syntax error: unexpected /, expected }
.\TestAlpro3.go:53:2: syntax error: non-declaration statement outside function body
.\TestAlpro3.go:68:2: syntax error: non-declaration statement outside function body
PS D:\SEMESTER 3\Algoritma Pemrograman 2\Praktikum\TestAlpro> go run "d:\SEMESTER 3\Algoritma Pemrograman 2\Praktikum\TestAlpro\TestAlpro3.go"
Masukkan jumlah provinsi: 2
Masukkan data provinsi ke-1 (Nama Populasi Pertumbuhan): Jsteng 5000000 1%
Masukkan data provinsi ke-2 (Nama Populasi Pertumbuhan): Jabar 6000000 2%
Provinsi dengan pertumbuhan tertinggi adalah Jabar
Masukkan nama provinsi yang ingin dicari: Jabar
Provinsi ditemukan: Jabar dengan populasi 6000000 dan pertumbuhan 2.00%
Populasi provinsi Jabar di tahun depan: 6120000
PS D:\SEMESTER 3\Algoritma Pemrograman 2\Praktikum\TestAlpro>
```

4. Kompetis 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 11 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? Halaman Freskum Algoritma Pemrograman "Mesdhas" adalah na tengah dari suatu koldt data, yang sudah terujich data genep, mak tal medies adalah revuta dari bendus sal tengahnya. Buatlah program median yang mencetak nilai median terhadap selunch data yang sudah terbecs, ha data yang dibaca saut tu adalah . Masulkan berbentuk ranghaian bilangan bulat. Masukan tidak akan berts lebih dari 1000000 data, tidak termasuk belangen 0. Data 0 merupakan tanda bahwa median harus dicetak, tidak termasuk data yang dicart medlarnys. Data masukan chabhet dengan bilangan bulat -5313541. Keluaran adalah median yang diminta, satu data perharis.

Source Code

```
1 package main
2
3 import (
4     "fmt"
5     "sort"
6 )
7
8 func main() { //di dalam fungsi ini terdapat perintah untuk menghitung median dan menambahkan angka ke array main redeclared in this block
9     const END_MARKER = -5313541
10    var numbers []int
11    var input int
12
13    for {
14        fmt.Scan(&input)
15        if input == END_MARKER {
16            break
17        }
18        if input == 0 {
19            if len(numbers) == 0 {
20                fmt.Println(0)
21            } else {
22                median := menghitungMedian_2311102263(numbers)
23                fmt.Printf("%i\n", median)
24            }
25        } else if {
26            numbers = append(numbers, input)
27        }
28    }
29 }
30
31 func menghitungMedian_2311102263(arr []int) float64 { //di dalam fungsi menghitung median ini terdapat perintah untuk mengurutkan array dan duplikat dan menentukan jumlah data apakah ganjil maupun genap
32    // Salin array dan urutkan
33    sortedArr := make([]int, len(arr))
34    copy(sortedArr, arr)
35    sort.Ints(sortedArr)
36
37    n := len(sortedArr)
38    if n%2 == 1 {
39        return float64(sortedArr[n/2])
40    }
41    return float64(sortedArr[n/2-1]+sortedArr[n/2]) / 2.0
42 }
43 }
```

Output :

```
PS D:\SEMESTER 3\Algoritma Pemograman 2\Praktikum\TestAlpro> go run "d:\SEMESTER 3\Algoritma Pemograman 2\Praktikum\TestAlpro\TestAlpro4.go"
7 23 11 8 5 19 29 13 17 0 0 -5313541
13.0
13.0
PS D:\SEMESTER 3\Algoritma Pemograman 2\Praktikum\TestAlpro> []
```

5. 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 (1 New 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 struct partai yang berisi nama dan suara. Data perolehan suara disimpan pada array of partai (kapasitas 1000000). Array tersebutlah yang diurutkan.

Source Code

```
1 package main
2
3 import (
4     "fmt"
5     "sort"
6 )
7
8 // Struct untuk menyimpan data partai (ID dan jumlah suara)
9 type Partai_2311102263 struct {
10     ID int
11     Suara int
12 }
13
14 func main() { //fungsi utas untuk menjalankan fungsi fungsi main redeclared in this block (see details)
15
16     const PENANDA_AKHIR = -1
17     const JUMLAH_MAX_PARTAI = 1000000
18
19     // Array untuk menyimpan jumlah suara tiap partai
20     suara := make([]int, JUMLAH_MAX_PARTAI+1)
21
22     var input int
23
24     // Membaca input suara
25     for {
26         fmt.Scan(&input)
27         if input == PENANDA_AKHIR {
28             break
29         }
30         // Menambahkan suara ke partai terkait
31         suara[input]++
32     }
33
34     var daftarPartai []Partai_2311102263
35     for i := 1; i <= JUMLAH_MAX_PARTAI; i++ {
36         if suara[i] > 0 {
37             daftarPartai = append(daftarPartai, Partai_2311102263{ID: i, Suara: suara[i]})
38         }
39     }
40
41     // Mengurutkan partai berdasarkan jumlah suara (descending)
42     // Jika jumlah suara sama, diurutkan berdasarkan ID partai (ascending)
43     sort.Slice(daftarPartai, func(i, j int) bool {
44         if daftarPartai[i].Suara == daftarPartai[j].Suara {
45             return daftarPartai[i].ID < daftarPartai[j].ID
46         }
47         return daftarPartai[i].Suara > daftarPartai[j].Suara
48     })
49
50     // Untuk Menampilkan hasil
51     for _, partai := range daftarPartai {
52         fmt.Printf("%d(%d) ", partai.ID, partai.Suara)
53     }
54     fmt.Println()
55 }
56 }
```

Output :

```
PS D:\SEMESTER 3\Algoritma Pemograman 2\Praktikum\TestAlpro> go run "d:\SEMESTER 3\Algoritma Pemograman 2\Praktikum\TestAlpro\TestAlpro.go"
go run "d:\SEMESTER 3\Algoritma Pemograman 2\Praktikum\TestAlpro\TestAlpro.go"
8 8 8 8 8 8 8 8 8 8 8 8 8 8 8 -1
2(34) 3(20) 8(15)
PS D:\SEMESTER 3\Algoritma Pemograman 2\Praktikum\TestAlpro> |
```