

**LAPORAN PRAKTIKUM  
ALGORITMA DAN PEMROGRAMAN 2**

**Test ALPRO**



**Disusun Oleh :**

**Bhara Yudha Pamungkas / 2311102186**

**IF-11-06**

**Dosen Pengampu :**

**Abednego Dwi Septiadi, S.Kom., M.Kom**

**PROGRAM STUDI S1 TEKNIK INFORMATIKA**

**FAKULTAS INFORMATIKA**

**TELKOM UNIVERSITY PURWOKERTO**

**2024**

## UNGUIDED 1

Source Code :

```
package main

import (
    "bufio"
    "fmt"
    "os"
    "strconv"
    "strings"
)

func main() {
    // Baca input dari pengguna
    scanner := bufio.NewScanner(os.Stdin)
    fmt.Println("Masukkan baris pertama (pisahkan dengan spasi):")
    scanner.Scan()
    line1 := scanner.Text()

    fmt.Println("Masukkan baris kedua (pisahkan dengan spasi):")
    scanner.Scan()
    line2 := scanner.Text()

    // Parse input menjadi slice integer tanpa duplikat
    set1 := parseInputToSet(line1)
    set2 := parseInputToSet(line2)

    // Cari irisan antara dua himpunan
    intersection := findIntersection(set1, set2)

    // Tampilkan hasil
    fmt.Println("Hasil Irisan:", intersection)
}
```

```

// Fungsi untuk mengubah input string menjadi map (himpunan unik)
func parseInputToSet(input string) map[int]bool {
    set := make(map[int]bool)
    numbers := strings.Fields(input)
    for _, num := range numbers {
        val, err := strconv.Atoi(num)
        if err == nil {
            if !set[val] {
                set[val] = true
            } else {
                // Jika duplikat ditemukan, hentikan pengisian
                break
            }
        }
    }
    return set
}

// Fungsi untuk menemukan irisan antara dua himpunan
func findIntersection(set1, set2 map[int]bool) []int {
    intersection := []int{}
    for num := range set1 {
        if set2[num] {
            intersection = append(intersection, num)
        }
    }
    return intersection
}

```

### Screenshot Output :

```

Masukkan baris pertama (pisahkan dengan spasi):
1 2
Masukkan baris kedua (pisahkan dengan spasi):
2 3
Hasil Irisan: [2]

```

### Deskripsi Program :

Program di atas adalah implementasi dalam Golang untuk mencari irisan dari dua baris input angka, dengan memastikan tidak ada duplikasi dalam setiap baris. Input diambil dari pengguna, diubah menjadi himpunan unik menggunakan struktur data map, dan prosesnya berhenti jika duplikasi angka terdeteksi dalam satu baris. Irisan antara kedua himpunan dihitung dengan membandingkan elemen-elemen dari kedua map menggunakan fungsi findIntersection. Output berupa daftar angka yang muncul di kedua baris input. Program ini efisien karena memanfaatkan sifat map untuk pencarian cepat, sehingga cocok untuk memproses data dengan elemen unik secara otomatis.

## UNGUIDED II

### Source Code :

```
1 package main
2
3 import (
4     "bufio"
5     "fmt"
6     "os"
7     "strconv"
8     "strings"
9 )
10
11 // Definisi struct mahasiswa
12 type Mahasiswa struct {
13     NIM string
14     Nama string
15     Nilai int
16 }
17
18 // Definisi array mahasiswa dengan kapasitas maksimal 51
19 const nMax = 51
20
21 type ArrayMahasiswa [nMax]Mahasiswa
22
23 func main() {
24     var dataMahasiswa ArrayMahasiswa
25     var n int
26
27     // Input data mahasiswa
28     n = inputMahasiswa(&dataMahasiswa)
29
30     // Cari nilai pertama dan terakhir dari mahasiswa dengan NIM tertentu
31     fmt.Println("Masukkan NIM mahasiswa yang ingin dicari nilainya:")
32     var searchNIM string
33     fmt.Scanln(&searchNIM)
34
35     nilaiPertama, nilaiTerakhir := cariNilai(dataMahasiswa, n, searchNIM)
36     if nilaiPertama != -1 && nilaiTerakhir != -1 {
37         fmt.Printf("Nilai pertama: %d, Nilai terakhir: %d\n", nilaiPertama, nilaiTerakhir)
38     } else {
39         fmt.Println("Mahasiswa dengan NIM tersebut tidak ditemukan.")
40     }
41 }
42
43 // Fungsi untuk menerima masukan data mahasiswa
44 func inputMahasiswa(data *ArrayMahasiswa) int {
45     var n int
46     scanner := bufio.NewScanner(os.Stdin)
```

```

    fmt.Println("Masukkan jumlah data mahasiswa (maksimal 51):")
    fmt.Scanln(&n)

    if n > nMax {
        fmt.Printf("Jumlah mahasiswa tidak boleh lebih dari %d\n", nMax)
        return 0
    }

    for i := 0; i < n; i++ {
        fmt.Printf("Masukkan data mahasiswa ke-%d (format: NIM Nama Nilai):\n", i+1)
        scanner.Scan()
        input := scanner.Text()
        parts := strings.Fields(input)

        if len(parts) != 3 {
            fmt.Println("Format input salah! Harus 3 elemen: NIM Nama Nilai.")
            i--
            continue
        }

        nilai, err := strconv.Atoi(parts[2])
        if err != nil {
            fmt.Println("Nilai harus berupa angka.")
            i--
            continue
        }

        data[i] = Mahasiswa{
            NIM:  parts[0],
            Nama: parts[1],
            Nilai: nilai,
        }
    }

    return n
}

// Fungsi untuk mencari nilai pertama dan terakhir mahasiswa berdasarkan NIM
func cariNilai(data ArrayMahasiswa, n int, searchNIM string) (int, int) {
    var nilaiPertama, nilaiTerakhir int
    nilaiPertama = -1
    nilaiTerakhir = -1

    for i := 0; i < n; i++ {
        if data[i].NIM == searchNIM {
            if nilaiPertama == -1 {

```

```

                nilaiPertama = data[i].Nilai
            }
            nilaiTerakhir = data[i].Nilai
        }
    }

    return nilaiPertama, nilaiTerakhir
}

```

### Screenshot Output :

```
Masukkan jumlah data mahasiswa (maksimal 51):
3
Masukkan data mahasiswa ke-1 (format: NIM Nama Nilai):
343448 dedi 99
Masukkan data mahasiswa ke-2 (format: NIM Nama Nilai):
344424 fafa 88
Masukkan data mahasiswa ke-3 (format: NIM Nama Nilai):
456435 goji 99
Masukkan NIM mahasiswa yang ingin dicari nilainya:
456435
Nilai pertama: 99, Nilai terakhir: 99
```

### Deskripsi Program :

Program di atas adalah implementasi untuk mencatat data mahasiswa menggunakan Golang. Data mahasiswa berupa NIM, Nama, dan Nilai disimpan dalam array dengan kapasitas maksimum 51, menggunakan struktur data struct bernama Mahasiswa. Fungsi inputMahasiswa digunakan untuk menerima input data mahasiswa dari pengguna, memastikan format yang benar, dan menyimpannya ke dalam array. Setelah data dimasukkan, fungsi cariNilai digunakan untuk mencari nilai pertama dan terakhir dari mahasiswa berdasarkan NIM yang dimasukkan pengguna. Jika NIM ditemukan dalam data, program mencetak nilai pertama dan terakhir, sedangkan jika tidak ditemukan, program menampilkan pesan bahwa NIM tersebut tidak ada. Program ini memastikan validasi input dan efisiensi pencarian data, sehingga cocok untuk mengelola data mahasiswa sederhana.

### UNGUIDED III

#### Source Code :

```
fmt.Println("Masukkan jumlah data mahasiswa (maksimal 51):")
fmt.Scanln(&n)

if n > nMax {
    fmt.Printf("Jumlah mahasiswa tidak boleh lebih dari %d\n", nMax)
    return 0
}

for i := 0; i < n; i++ {
    fmt.Printf("Masukkan data mahasiswa ke-%d (format: NIM Nama Nilai):\n", i+1)
    scanner.Scan()
    input := scanner.Text()
    parts := strings.Fields(input)

    if len(parts) != 3 {
        fmt.Println("Format input salah! Harus 3 elemen: NIM Nama Nilai.")
        i--
        continue
    }

    nilai, err := strconv.Atoi(parts[2])
    if err != nil {
        fmt.Println("Nilai harus berupa angka.")
        i--
        continue
    }

    data[i] = Mahasiswa{
        NIM:    parts[0],
        Nama:   parts[1],
        Nilai:  nilai,
    }
}

return n
}

// Fungsi untuk mencari nilai pertama dan terakhir mahasiswa berdasarkan NIM
func cariNilai(data ArrayMahasiswa, n int, searchNIM string) (int, int) {
    var nilaiPertama, nilaiTerakhir int
    nilaiPertama = -1
    nilaiTerakhir = -1

    for i := 0; i < n; i++ {
        if data[i].NIM == searchNIM {
            if nilaiPertama == -1 {
```



## Screenshot Output :

```
1 package main
2
3 import (
4     "bufio"
5     "fmt"
6     "os"
7     "strconv"
8     "strings"
9 )
10
11 // Definisi struct mahasiswa
12 type Mahasiswa struct {
13     NIM string
14     Nama string
15     Nilai int
16 }
17
18 // Definisi array mahasiswa dengan kapasitas maksimal 51
19 const nMax = 51
20
21 type ArrayMahasiswa [nMax]Mahasiswa
22
23 func main() {
24     var dataMahasiswa ArrayMahasiswa
25     var n int
26
27     // Input data mahasiswa
28     n = inputMahasiswa(&dataMahasiswa)
29
30     // Cari nilai pertama dan terakhir dari mahasiswa dengan NIM tertentu
31     fmt.Println("Masukkan NIM mahasiswa yang ingin dicari nilainya:")
32     var searchNIM string
33     fmt.Scanln(&searchNIM)
34
35     nilaiPertama, nilaiTerakhir := cariNilai(dataMahasiswa, n, searchNIM)
36     if nilaiPertama != -1 && nilaiTerakhir != -1 {
37         fmt.Printf("Nilai pertama: %d, Nilai terakhir: %d\n", nilaiPertama, nilaiTerakhir)
38     } else {
39         fmt.Println("Mahasiswa dengan NIM tersebut tidak ditemukan.")
40     }
41 }
42
43 // Fungsi untuk menerima masukan data mahasiswa
44 func inputMahasiswa(data *ArrayMahasiswa) int {
45     var n int
46     scanner := bufio.NewScanner(os.Stdin)
```

## Deskripsi Program :

Program ini membaca data 35 provinsi berupa nama, populasi, dan angka pertumbuhan, serta satu nama provinsi target. Program mencari provinsi dengan angka pertumbuhan tertinggi, indeks provinsi target dalam daftar, dan menghitung prediksi populasi untuk provinsi yang memiliki pertumbuhan lebih dari 2% dalam setahun. Hasilnya menampilkan nama provinsi dengan pertumbuhan tertinggi, indeks provinsi target, dan daftar prediksi populasi tahun depan untuk provinsi yang memenuhi syarat. Program menggunakan struktur data sederhana dan perhitungan berbasis loop untuk menghasilkan output sesuai spesifikasi.



## **UNGUIDED IV**

**Source Code :**

```

fmt.Println("Masukkan jumlah data mahasiswa (maksimal 51):")
fmt.Scanln(&n)

if n > nMax {
    fmt.Printf("Jumlah mahasiswa tidak boleh lebih dari %d\n", nMax)
    return 0
}

for i := 0; i < n; i++ {
    fmt.Printf("Masukkan data mahasiswa ke-%d (format: NIM Nama Nilai):\n", i+1)
    scanner.Scan()
    input := scanner.Text()
    parts := strings.Fields(input)

    if len(parts) != 3 {
        fmt.Println("Format input salah! Harus 3 elemen: NIM Nama Nilai.")
        i--
        continue
    }

    nilai, err := strconv.Atoi(parts[2])
    if err != nil {
        fmt.Println("Nilai harus berupa angka.")
        i--
        continue
    }

    data[i] = Mahasiswa{
        NIM:    parts[0],
        Nama:   parts[1],
        Nilai:  nilai,
    }
}

return n
}

// Fungsi untuk mencari nilai pertama dan terakhir mahasiswa berdasarkan NIM
func cariNilai(data ArrayMahasiswa, n int, searchNIM string) (int, int) {
    var nilaiPertama, nilaiTerakhir int
    nilaiPertama = -1
    nilaiTerakhir = -1

    for i := 0; i < n; i++ {
        if data[i].NIM == searchNIM {
            if nilaiPertama == -1 {

```

### Screenshot Output :

```

02192_modul_11 (guides\guides1.go)
Masukkan baris pertama (pisahkan dengan spasi):
1 2
Masukkan baris kedua (pisahkan dengan spasi):
2 3
Hasil Irisan: [2]

```

### Deskripsi Program :

Program ini membaca data bilangan bulat hingga menemukan angka **-5313541** sebagai penanda akhir input. Setiap kali angka **0** muncul dalam input, program mengurutkan data yang sudah dibaca sebelumnya dan menghitung nilai median. Median dihitung berdasarkan jumlah elemen: jika ganjil, median adalah elemen tengah; jika genap, median adalah rata-rata dua elemen tengah (dibulatkan ke bawah). Setelah median dihitung, program mencetaknya ke layar. Program terus memproses data baru hingga tanda akhir input ditemukan.

**UNGUIDED V**

**Source Code :**

```

1  package main
2
3  import (
4      "bufio"
5      "fmt"
6      "os"
7      "strconv"
8      "strings"
9  )
10
11 // Definisi struct mahasiswa
12 type Mahasiswa struct {
13     NIM    string
14     Nama   string
15     Nilai  int
16 }
17
18 // Definisi array mahasiswa dengan kapasitas maksimal 51
19 const nMax = 51
20
21 type ArrayMahasiswa [nMax]Mahasiswa
22
23 func main() {
24     var dataMahasiswa ArrayMahasiswa
25     var n int
26
27     // Input data mahasiswa
28     n = inputMahasiswa(&dataMahasiswa)
29
30     // Cari nilai pertama dan terakhir dari mahasiswa dengan NIM tertentu
31     fmt.Println("Masukkan NIM mahasiswa yang ingin dicari nilainya:")
32     var searchNIM string
33     fmt.Scanln(&searchNIM)
34
35     nilaiPertama, nilaiTerakhir := cariNilai(dataMahasiswa, n, searchNIM)
36     if nilaiPertama != -1 && nilaiTerakhir != -1 {
37         fmt.Printf("Nilai pertama: %d, Nilai terakhir: %d\n", nilaiPertama, nilaiTerakhir)
38     } else {
39         fmt.Println("Mahasiswa dengan NIM tersebut tidak ditemukan.")
40     }
41 }
42
43 // Fungsi untuk menerima masukan data mahasiswa
44 func inputMahasiswa(data *ArrayMahasiswa) int {
45     var n int
46     scanner := bufio.NewScanner(os.Stdin)
47

```

### Screenshot Output :

```

Masukkan jumlah data mahasiswa (maksimal 51):
3
Masukkan data mahasiswa ke-1 (format: NIM Nama Nilai):
343448 dedi 99
Masukkan data mahasiswa ke-2 (format: NIM Nama Nilai):
344424 fafa 88
Masukkan data mahasiswa ke-3 (format: NIM Nama Nilai):
456435 goji 99
Masukkan NIM mahasiswa yang ingin dicari nilainya:
456435
Nilai pertama: 99, Nilai terakhir: 99

```

### Deskripsi Program :

Program ini membaca data bilangan bulat hingga menemukan angka **-5313541** sebagai penanda akhir input. Setiap kali angka **0** muncul dalam input, program mengurutkan data yang sudah dibaca sebelumnya dan menghitung nilai median. Median dihitung berdasarkan jumlah elemen: jika ganjil, median adalah elemen tengah; jika genap, median adalah rata-rata dua elemen tengah (dibulatkan ke bawah). Setelah median dihitung, program mencetaknya ke layar. Program terus memproses data baru hingga tanda akhir input ditemukan.