

**LAPORAN PRAKTIKUM
ALGORITMA DAN PEMROGRAMAN 2**

MODUL XV

TEST



Disusun Oleh :

PETRA PRIADI S.P GINTING (2311102273)

IF-06

Dosen Pengampu :

ABEDNEGO DWI SEPTIADI

PROGRAM STUDI S1 TEKNIK INFORMATIKA

FAKULTAS INFORMATIKA

TELKOM UNIVERSITY PURWOKERTO

2024

1. Sourcecode

```
package main

import (
    "fmt"
)

type set [2022]int

// Fungsi ada untuk mengecek apakah elemen val ada di
// dalam himpunan T.
func ada(T set, n int, val int) bool {
    for i := 0; i < n; i++ {
        if T[i] == val {
            return true
        }
    }
    return false
}

// Fungsi inputSet untuk menerima masukan hingga ada
// elemen duplikat atau himpunan penuh.
func inputSet(T *set, n *int) {
    var val int
    *n = 0
    seen := make(map[int]bool) // Untuk mengecek
    duplikat dengan cepat

    for {
        fmt.Scan(&val)
        if seen[val] {
            break
        }
        if *n >= len(T) {
            break
        }
        T[*n] = val
        *n++
        seen[val] = true
    }
}

// Fungsi temukanIrisan untuk mencari irisan antara dua
// himpunan T1 dan T2.
func temukanIrisan(T1, T2 set, n1, n2 int, T3 *set, n3
*int) {
    *n3 = 0
    for i := 0; i < n1; i++ {
```

```

        if ada(T2, n2, T1[i]) && !ada(*T3, *n3, T1[i])
        {
            T3[*n3] = T1[i]
            (*n3)++
        }
    }

    // Fungsi printSet untuk mencetak elemen dalam himpunan.
    func printSet(T set, n int) {
        for i := 0; i < n; i++ {
            fmt.Print(T[i], " ")
        }
        fmt.Println()
    }

    func main() {
        var s2311102273, s2, s3 set
        var n1, n2, n3 int

        // Memasukkan elemen untuk himpunan pertama
        fmt.Println("Masukkan elemen untuk himpunan pertama:")
        inputSet(&s2311102273, &n1)

        // Memasukkan elemen untuk himpunan kedua
        fmt.Println("Masukkan elemen untuk himpunan kedua:")
        inputSet(&s2, &n2)

        // Menemukan irisan dari himpunan pertama dan kedua
        temukanIrisan(s2311102273, s2, n1, n2, &s3, &n3)

        // Mencetak irisan himpunan
        fmt.Println("Irisan dari kedua himpunan adalah:")
        printSet(s3, n3)
    }

```

Screenshoot Output

```

PS C:\Users\Lenovo> go run "d:\Pemrograman\Pemrograman Semester 3\2311102273_Petra Priadi S.P Ginting_Praktikum\Modul15Test\nol.go"
Masukkan elemen untuk himpunan pertama:
11 28 33 64 95 16 100 15 64 3 11 7 28 33 6 28
Masukkan elemen untuk himpunan kedua :
11 28 33
Irisan dari kedua himpunan adalah:
11 28 33
PS C:\Users\Lenovo>

```

2. Sourcecode

```
package main

import "fmt"

const MAX_SISWA = 51
const MAX_NILAI = 10

type Student struct {
    NIM    string
    Nama   string
    Nilai []int
}

func main() {
    var students [MAX_SISWA]Student
    var hitungSiswa int

    // memasukkan jumlah siswa
    fmt.Print("Masukkan jumlah mahasiswa: ")
    fmt.Scan(&hitungSiswa)

    // memasukkan detail siswa
    for i := 0; i < hitungSiswa; i++ {
        fmt.Printf("\nMahasiswa ke-%d\n", i+1)
        fmt.Print("NIM: ")
        fmt.Scan(&students[i].NIM)
```

```

        fmt.Print("Nama: ")

        fmt.Scan(&students[i].Nama)

        // memasukkan nilai lebih untuk setiap siswa
        var jumlahNilai int

        fmt.Print("Masukkan jumlah nilai: ")

        fmt.Scan(&jumlahNilai)

        students[i].Nilai = make([]int, jumlahNilai)
        for j := 0; j < jumlahNilai; j++ {
            fmt.Printf("Masukkan Nilai ke-%d: ",
j+1)

            fmt.Scan(&students[i].Nilai[j])
        }
    }

    // mencari siswa
    var cariNIM_2311102273 string

    fmt.Print("\nMasukkan NIM yang dicari: ")

    fmt.Scan(&cariNIM_2311102273)

    // menemukan nilai pertama dan tertinggi
    var menemukanSiswa []Student

    // Cari semua siswa dengan NIM yang sama
    for i := 0; i < hitungSiswa; i++ {
        if students[i].NIM == cariNIM_2311102273 {

```

```

                                menemukanSiswa = append(menemukanSiswa,
students[i])

                                }

                                }

                                // Tampilkan hasil pencarian
                                if len(menemukanSiswa) > 0 {

                                        fmt.Printf("Ditemukan %d mahasiswa dengan NIM
%s:\n", len(menemukanSiswa), cariNIM_2311102273)

                                        for _, student := range menemukanSiswa {

                                                fmt.Printf("\nNama: %s\n", student>Nama)

                                                // Tampilkan semua nilai
                                                fmt.Println("Daftar Nilai:")

                                                for j, grade := range student.Nilai {

                                                        fmt.Printf("Nilai ke-%d: %d\n",
j+1, grade)

                                                        }

                                                // Cari nilai pertama dan tertinggi
                                                if len(student.Nilai) > 0 {

                                                        nilaiPertama := student.Nilai[0]

                                                        nilaiTertinggi :=
findHighestGrade(student.Nilai)

                                                        fmt.Printf("Nilai pertama: %d\n",
nilaiPertama)

                                                        fmt.Printf("Nilai tertinggi:
%d\n", nilaiTertinggi)

```

```

    }

    }

    } else {

        fmt.Printf("Mahasiswa dengan NIM %s tidak
ditemukan\n", cariNIM_2311102273)

    }

}

// Fungsi untuk mencari nilai tertinggi
func findHighestGrade(nilai []int) int {

    if len(nilai) == 0 {

        return -1

    }

    tertinggi := nilai[0]

    for _, grade := range nilai {

        if grade > tertinggi {

            tertinggi = grade

        }

    }

    return tertinggi

}

```

Screenshoot Output

```
Mahasiswa ke-1
NIM: 114
Nama: Nana
Masukkan jumlah nilai: 1
Masukkan Nilai ke-1: 97

Mahasiswa ke-2
NIM: 113
Nama: Jojo
Masukkan jumlah nilai: 3
Masukkan Nilai ke-1: 70
Masukkan Nilai ke-2: 50
Masukkan Nilai ke-3: 80

Mahasiswa ke-3
NIM: 118
Nama: Rere
Masukkan jumlah nilai: 2
Masukkan Nilai ke-1: 88
Masukkan Nilai ke-2: 88

Mahasiswa ke-4
NIM: 116
Nama: Koko
Masukkan jumlah nilai: 2
Masukkan Nilai ke-1: 40
Masukkan Nilai ke-2: 60

Mahasiswa ke-5
NIM: 117
Nama: Keke
Masukkan jumlah nilai: 1
Masukkan Nilai ke-1: 90

Mahasiswa ke-6
NIM: 119
Nama: Roro
Masukkan jumlah nilai: 1
Masukkan Nilai ke-1: 100

Masukkan NIM yang dicari: 114
Ditemukan 1 mahasiswa dengan NIM 114:

Nama: Nana
Daftar Nilai:
Nilai ke-1: 97
Nilai pertama: 97
Nilai tertinggi: 97
PS C:\Users\Lenovo>
```

3. Sourcecode

```
package main

import (
    "fmt"
)

const nProvinsi = 5

type (
    NamaProvinsi    [nProvinsi]string
    PopProvinsi     [nProvinsi]int
    TumbuhProvinsi  [nProvinsi]float64
)

// InputData: Membaca data nama provinsi, populasi, dan
// angka pertumbuhan dari pengguna
```



```

func InputData(prov *NamaProvinsi, pop_2311102273
*PopProvinsi, tumbuh *TumbuhProvinsi, namaCari *string)
{
    for i := 0; i < nProvinsi; i++ {
        fmt.Printf("Masukkan nama provinsi ke-%d: ",
i+1)

        fmt.Scanln(&prov[i])
        fmt.Printf("Masukkan populasi provinsi ke-%d:
", i+1)

        fmt.Scanln(&pop_2311102273[i])
        fmt.Printf("Masukkan angka pertumbuhan
provinsi ke-%d: ", i+1)
        fmt.Scanln(&tumbuh[i])
    }
    fmt.Printf("Masukkan nama provinsi yang ingin
dicari: ")
    fmt.Scanln(namaCari)
}

// ProvinsiTercepat: Menemukan indeks provinsi dengan
angka pertumbuhan tertinggi
func ProvinsiTercepat(tumbuh TumbuhProvinsi) int {
    maxIndex := 0
    for i := 1; i < nProvinsi; i++ {
        if tumbuh[i] > tumbuh[maxIndex] {
            maxIndex = i
        }
    }
    return maxIndex
}

// IndeksProvinsi: Mengembalikan indeks provinsi
berdasarkan nama yang dicari
func IndeksProvinsi(prov NamaProvinsi, nama string) int
{
    for i := 0; i < nProvinsi; i++ {
        if prov[i] == nama {
            return i
        }
    }
    return -1
}

// Prediksi: Menampilkan prediksi jumlah penduduk untuk
provinsi dengan pertumbuhan > 2%
func Prediksi(prov NamaProvinsi, pop_2311102273
PopProvinsi, tumbuh TumbuhProvinsi) {
    fmt.Println("Prediksi jumlah penduduk tahun depan
untuk provinsi dengan pertumbuhan di atas 2%:")
}

```

```

        for i := 0; i < nProvinsi; i++ {
            if tumbuh[i] > 2 {
                prediksi := float64(pop_2311102273[i]) *
(1 + tumbuh[i]/100)
                fmt.Printf("%s: %.0f\n", prov[i],
prediksi)
            }
        }
    }

func main() {
    var prov NamaProvinsi
    var pop_2311102273 PopProvinsi
    var tumbuh TumbuhProvinsi
    var namaCari string

    // Memasukkan data provinsi
    InputData(&prov, &pop_2311102273, &tumbuh,
&namaCari)

    // Menemukan provinsi dengan angka pertumbuhan
tercepat
    tercepat := ProvinsiTercepat(tumbuh)
    fmt.Printf("Provinsi dengan pertumbuhan tercepat:
%s\n", prov[tercepat])

    // Mencari indeks provinsi yang diinputkan oleh
pengguna
    indeksCari := IndeksProvinsi(prov, namaCari)
    if indeksCari == -1 {
        fmt.Printf("Provinsi dengan nama '%s' tidak
ditemukan.\n", namaCari)
    } else {
        fmt.Printf("Indeks provinsi '%s': %d\n",
namaCari, indeksCari)
    }

    // Menampilkan prediksi jumlah penduduk
    Prediksi(prov, pop_2311102273, tumbuh)
}

```

Screenshoot Output

```

PS C:\Users\Lenovo> go run "d:\Pemrograman\Pemrograman Semester 3\2311102273_Petra Priadi S.P Ginting_Praktikum\Modul15Test\n03.go"
Masukkan nama provinsi ke-1: Aceh
Masukkan populasi provinsi ke-1: 2
Masukkan angka pertumbuhan provinsi ke-1: 10
Masukkan nama provinsi ke-2: Sumut
Masukkan populasi provinsi ke-2: 27
Masukkan angka pertumbuhan provinsi ke-2: 1
Masukkan nama provinsi ke-3: Sumbar
Masukkan populasi provinsi ke-3: 18
Masukkan angka pertumbuhan provinsi ke-3: 6
Masukkan nama provinsi ke-4: Jambi
Masukkan populasi provinsi ke-4: 19
Masukkan angka pertumbuhan provinsi ke-4: 3
Masukkan nama provinsi ke-5: Sultan
Masukkan populasi provinsi ke-5: 20
Masukkan angka pertumbuhan provinsi ke-5: 3
Masukkan nama provinsi yang ingin dicari: Aceh
Provinsi dengan pertumbuhan tercepat: Aceh
Indeks provinsi 'Aceh': 0
Prediksi jumlah penduduk tahun depan untuk provinsi dengan pertumbuhan di atas 2%:
Aceh: 2
Sumbar: 19
Jambi: 20
Sultan: 21
PS C:\Users\Lenovo>

```

4. Sourcecode

```

package main

import (
    "fmt"
)

// Definisi tipe khusus untuk daftar bilangan
type IntArray []int

// Method untuk menyisipkan angka_2311102273 ke dalam
array (insertion sort)
func (arr *IntArray) insertAndSort(num int) {
    *arr = append(*arr, num) // Tambahkan
    angka_2311102273 ke array
    for i := 1; i < len(*arr); i++ {
        key := (*arr)[i]
        j := i - 1

        // Geser elemen ke kanan untuk menjaga urutan
        for j >= 0 && (*arr)[j] > key {
            (*arr)[j+1] = (*arr)[j]
            j--
        }
        (*arr)[j+1] = key
    }
}

// Method untuk menghitung median dari IntArray
func (arr IntArray) median() int {
    n := len(arr)
    if n%2 == 1 {
        // Jika jumlah elemen ganjil

```

```

        return arr[n/2]
    }
    // Jika jumlah elemen genap
    return (arr[n/2-1] + arr[n/2]) / 2
}

func main() {
    var angka_2311102273 IntArray
    var input int

    fmt.Println("Masukkan bilangan satu per satu (0
    untuk menghitung median, -5313 untuk keluar):")

    for {
        _, err := fmt.Scan(&input)
        if err != nil {
            fmt.Println("Error membaca input:", err)
            break
        }

        if input == -5313 {
            // Keluar dari program
            break
        }

        if input == 0 {
            // Cetak median jika angka_2311102273 0
            if len(angka_2311102273) > 0 {
                fmt.Println("Median saat ini:",
                angka_2311102273.median())
            }
            continue
        }

        // Tambahkan angka_2311102273 ke array dan
        sortir
        angka_2311102273.insertAndSort(input)
    }
}

```

Screenshoot Output

```

PS C:\Users\Lenovo> go run "d:\Pemrograman\Pemrograman Semester 3\2311102273 Petra Priadi S.P Ginting_Praktikum\Modul15Test\n04.go"
Masukkan bilangan satu per satu (0 untuk menghitung median, -5313 untuk keluar):
7 23 11 0 5 19 2 29 3 13 17 0 -5313
Median saat ini: 11
Median saat ini: 12
PS C:\Users\Lenovo>

```

5. Sourcecode

```
package main

import (
    "fmt"
)

const NMAX = 1000000

// Struct untuk menyimpan nama partai dan jumlah suara
type partai struct {
    nama    int
    suara   int
}

// Array of partai dengan kapasitas maksimum
type tabPartai [NMAX]partai

func posisi(t tabPartai, n int, nama int) int {
    // Mencari posisi partai berdasarkan nama dengan
    pencarian sekuensial
    for i := 0; i < n; i++ {
        if t[i].nama == nama {
            return i
        }
    }
    return -1 // Mengembalikan -1 jika tidak ditemukan
}

func insertionSortDescending(t *tabPartai, n int) {
    // Proses pengurutan dengan insertion sort secara
    descending berdasarkan jumlah suara
    for i := 1; i < n; i++ {
        key := t[i]
        j := i - 1
        for j >= 0 && t[j].suara < key.suara {
            t[j+1] = t[j]
            j--
        }
        t[j+1] = key
    }
}

func main() {
    var p_2311102273 tabPartai
    var n int // Jumlah partai yang terdata

    fmt.Println("Masukkan suara partai (akhiri dengan -
1):")
}
```

```

        for {
            var nama int
            fmt.Scan(&nama)
            if nama == -1 {
                break
            }

            pos := posisi(p_2311102273, n, nama)
            if pos == -1 {
                // Tambahkan partai baru jika belum ada
                p_2311102273[n] = partai{nama: nama,
suara: 1}
                n++
            } else {
                // Tambahkan suara ke partai yang sudah
ada
                p_2311102273[pos].suara++
            }
        }

        // Urutkan array p_2311102273 berdasarkan suara
        secara descending
        insertionSortDescending(&p_2311102273, n)

        // Tampilkan hasil
        fmt.Println("Hasil perolehan suara:")
        for i := 0; i < n; i++ {
            fmt.Printf("%d(%d)\n", p_2311102273[i].nama,
p_2311102273[i].suara)
        }
    }
}

```

Screenshoot Output

```

PS C:\Users\Lenovo> go run "d:\Pemrograman\Pemrograman Semester 3\2311102273_Petra Priadi S.P Ginting_Praktikum\Modul15Test\n05.go"
Masukkan suara partai (akhiri dengan -1):
100
20
26
36
27
-1
Hasil perolehan suara:
100(1)
20(1)
26(1)
36(1)
27(1)
PS C:\Users\Lenovo>

```