

**LAPORAN PRAKTIKUM
ALGORITMA DAN PEMROGRAMAN 2**

MODUL 11

PENCARIAN NILAI EKSTRIM PADA HIMPUNAN DATA



Disusun Oleh :

Yoga Hogantara / 2311102153

IF-11-05

Dosen Pengampu :

Arif Amrulloh, S.Kom., M.Kom

PROGRAM STUDI S1 TEKNIK INFORMATIKA

FAKULTAS INFORMATIKA

TELKOM UNIVERSITY PURWOKERTO

2024

I. DASAR TEORI

Pencarian merupakan suatu proses menemukan suatu yang ingin di ketahui. Pada modul ini membahas pencarian nilai max/min pada sekumpulan data yang biasa disebut pencarian nilai ekstrim.

Ide algoritma ini sederhana, karena diproses menggunakan sequential, maka nilai atau indeks ke nilai maksimum dari data yang telah diproses disimpan untuk membandingkan dengan data berikutnya. Nilai yang berhasil disimpan sampai algoritma tersebut berakhir adalah nilai maksimum yang dicari.

II. UNGUIDED

1. Sourcecode

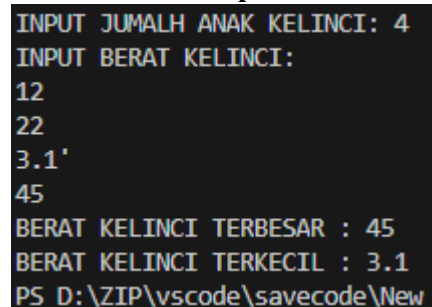
```
package main
import (
    "fmt"
)

func main() {
    var n int
    fmt.Print("INPUT JUMALH ANAK KELINCI: ")
    fmt.Scan(&n)
    if n > 1000{
        fmt.Println("MAKS 1000")
        return
    }
    berat:=make([]float64,n)

    fmt.Println("INPUT BERAT KELINCI:")
    for i:=0;i<n;i++){
        fmt.Scan(&berat[i])
    }
    max,min:=berat[0],berat[0]

    for i := 0; i < n; i++ {
        if berat[i]<min{
            min=berat[i]
        }
        if berat[i]>max{
            max=berat[i]
        }
    }
    fmt.Println("BERAT KELINCI TERBESAR :",max)
    fmt.Println("BERAT KELINCI TERKECIL :",min)
}
```

Screenshoot Output



```
INPUT JUMALH ANAK KELINCI: 4
INPUT BERAT KELINCI:
12
22
3.1'
45
BERAT KELINCI TERBESAR : 45
BERAT KELINCI TERKECIL : 3.1
PS D:\ZIP\vscode\savecode\New
```

Deskripsi Program

Program di atas berfungsi untuk mencari sebuah nilai maksimum / berat tertinggi dari kelinci dan berat minimum dari berat kelinci yang diinputkan oleh user. Dengan menggunakan array untuk menyimpan data dan kondisi if untuk maksimal jumlah kelinci yang diinputkan. Pada program terdapat perulangan for untuk pencarian nilai terbesar dan terkecil dari berat yang sudah diinputkan sebelumnya dan pada akhirnya akan mengoutputkan hasil berat maksimum dan minimum yang sudah ditemukan.

2. Source code

```
package main
import (
    "fmt"
)

func main() {
    var x, y int
    fmt.Print("INPUT JUMLAH WADAH: ")
    fmt.Scan(&x)
    fmt.Print("INPUT JUMLAH IKAN/WADAH: ")
    fmt.Scan(&y)

    berat := make([][]float64, x)

    for i := 0; i < x; i++ {
        fmt.Printf("Masukkan berat ikan untuk wadah ke-%d:\n", i+1)
        berat[i] = make([]float64, y)
        for j := 0; j < y; j++ {
            fmt.Scan(&berat[i][j])
        }
    }
    for i := 0; i < x; i++ {
        var jumlah float64
        for j := 0; j < y; j++ {
            jumlah += berat[i][j]
        }
        rata2 := jumlah / float64(y)
        fmt.Printf("Wadah %d: jumlah total = %.2f, rata-rata = %.2f\n", i+1, jumlah, rata2)
    }
}
```

Screenshoot Output

```
INPUT JUMLAH WADAH: 2
INPUT JUMLAH IKAN/WADAH: 3
Masukkan berat ikan untuk wadah ke-1:
1.2
2
3
Masukkan berat ikan untuk wadah ke-2:
2
4
1.1
Wadah 1: jumlah total = 6.20, rata-rata = 2.07
Wadah 2: jumlah total = 7.10, rata-rata = 2.37
PS D:\ZIP\vscode\savecode\New folder>
```

Deskripsi Program

Pada program tersebut berfungsi untuk menghitung jumlah dan rata rata berat ikan yang berada dalam wadah yang di inginkan. Program tersebut meminta user untuk meng inputkan jumlah wadah, jumlah ikan, dan berat ikan yang ingin diinputkan. Program tersebut meggunakan perulangan for untuk menginputkan berat ikan pada wadah dan juga untuk menghitung jumlah ikan dan rata rata berat ikan dari inputan. Output hasil pada program akan menampilkan total dan rata rata berat ikan.

3. Source code

```
package main
import "fmt"

type arrBalita [100]float64
func hitungMinMax(arrBerat arrBalita, n int, bMin, bMax
*float64) {
    *bMin = arrBerat[0]
    *bMax = arrBerat[0]
    for i := 1; i < n; i++ {
        if arrBerat[i] < *bMin {
            *bMin = arrBerat[i]
        }
        if arrBerat[i] > *bMax {
            *bMax = arrBerat[i]
        }
    }
}

func rerata(arrBerat arrBalita, n int) float64 {
    var total float64
    for i := 0; i < n; i++ {
        total += arrBerat[i]
    }
    return total / float64(n)
}

func main() {
```

```

var n int
var arrBerat arrBalita
var bMin, bMax float64

fmt.Print("Masukkan banyak data berat balita: ")
fmt.Scan(&n)
for i := 0; i < n; i++ {
    fmt.Printf("Masukkan berat balita ke-%d: ",
i+1)
    fmt.Scan(&arrBerat[i])
}

hitungMinMax(arrBerat, n, &bMin, &bMax)
rata := rerata(arrBerat, n)

fmt.Printf("Berat balita minimum: %.2f kg\n", bMin)
fmt.Printf("Berat balita maksimum: %.2f kg\n",
bMax)
fmt.Printf("Rerata berat balita: %.2f kg\n", rata)
}

```

Screenshoot Output

```

Masukkan banyak data berat balita: 4
Masukkan berat balita ke-1: 5.3
Masukkan berat balita ke-2: 6.2
Masukkan berat balita ke-3: 4.1
Masukkan berat balita ke-4: 9.9
Berat balita minimum: 4.10 kg
Berat balita maksimum: 9.90 kg
Rerata berat balita: 6.38 kg
PS D:\ZIP\vscode\savecode\New folder>

```

Deskripsi Program

Program tersebut berfungsi untuk menghitung berat balita yang di inputkan oleh user dengan menggunakan perulangan for untuk menginputkan berat balitanya. Program tersebut menggunakan beberapa fungsi untuk menghitung berat tertinggi dan terkecil, rata rata berat balita dari inputan. Pada fungsi htiungMinMax menggunakan perulangan for untuk menghitung berat nya, pada fungsi rerata menggunakan perulangan for juga untuk menghitung rata rata berat balita.