

LAPORAN PRAKTIKUM
ALGORITMA DAN PEMROGRAMAN 2
MODUL 10
PENCARIAN NILAI EKSTRIM PADA HIMPUNAN DATA



Oleh :

FAHRUR RIZQI

2311102059

S1 IF-11-02

S1 TEKNIK INFORMATIKA
TELKOM UNIVERSITY PURWOKERTO
2024

I. DASAR TEORI

Modul 10 membahas tentang pencarian nilai ekstrim pada himpunan data. Pencarian nilai ekstrim adalah suatu proses yang lazim dilakukan dalam kehidupan sehari-hari, seperti mencari nilai terkecil atau terbesar pada sekumpulan data. Algoritma sederhana untuk pencarian nilai ekstrim adalah dengan memproses data secara sekuensial dan menyimpan nilai maksimum atau minimum yang telah diproses untuk dibandingkan dengan data berikutnya. Nilai yang berhasil disimpan sampai algoritma tersebut berakhir adalah nilai maksimum atau minimum yang dicari.

Algoritma tersebut secara umum dapat dijelaskan sebagai berikut:

1. Jadikan data pertama sebagai nilai ekstrim.
2. Lakukan validasi nilai ekstrim dari data kedua hingga data terakhir. Jika nilai ekstrim tidak valid, maka update nilai ekstrim tersebut dengan data yang dicek.
3. Apabila semua data telah dicek, maka nilai ekstrim yang dimiliki adalah valid.

Selain itu, modul ini juga membahas pencarian nilai ekstrim pada array bertipe data dasar dan bertipe data terstruktur. Misalnya, terdapat contoh potongan program dalam bahasa Go untuk mencari nilai terkecil pada array bertipe data dasar dan mencari indeks nilai terkecil pada array bertipe data dasar. Selain itu, terdapat juga contoh pencarian nilai ekstrim pada array bertipe data terstruktur, seperti mencari data mahasiswa dengan nilai terbesar dan mencari indeks mahasiswa dengan nilai terbesar.

Modul ini juga menyajikan soal latihan yang melibatkan pencarian nilai ekstrim pada tipe data dasar maupun terstruktur, seperti mencari berat anak kelinci terkecil dan terbesar, mencari berat ikan terkecil dan terbesar, serta mencari berat balita terkecil, terbesar, dan reratanya.

Kesimpulan

Dari modul yang dibahas tentang pencarian nilai ekstrim pada himpunan data, terdapat beberapa konsep dasar yang perlu dipahami. Pertama, pencarian nilai ekstrim adalah proses umum yang dilakukan dalam kehidupan sehari-hari, seperti mencari nilai terkecil atau terbesar dalam sekumpulan data. Algoritma sederhana untuk pencarian nilai ekstrim melibatkan proses pemrosesan data secara sekuensial

dan penyimpanan nilai maksimum atau minimum yang telah diproses untuk dibandingkan dengan data berikutnya. Nilai yang berhasil disimpan hingga algoritma selesai adalah nilai maksimum atau minimum yang dicari

II. GUIDED

I. Guided 1

Sourcode

```
package main

import (
    "fmt"
)

func main() {
    var N int
    var berat [1000]float64

    fmt.Print("Masukan jumlah anak kelinci: ")
    fmt.Scan(&N)

    fmt.Println("Masukan berat anak kelinci: ")
    for i := 0; i < N; i++ {
        fmt.Scan(&berat[i])
    }

    min := berat[0]
    max := berat[0]

    for i := 1; i < N; i++ {
        if berat[i] < min {
            min = berat[i]
        }
        if berat[i] > max {
            max = berat[i]
        }
    }
```

```
}

    fmt.Printf("Berat terkecil: %.2f\n", min)
    fmt.Printf("Berat terbesar: %.2f\n", max)
}
```

Output :

```
PS C:\Users\ASUS\Documents\File Fahrur\SEMESTER 3\PRAKTIKUM ALPRO 2\MODUL 10>
go run "c:\Users\ASUS\Documents\File Fahrur\SEMESTER 3\PRAKTIKUM ALPRO 2\MODUL 10>
Masukan jumlah anak kelinci: 4
Masukan berat anak kelinci:
4.2
2.1
4.9
2.7
Berat terkecil: 2.10
Berat terbesar: 4.90
PS C:\Users\ASUS\Documents\File Fahrur\SEMESTER 3\PRAKTIKUM ALPRO 2\MODUL 10>
```

Deskripsi Program

Kode tersebut untuk menentukan berat terkecil dan terbesar dari sejumlah anak kelinci.

Berikut adalah penjelasan singkat setiap bagian:

1. Program meminta input jumlah anak kelinci (**N**) dan berat masing-masing anak kelinci yang disimpan dalam array **berat**.
2. Program menginisialisasi nilai **min** dan **max** dengan berat anak kelinci pertama.
3. Program kemudian melakukan iterasi untuk memeriksa setiap berat anak kelinci lainnya:
 - Jika berat lebih kecil dari **min**, nilai **min** diperbarui.
 - Jika berat lebih besar dari **max**, nilai **max** diperbarui.
4. Terakhir, program mencetak berat terkecil (**min**) dan terbesar (**max**) dengan format dua angka di belakang koma.

Program ini berguna untuk menentukan rentang berat anak kelinci dari input yang diberikan.

II. Guided 2

Sourcode

```
package main

import (
    "fmt"
)

func main() {
    var x, y int
    fmt.Print("Masukkan jumlah ikan dan kapasitas wadah: ")
    fmt.Scan(&x, &y)

    berat := make([]float64, x)
    fmt.Println("Masukkan berat tiap ikan:")
    for i := 0; i < x; i++ {
        fmt.Scan(&berat[i])
    }

    jumlahWadah := (x + y - 1) / y // pembulatan ke atas jika x tidak habis dibagi y
    totalBeratWadah := make([]float64, jumlahWadah)

    for i := 0; i < x; i++ {
        indeksWadah := i / y
        totalBeratWadah[indeksWadah] += berat[i]
    }

    // Output total berat tiap wadah
    fmt.Println("Total berat tiap wadah:")
}
```

```

for _, total := range totalBeratWadah {
    fmt.Printf("%.2f ", total)
}
fmt.Println()

// Output rata-rata berat tiap wadah
fmt.Println("Rata-rata berat tiap wadah:")
for _, total := range totalBeratWadah {
    rataRata := total / float64(y)
    fmt.Printf("%.2f ", rataRata)
}
fmt.Println()
}

```

Output :

```

PS C:\Users\ASUS\Documents\File Fahrur\SEMESTER 3\PRAKTIKUM ALPRO 2\MODUL 10> go
Fahrur\SEMESTER 3\PRAKTIKUM ALPRO 2\MODUL 10\tempCodeRunnerFile.go"
Masukkan jumlah ikan dan kapasitas wadah: 4 3
Masukkan berat tiap ikan:
1.4
5.2
4
3.1
Total berat tiap wadah:
10.60 3.10
Rata-rata berat tiap wadah:
3.53 1.03
PS C:\Users\ASUS\Documents\File Fahrur\SEMESTER 3\PRAKTIKUM ALPRO 2\MODUL 10>

```

Deskripsi Program

Program ini untuk membagi sejumlah ikan ke dalam beberapa wadah, menghitung total berat tiap wadah, dan menghitung rata-rata berat tiap wadah. Berikut penjelasan singkat:

1. Program meminta input jumlah ikan (**x**) dan kapasitas maksimal ikan per wadah (**y**).
2. Berat masing-masing ikan dimasukkan ke dalam array **berat**.
3. Program menghitung jumlah wadah yang diperlukan, dengan pembulatan ke atas jika

jumlah ikan tidak habis dibagi kapasitas.

4. Program mendistribusikan berat ikan ke setiap wadah dan menyimpan total berat tiap wadah dalam array `totalBeratWadah`.

5. Program mencetak total berat dan rata-rata berat tiap wadah. Rata-rata dihitung dengan membagi total berat tiap wadah dengan kapasitas wadah.

Program ini berguna untuk membagi ikan secara proporsional ke beberapa wadah sesuai kapasitas yang ditentukan.

III. UNGUIDED

a. Unguided

Sourcode

```
package main

import (
    "fmt"
)

type arrBalita [100]float64

func hitungMinMax(arrBerat arrBalita, n int, bMin, bMax *float64) {
    *bMin = arrBerat[0]
    //fahrur059
    *bMax = arrBerat[0]

    for i := 1; i < n; i++ {
        if arrBerat[i] < *bMin {
            *bMin = arrBerat[i]
        }
        if arrBerat[i] > *bMax {
            *bMax = arrBerat[i]
        }
    }
}

func rataRata(arrBerat arrBalita, n int) float64 {
    var total float64
    //fahrur059
    for i := 0; i < n; i++ {
        total += arrBerat[i]
```

```

    }
    return total / float64(n)
}

func main() {
    var n int
    var berat arrBalita
    var bMin, bMax float64

    fmt.Print("Masukan banyak data berat balita: ")
    fmt.Scan(&n)

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

    hitungMinMax(berat, n, &bMin, &bMax)
    //fahrur059
    rata := rataRata(berat, n)

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

```

Output :

```
PS C:\Users\ASUS\Documents\File Fahrur\SEMESTER 3\PRAKTIKUM ALPRO 2\MODUL 10> go
Fahrur\SEMESTER 3\PRAKTIKUM ALPRO 2\MODUL 10\tempCodeRunnerFile.go"
Masukan banyak data berat balita: 4
Masukan berat balita ke-1: 3.3
Masukan berat balita ke-2: 6.2
Masukan berat balita ke-3: 4.1
Masukan berat balita ke-4: 9.9
Berat balita minimum: 3.30 kg
Berat balita maksimum: 9.90 kg
Rata-rata berat balita: 5.88 kg
PS C:\Users\ASUS\Documents\File Fahrur\SEMESTER 3\PRAKTIKUM ALPRO 2\MODUL 10>
```

Deskripsi Program

Program ini untuk menghitung berat minimum, maksimum, dan rata-rata dari data berat beberapa balita. Berikut penjelasan singkatnya:

1. Program mendefinisikan tipe data `arrBalita` sebagai array `float64` dengan kapasitas 100 elemen.
2. Fungsi `hitungMinMax` menerima array berat balita (`arrBerat`) dan jumlah datanya (`n`), lalu mengisi nilai minimum (`bMin`) dan maksimum (`bMax`) dengan berat terkecil dan terbesar.
3. Fungsi `rataRata` menghitung rata-rata berat balita dengan menjumlahkan semua elemen di `arrBerat` dan membaginya dengan jumlah balita (`n`).
4. Fungsi `main` meminta input jumlah balita dan berat masing-masing balita, kemudian memanggil `hitungMinMax` dan `rataRata` untuk menghitung serta mencetak berat minimum, maksimum, dan rata-rata.

Program ini membantu menghitung statistik berat dari data balita yang diinputkan.