

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



Oleh:
FAJAR FARIZQI AZMI
2311102192
IF-11-02
S1 TEKNIK INFORMATIKA
TELKOM UNIVERSITI PURWOKERTO
2024

I. Dasar Teori

Pada modul 10 ini kita sudah 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 deretannya

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 di cari.

II. Guided

Guided 1

Source code :

```
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)

    }

```

Screenshot output :

```

PS D:\semester3\Alpro 2 materi dan praktikum\Fajar Farizqi Azmi_2311102192_modul 10> go run "d:\semester3\Alpro 2 materi dan praktikum\Fajar Farizqi Azmi_2311102192_modul 10\Guided\Guided1.go"
Masukan jumlah anak kelinci: 7
Masukan berat anak kelinci:
3.2
4.5
4.3
5.4
6.5
7.6
8.7
Berat terkecil: 3.20
Berat terbesar: 8.70
PS D:\semester3\Alpro 2 materi dan praktikum\Fajar Farizqi Azmi_2311102192_modul 10>

```

Deskripsi program :

Program di atas adalah sebuah program berbasis Go yang digunakan untuk menentukan berat terkecil dan berat terbesar dari sejumlah anak kelinci berdasarkan input berat mereka.

Deklarasi variabel nya :

2. N: Variabel integer untuk menyimpan jumlah anak kelinci.
1. berat [1000]float64: Array dengan kapasitas 1000 elemen bertipe float64 untuk menyimpan berat masing-masing anak kelinci

Guided 2

Source code :

```

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()

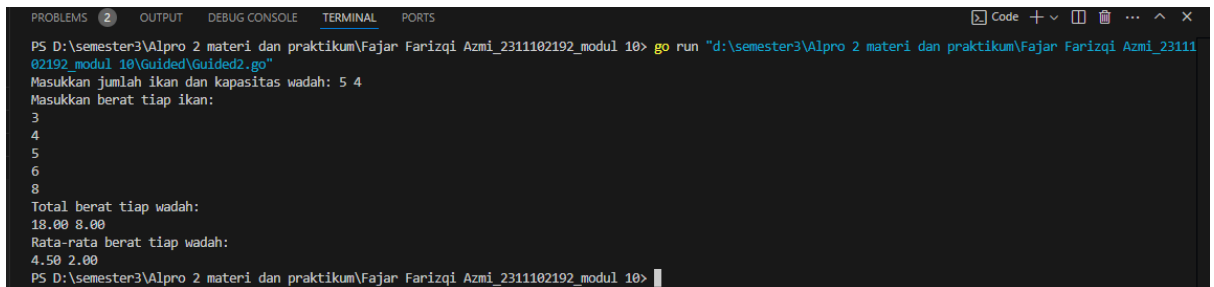
    //Output rata-rata berat tiap wadah
    fmt.Println("Rata-rata berat tiap wadah: ")
    for _, total := range totalBeratWadah {
        rataRata := total / float64(y)
    }
}

```

```
    fmt.Printf("%.2f ", rataRata)
}

fmt.Println()
}
```

Screenshot output :



```
PS D:\semester3\Alpro 2 materi dan praktikum\Fajar Farizqi Azmi_2311102192_modul 10> go run "d:\semester3\Alpro 2 materi dan praktikum\Fajar Farizqi Azmi_2311102192_modul 10\Guided\Guided2.go"
Masukkan jumlah ikan dan kapasitas wadah: 5 4
Masukkan berat tiap ikan:
3
4
5
6
8
Total berat tiap wadah:
18.00 8.00
Rata-rata berat tiap wadah:
4.50 2.00
PS D:\semester3\Alpro 2 materi dan praktikum\Fajar Farizqi Azmi_2311102192_modul 10>
```

Deskripsi program :

Program ini digunakan untuk mendistribusikan sejumlah ikan ke dalam wadah-wadah dengan kapasitas tertentu, lalu menghitung total berat dan rata-rata berat ikan di setiap wadah.

1. Mengelompokkan ikan secara merata ke dalam wadah sesuai dengan kapasitas maksimal per wadah.
2. Menghitung total berat ikan dalam setiap wadah.
3. Menghitung rata-rata berat ikan di tiap wadah untuk memberikan informasi distribusi yang lebih rinci.
4. Memastikan pembagian ikan optimal dengan pembulatan jumlah wadah ke atas, jika jumlah ikan tidak habis dibagi kapasitas wadah.

Program ini berguna dalam aplikasi pengelolaan logistik dan inventarisasi, terutama untuk memastikan distribusi beban yang merata.

III. Unguided

Source code :

```
// FAJAR FARIZQI AZMI
// 2311102192

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 rataRata(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 berat arrBalita
    var bMin, bMax float64

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

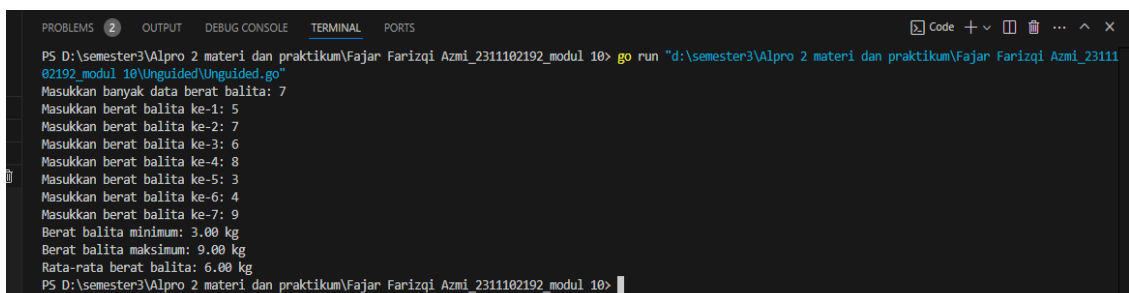
    for i := 0; i < n; i++ {
        fmt.Printf("Masukkan berat balita ke-%d: ", i+1)
        fmt.Scan(&berat[i])
    }

    hitungMinMax(berat, n, &bMin, &bMax)
    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)
}

```

Screenshot output :



```

PS D:\semester3\Alpro 2 materi dan praktikum\Fajar Farizqi Azmi_2311102192_modul 10> go run "d:\semester3\Alpro 2 materi dan praktikum\Fajar Farizqi Azmi_2311102192_modul 10\Unguided\Unguided.go"
Masukkan banyak data berat balita: 7
Masukkan berat balita ke-1: 5
Masukkan berat balita ke-2: 7
Masukkan berat balita ke-3: 6
Masukkan berat balita ke-4: 8
Masukkan berat balita ke-5: 3
Masukkan berat balita ke-6: 4
Masukkan berat balita ke-7: 9
Berat balita minimum: 3.00 kg
Berat balita maksimum: 9.00 kg
Rata-rata berat balita: 6.00 kg
PS D:\semester3\Alpro 2 materi dan praktikum\Fajar Farizqi Azmi_2311102192_modul 10>

```

Deskripsi program :

Program ini dirancang untuk menghitung statistik dasar dari kumpulan data berat badan balita, yaitu:

- Nilai minimum: Berat badan balita terendah.
- Nilai maksimum: Berat badan balita tertinggi.
- Rata-rata: Rata-rata berat badan seluruh balita.

Program ini merupakan program untuk menghitung statistik dasar dari suatu kumpulan data. Program ini dapat digunakan sebagai dasar untuk pengembangan program yang lebih kompleks dalam analisis data.