

**LAPORAN PRAKTIKUM  
ALGORITMA DAN PEMROGRAMAN 2**

**MODUL 3**

**PENCARIAN MODUL EKSTRIM PADA HIMPUNAN DATA**



Oleh:

**RYAN AKEYLA NOVIANTO WIDODO**

103112400081

12 IF 01

**S1 TEKNIK INFORMATIKA  
TELKOM UNIVERSITY PURWOKERTO**

**2025**

## **I. DASAR TEORI**

**Dasar teori penentuan nilai ekstrem (minimum dan maksimum) dalam array di Golang bergantung pada perbandingan berulang setiap elemen dalam array. Tidak ada fungsi bawaan Golang yang secara langsung memberikan nilai minimum dan maksimum sekaligus. Oleh karena itu, kita perlu melakukan iterasi melalui array dan melacak nilai minimum dan maksimum sejauh ini.**

**Algoritma:**

- 1. Inisialisasi: Nilai minimum ( min ) dan maksimum ( max ) diinisialisasi dengan elemen pertama array. Hal ini mengasumsikan array tidak kosong. Jika array kosong, penanganan khusus diperlukan (misalnya, mengembalikan error atau nilai default).**
- 2. Iterasi: Program melakukan iterasi (loop) melalui setiap elemen array mulai dari indeks ke-1 (elemen kedua).**
- 3. Perbandingan: Pada setiap iterasi, elemen saat ini dibandingkan dengan nilai min dan max yang sudah ada:**
  - Jika elemen saat ini lebih kecil dari min , maka min diperbarui dengan nilai elemen saat ini.**
  - Jika elemen saat ini lebih besar dari max , maka max diperbarui dengan nilai elemen saat ini.**
- 4. Hasil: Setelah iterasi selesai, min akan menyimpan nilai minimum dalam array, dan max akan menyimpan nilai maksimum dalam array.**

## II. GUIDED

Source Code + Screenshot hasil program beserta penjelasan

### Guided 1

Coding:

```
edit 7 77 Guided1.go 77 main
// RYAN AKEYLA NOVIANTO WIDODO
// 103112400081

package main

import "fmt"

func main() {
    var n int
    fmt.Print("Masukkan jumlah tanaman: ")
    fmt.Scan(&n)

    var tinggiTanaman [500]float64
    for i := 0; i < n; i++ {
        fmt.Printf("Masukkan tinggi tanaman ke-%d (cm): ", i+1)
        fmt.Scan(&tinggiTanaman[i])
    }

    min, max := tinggiTanaman[0], tinggiTanaman[0]
    for i := 1; i < n; i++ {
        if tinggiTanaman[i] < min {
            min = tinggiTanaman[i]
        }
        if tinggiTanaman[i] > max {
            max = tinggiTanaman[i]
        }
    }

    fmt.Printf("\nTinggi tanaman tertinggi: %.2f cm\n", max)
    fmt.Printf("Tinggi tanaman terpendek: %.2f cm\n", min)
}
```

Hasil Coding:

```
PS C:\Ryan\go run "c:\Ryan\Semester 2\Alpro\praktikum 6 pekan 09\103112400081_MODU
Masukkan jumlah tanaman: 4
Masukkan tinggi tanaman ke-1 (cm): 10
Masukkan tinggi tanaman ke-2 (cm): 23
Masukkan tinggi tanaman ke-3 (cm): 12
Masukkan tinggi tanaman ke-4 (cm): 11

Tinggi tanaman tertinggi: 23.00 cm
Tinggi tanaman terpendek: 10.00 cm
```

Tujuan program ini adalah membuat program yang mana kita mencari tinggi tanaman tertinggi dan terpendek dari sejumlah tanaman yang dimasukkan oleh pengguna. Pengguna diminta memasukkan jumlah tanaman, lalu memasukkan tinggi masing-masing tanaman dalam sentimeter (cm). Program kemudian memproses data tersebut untuk menentukan dan menampilkan tinggi tanaman tertinggi dan terpendek.

## Guided 2

Coding:

```
Guided2 > go Guided2.go > main
1 // RYAN AKEYLA NOVIANTO WIDODO
2 // 103112400081
3
4 package main
5
6 import (
7     "fmt"
8 )
9
10 func main() {
11     var x, y int
12     fmt.Print("Masukkan jumlah buku dan jumlah buku per rak: ")
13     fmt.Scan(&x, &y)
14
15     var hargaBuku [500]float64
16     fmt.Println("\nMasukkan Harga setiap buku (dalam ribuan Rp): ")
17     for i := 0; i < x; i++ {
18         fmt.Scan(&hargaBuku[i])
19     }
20
21     var hargaRataRata []float64
22     for i := 0; i < x; i += y {
23         total := 0.0
24         for j := i; j < i+y && j < x; j++ {
25             total += hargaBuku[j]
26         }
27         hargaRataRata = append(hargaRataRata, total/float64(y))
28     }
29     min, max := hargaBuku[0], hargaBuku[0]
30     for _, harga := range hargaBuku[:x] {
31         if harga < min {
32             min = harga
33         }
34         if harga > max {
35             max = harga
36         }
37     }
38     fmt.Printf("\nRata-rata harga per rak: ")
39     for _, avg := range hargaRataRata {
40         fmt.Printf("%2f", avg)
41     }
42     fmt.Printf("\nHarga termahal: %2.f Rp\n", max)
43     fmt.Printf("\nHarga termurah: %2.f Rp\n", min)
44 }
45
```

Hasil Coding:

```
PS C:\Ryan\Semester 2\Alpro\praktikum 6 pekan 09\103112400081_MODUL7> go run "c:\Ryan\Sem
Masukkan jumlah buku dan jumlah buku per rak: 6 2

Masukkan Harga setiap buku (dalam ribuan Rp):
20
30
70
80
56
34

Rata-rata harga per rak: 25.00000075.00000045.000000
Harga termahal: 20 Rp
Harga termurah: 20 Rp
PS C:\Ryan\Semester 2\Alpro\praktikum 6 pekan 09\103112400081_MODUL7>
```

Tujuan program ini adalah membuat program yang mana kita menghitung rata-rata harga buku per rak, harga buku termahal, dan harga buku termurah. Pengguna diminta memasukkan jumlah total buku ( x ) dan jumlah buku per rak ( y ). Kemudian, pengguna memasukkan harga setiap buku. Program selanjutnya menghitung rata-rata harga buku untuk setiap rak, dan juga mencari harga buku termahal dan termurah dari keseluruhan buku.

## Guided 3

Coding:

```
Guided3 > -o Guided3.go > main
1 // RYAN AKEYLA NOVIANTO WIDODO
2 // 103112400081
3
4 package main
5
6 import "fmt"
7
8 func main() {
9     var n int
10    fmt.Print("Masukkan jumlah siswa: ")
11    fmt.Scan(&n)
12
13    var nilaiSiswa [200]float64
14    var totalNilai float64 = 0
15
16    fmt.Println("\nMasukkan nilai ujian masing-masing siswa: ")
17    for i := 0; i < n; i++ {
18        fmt.Scan(&nilaiSiswa[i])
19        totalNilai += nilaiSiswa[i]
20    }
21
22    rataRata := totalNilai / float64(n)
23
24    min, max := nilaiSiswa[0], nilaiSiswa[0]
25    var diAtasRataRata int = 0
26    for _, nilai := range nilaiSiswa[:n] {
27        if nilai < min {
28            min = nilai
29        }
30        if nilai > max {
31            max = nilai
32        }
33        if nilai > rataRata {
34            diAtasRataRata++
35        }
36    }
37
38    fmt.Printf("\nNilai terendah: %.0f\n", min)
39    fmt.Printf("Nilai tertinggi: %.0f\n", max)
40    fmt.Printf("Rata-rata kelas: %.2f\n", rataRata)
41    fmt.Printf("Jumlah siswa di atas rata-rata: %d\n", diAtasRataRata)
42 }
```

Hasil Coding:

```
PS C:\Ryan\Semester 2\Alpro\praktikum 6 pekan 09\103112400081_MODUL7> go run "c:\Ryan\S
Masukkan jumlah siswa: 4

Masukkan nilai ujian masing-masing siswa:
80
90
79
77

Nilai terendah: 77
Nilai tertinggi: 90
Rata-rata kelas: 81.50
Jumlah siswa di atas rata-rata: 1
PS C:\Ryan\Semester 2\Alpro\praktikum 6 pekan 09\103112400081_MODUL7> 
```

Tujuan program ini adalah membuat program yang mana kita menghitung dan menampilkan nilai statistik dari nilai ujian sejumlah siswa. Program meminta pengguna memasukkan jumlah siswa, lalu nilai ujian masing-masing siswa. Setelah itu, program menghitung dan menampilkan:

- Nilai terendah: Nilai ujian terendah di antara semua siswa.
- Nilai tertinggi: Nilai ujian tertinggi di antara semua siswa.
- Rata-rata kelas: Rata-rata nilai ujian seluruh siswa.
- Jumlah siswa di atas rata-rata: Banyaknya siswa yang memiliki nilai ujian di atas rata-rata kelas.

### III. UNGUIDED

Source Code + Screenshot hasil program beserta penjelasan

#### Unguided 1

Coding:

```
Unguided1 > -o Unguided1.go > main
1 // RYAN AKEYLA NOVIANTO WIDODO
2 // 103112400081
3
4 package main
5
6 import (
7     "fmt"
8     "sort"
9 )
10
11 func main() {
12     var N int
13     fmt.Print("Masukkan jumlah anak kelinci: ")
14     fmt.Scan(&N)
15
16     weights := make([]float64, N)
17     fmt.Println("Masukkan berat anak kelinci:")
18     for i := 0; i < N; i++ {
19         fmt.Printf("Berat anak kelinci ke-%d: ", i+1)
20         fmt.Scan(&weights[i])
21     }
22     sort.Float64s(weights)
23
24     fmt.Printf("Berat terkecil: %.2f\n", weights[0])
25     fmt.Printf("Berat terbesar: %.2f\n", weights[N-1])
26 }
27
```

Hasil Coding:

```
PS C:\Ryan\Semester 2\Alpro\praktikum 6 pekan 09\103112400081_MODUL7> go run "c:\Ryan\Semester 2\Alpro\praktikum 6 pekan 09\103112400081_MODUL7\Unguided1.go"
Masukkan jumlah anak kelinci: 4
Masukkan berat anak kelinci:
Berat anak kelinci ke-1: 3
Berat anak kelinci ke-2: 2.5
Berat anak kelinci ke-3: 5
Berat anak kelinci ke-4: 3.3
Berat terkecil: 2.50
Berat terbesar: 5.00
PS C:\Ryan\Semester 2\Alpro\praktikum 6 pekan 09\103112400081_MODUL7>
```

Tujuan program ini adalah membuat program yang mana kita menghitung dan menampilkan nilai statistik dari nilai ujian sejumlah siswa. Program meminta pengguna memasukkan jumlah siswa, lalu nilai ujian masing-masing siswa

## Unguided 2

Coding:

```
Unguided2 > -o Unguided2.go > main
1 // RYAN AKEYLA NOVIANTO WIDODO
2 // 103112400081
3
4 package main
5
6 import (
7     "fmt"
8 )
9
10 func main() {
11     var x, y int
12     fmt.Println("Masukkan jumlah ikan (x) dan jumlah ikan per wadah (y): ")
13     fmt.Scan(&x, &y)
14
15     if x <= 0 || y <= 0 {
16         fmt.Println("Jumlah ikan dan kapasitas wadah harus lebih dari 0.")
17         return
18     }
19
20     beratIkan := make([]float64, x)
21     fmt.Println("\nMasukkan berat setiap ikan (dalam kg):")
22     for i := 0; i < x; i++ {
23         fmt.Print("Berat ikan ke-%d: ", i+1)
24         fmt.Scan(&beratIkan[i])
25     }
26
27     wadahCount := (x + y - 1) / y
28     totalBeratWadah := make([]float64, wadahCount)
29
30     for i := 0; i < x; i++ {
31         totalBeratWadah[i/y] += beratIkan[i]
32     }
33
34     fmt.Println("\nTotal berat ikan di setiap wadah:")
35     for i, total := range totalBeratWadah {
36         fmt.Printf("Wadah %d: %2.f kg\n", i+1, total)
37     }
38
39     totalBerat := 0.0
40     for _, berat := range totalBeratWadah {
41         totalBerat += berat
42     }
43     rataRata := totalBerat / float64(len(totalBeratWadah))
44
45     fmt.Printf("\nBerat rata-rata ikan di setiap wadah: %2.f kg\n", rataRata)
46 }
47
```

Hasil Coding:

```
PS C:\Ryan\Semester 2\Alpro\praktikum 6 pekan 09\103112400081_MODUL7> go run "c:\Ryan\Semester 2\Alpro\praktikum 6 pekan 09\103112400081_MODUL7\Unguided2.go"
Masukkan jumlah ikan (x) dan jumlah ikan per wadah (y): 5 3

Masukkan berat setiap ikan (dalam kg):
Berat ikan ke-%d: 1 3 2 5 4 4
Berat ikan ke-%d: 2Berat ikan ke-%d: 3Berat ikan ke-%d: 4Berat ikan ke-%d: 5
Total berat ikan di setiap wadah:
Wadah 1: 10 kg
Wadah 2: 8 kg

Berat rata-rata ikan di setiap wadah: 9 kg
PS C:\Ryan\Semester 2\Alpro\praktikum 6 pekan 09\103112400081_MODUL7>
```

Tujuan program ini adalah membuat program yang mana kita menghitung total berat dan rata-rata berat ikan di setiap wadah. Pengguna diminta memasukkan jumlah total ikan ( x ) dan jumlah ikan maksimum per wadah ( y ). Kemudian, pengguna memasukkan



berat setiap ikan. Program kemudian menghitung total berat ikan di setiap wadah dan rata-rata berat ikan per wadah.

## Unguided 3

Coding:

```
Unguided3 > -o Unguided3.go > main
1  // RYAN AKEYLA NOVIANTO WIDODO
2  // 103112400081
3
4  package main
5
6  import (
7      "fmt"
8      "math"
9  )
10
11 func main() {
12     var n int
13     fmt.Print("Masukkan banyak data berat balita: ")
14     fmt.Scanln(&n)
15
16     berat := make([]float64, n)
17     for i := 0; i < n; i++ {
18         fmt.Printf("Masukkan berat balita ke-%d: ", i+1)
19         fmt.Scanln(&berat[i])
20     }
21
22     minWeight := berat[0]
23     maxWeight := berat[0]
24     sum := 0.0
25
26     for _, w := range berat {
27         minWeight = math.Min(minWeight, w)
28         maxWeight = math.Max(maxWeight, w)
29         sum += w
30     }
31
32     avg := sum / float64(n)
33
34     fmt.Printf("Berat balita minimum: %.2f kg\n", minWeight)
35     fmt.Printf("Berat balita maksimum: %.2f kg\n", maxWeight)
36     fmt.Printf("Rerata Berat balita : %.2f kg\n", avg)
37 }
38
```

Hasil Coding:

```
Rerata Berat balita : 0.00 kg
PS C:\Ryan\Semester 2\Alpro\praktikum 6 pekan 09\103112400081_MODUL7> go run "c:\Ryan\Semester 2
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 C:\Ryan\Semester 2\Alpro\praktikum 6 pekan 09\103112400081_MODUL7> |
```

Tujuan program ini adalah membuat program yang mana kita menghitung dan menampilkan berat minimum, berat maksimum, dan rata-rata berat dari sejumlah data berat badan balita. Pengguna diminta memasukkan jumlah data berat balita, lalu memasukkan berat setiap balita. Program kemudian memproses data untuk mencari dan menampilkan nilai minimum, maksimum, dan rata-rata berat badan balita tersebut.

#### **IV. KESIMPULAN**

Kesimpulannya, penentuan nilai minimum dan maksimum dalam sebuah array di Golang dilakukan dengan iterasi linier melalui seluruh elemen array, membandingkan setiap elemen dengan nilai minimum dan maksimum yang telah ditemukan sejauh ini. Algoritma ini efisien dengan kompleksitas waktu  $O(n)$  dan ruang  $O(1)$ . Penggunaan fungsi bawaan seperti `math.Min` dan `math.Max` dapat meningkatkan efisiensi kode. Penting untuk menangani kasus array kosong untuk mencegah error.

## **V. REFERENSI**

Modul 7 Fungsi, Algoritma Pemrograman 2

<https://dasarpemrogramangolang.novalagung.com/>