

Nomor 1.

```
package main

import "fmt"

// Definisi tipe data array untuk menyimpan himpunan integer
type set_aldo_323 [2022]int

// Fungsi untuk mengecek keberadaan suatu nilai di dalam array
func exist_aldo_323(T_aldo_323 set_aldo_323, n_aldo_323 int, val_aldo_323 int) bool {
    for i_aldo_323 := 0; i_aldo_323 < n_aldo_323; i_aldo_323++ {
        if T_aldo_323[i_aldo_323] == val_aldo_323 {
            return true
        }
    }
    return false
}

// Fungsi untuk mengisi array dengan input bilangan unik dari pengguna
func inputSet_aldo_323(T_aldo_323 *set_aldo_323, n_aldo_323 *int) {
    fmt.Println("Masukkan bilangan unik (akhiri dengan bilangan yang sudah dimasukkan):")
    var x_aldo_323 int
    for {
        fmt.Print("Input: ")
        fmt.Scan(&x_aldo_323)
        if exist_aldo_323(*T_aldo_323, *n_aldo_323, x_aldo_323) { // Cek apakah
            // bilangan sudah ada di array
            fmt.Println("Bilangan sudah ada di himpunan, proses selesai.")
            break
        }
        T_aldo_323[*n_aldo_323] = x_aldo_323 // Tambahkan bilangan ke array
        *n_aldo_323 += 1 // Increment jumlah elemen
    }
}

// Fungsi untuk mencari irisan dari dua array
func findIntersection_aldo_323(T1_aldo_323, T2_aldo_323 set_aldo_323, n_aldo_323,
m_aldo_323 int, T3_aldo_323 *set_aldo_323, h_aldo_323 *int) {
    for i_aldo_323 := 0; i_aldo_323 < n_aldo_323; i_aldo_323++ {
        if exist_aldo_323(T2_aldo_323, m_aldo_323, T1_aldo_323[i_aldo_323]) { // Cek
            // apakah elemen di T1 juga ada di T2
            if !exist_aldo_323(*T3_aldo_323, *h_aldo_323,
                T1_aldo_323[i_aldo_323]) { // Hindari duplikasi di T3
                T3_aldo_323[*h_aldo_323] = T1_aldo_323[i_aldo_323] //
                // Tambahkan elemen ke array hasil
                *h_aldo_323 += 1 // Increment jumlah elemen
            }
        }
    }
}

}
```

```

// Fungsi untuk mencetak elemen-elemen array secara horizontal
func printSet_aldo_323(T_aldo_323 set_aldo_323, n_aldo_323 int) {
    fmt.Print("Hasil irisan: ")
    for i_aldo_323 := 0; i_aldo_323 < n_aldo_323; i_aldo_323++ {
        if i_aldo_323 > 0 {
            fmt.Print(" ")
        }
        fmt.Print(T_aldo_323[i_aldo_323])
    }
    fmt.Println()
}

// Fungsi utama program
func main() {
    var s1_aldo_323, s2_aldo_323, s3_aldo_323 set_aldo_323 // Array untuk menyimpan
    dua himpunan dan hasil irisan
    var n1_aldo_323, n2_aldo_323, n3_aldo_323 int // Variabel untuk menyimpan
    ukuran dari masing-masing array

    // Input himpunan pertama
    fmt.Println("Input himpunan pertama:")
    inputSet_aldo_323(&s1_aldo_323, &n1_aldo_323)

    // Input himpunan kedua
    fmt.Println("Input himpunan kedua:")
    inputSet_aldo_323(&s2_aldo_323, &n2_aldo_323)

    // Cari irisan antara s1 dan s2
    findIntersection_aldo_323(s1_aldo_323, s2_aldo_323, n1_aldo_323, n2_aldo_323,
    &s3_aldo_323, &n3_aldo_323)

    // Cetak hasil irisan
    if n3_aldo_323 > 0 {
        printSet_aldo_323(s3_aldo_323, n3_aldo_323)
    } else {
        fmt.Println("Hasil irisan kosong.")
    }
}

```

```

PS C:\Users\USER\OneDrive\Desktop\Alpro 2> go run "c:\Users\USER\OneDrive\Desktop\Alpro 2\Muhammad Ragiel Prastyo_2311102183_M
odu112\Unguided\Ujian\nomor1.go"
Input himpunan pertama:
Masukkan bilangan unik (akhiri dengan bilangan yang sudah dimasukkan):
Input: 12
Input: 12
Bilangan sudah ada di himpunan, proses selesai.
Input himpunan kedua:
Masukkan bilangan unik (akhiri dengan bilangan yang sudah dimasukkan):
Input: 12
Input: 14
Input: 15
Input: 16
Input: 17
Input: 19
Input: 12
Bilangan sudah ada di himpunan, proses selesai.
Hasil irisan: 12
PS C:\Users\USER\OneDrive\Desktop\Alpro 2>

```

Nomor 2.

```

package main

import (
    "bufio"
    "fmt"
    "os"
    "strconv"
    "strings"
)

// Struktur data mahasiswa
type Mahasiswa_aldo_323 struct {
    NIM_aldo_323 string
    Nama_aldo_323 string
    Nilai_aldo_323 int
}

// Array untuk menyimpan data mahasiswa
var arrayMahasiswa_aldo_323 []Mahasiswa_aldo_323

// Fungsi untuk memasukkan data mahasiswa
func inputMahasiswa_aldo_323(N_aldo_323 int) {
    reader_aldo_323 := bufio.NewReader(os.Stdin)
    for i_aldo_323 := 0; i_aldo_323 < N_aldo_323; i_aldo_323++ {
        fmt.Printf("Masukkan data mahasiswa ke-%d\n", i_aldo_323+1)

        fmt.Print("NIM: ")
        NIM_aldo_323, _ := reader_aldo_323.ReadString('\n')
        NIM_aldo_323 = strings.TrimSpace(NIM_aldo_323)

        fmt.Print("Nama: ")
        Nama_aldo_323, _ := reader_aldo_323.ReadString('\n')
        Nama_aldo_323 = strings.TrimSpace(Nama_aldo_323)

        fmt.Print("Nilai: ")
        NilaiStr_aldo_323, _ := reader_aldo_323.ReadString('\n')
        NilaiStr_aldo_323 = strings.TrimSpace(NilaiStr_aldo_323)
        Nilai_aldo_323, err_aldo_323 := strconv.Atoi(NilaiStr_aldo_323)
    }
}

```

```

        if err_aldo_323 != nil {
            fmt.Println("Nilai harus berupa angka!")
            i_aldo_323-- // Ulangi input untuk mahasiswa ini
            continue
        }

        arrayMahasiswa_aldo_323 = append(arrayMahasiswa_aldo_323,
        Mahasiswa_aldo_323{NIM_aldo_323, Nama_aldo_323, Nilai_aldo_323})
    }
}

// Fungsi untuk mencari nilai pertama berdasarkan NIM
func cariNilaiPertama_aldo_323(NIM_aldo_323 string) int {
    for _, mahasiswa_aldo_323 := range arrayMahasiswa_aldo_323 {
        if mahasiswa_aldo_323.NIM_aldo_323 == NIM_aldo_323 {
            return mahasiswa_aldo_323.Nilai_aldo_323
        }
    }
    return -1 // Jika NIM tidak ditemukan
}

// Fungsi untuk mencari nilai terbesar berdasarkan NIM
func cariNilaiTerbesar_aldo_323(NIM_aldo_323 string) int {
    nilaiTerbesar_aldo_323 := -1
    for _, mahasiswa_aldo_323 := range arrayMahasiswa_aldo_323 {
        if mahasiswa_aldo_323.NIM_aldo_323 == NIM_aldo_323 {
            if mahasiswa_aldo_323.Nilai_aldo_323 > nilaiTerbesar_aldo_323 {
                nilaiTerbesar_aldo_323 = mahasiswa_aldo_323.Nilai_aldo_323
            }
        }
    }
    return nilaiTerbesar_aldo_323
}

// Fungsi utama
func main() {
    reader_aldo_323 := bufio.NewReader(os.Stdin)

    // Input jumlah data mahasiswa
    fmt.Print("Masukkan jumlah data mahasiswa: ")
    NStr_aldo_323, _ := reader_aldo_323.ReadString('\n')
    NStr_aldo_323 = strings.TrimSpace(NStr_aldo_323)
    N_aldo_323, err_aldo_323 := strconv.Atoi(NStr_aldo_323)
    if err_aldo_323 != nil || N_aldo_323 < 1 {
        fmt.Println("Jumlah data mahasiswa tidak valid!")
        return
    }

    // Input data mahasiswa
    inputMahasiswa_aldo_323(N_aldo_323)
}

```

```

// Mencari nilai pertama
fmt.Print("\nMasukkan NIM untuk mencari nilai pertama: ")
NIM_aldo_323, _ := reader_aldo_323.ReadString('\n')
NIM_aldo_323 = strings.TrimSpace(NIM_aldo_323)

nilaiPertama_aldo_323 := cariNilaiPertama_aldo_323(NIM_aldo_323)
if nilaiPertama_aldo_323 != -1 {
    fmt.Printf("Nilai pertama mahasiswa dengan NIM %s adalah %d\n", NIM_aldo_323,
nilaiPertama_aldo_323)
} else {
    fmt.Printf("Mahasiswa dengan NIM %s tidak ditemukan.\n", NIM_aldo_323)
}

// Mencari nilai terbesar
fmt.Print("\nMasukkan NIM untuk mencari nilai terbesar: ")
NIM_aldo_323, _ = reader_aldo_323.ReadString('\n')
NIM_aldo_323 = strings.TrimSpace(NIM_aldo_323)

nilaiTerbesar_aldo_323 := cariNilaiTerbesar_aldo_323(NIM_aldo_323)
if nilaiTerbesar_aldo_323 != -1 {
    fmt.Printf("Nilai terbesar mahasiswa dengan NIM %s adalah %d\n", NIM_aldo_323,
nilaiTerbesar_aldo_323)
} else {
    fmt.Printf("Mahasiswa dengan NIM %s tidak ditemukan.\n", NIM_aldo_323)
}
}

```

```

PS C:\Users\USER\OneDrive\Desktop\Alpro 2> go run "c:\Users\USER\OneDrive\Desktop\Alpro 2\Muhammad Ragiel Prastyo_2311102183_M
odul12\Unguided\Ujian\nomor2.go"
Masukkan jumlah data mahasiswa: 3
Masukkan data mahasiswa ke-1
NIM: 12
Nama: aldo
Nilai: 45
Masukkan data mahasiswa ke-2
NIM: 12
Nama: aldo
Nilai: 44
Masukkan data mahasiswa ke-3
NIM: 12
Nama: aldo
Nilai: 67

Masukkan NIM untuk mencari nilai pertama: 12
Nilai pertama mahasiswa dengan NIM 12 adalah 45

Masukkan NIM untuk mencari nilai terbesar: 12
Nilai terbesar mahasiswa dengan NIM 12 adalah 67
PS C:\Users\USER\OneDrive\Desktop\Alpro 2>

```

Nomor 3.

```
package main
```

```
import (
    "fmt"

```

```

"strings"
)

const nProv_aldo_323 int = 34

type (
    NamaProv_aldo_323 [nProv_aldo_323]string
    PopProv_aldo_323  [nProv_aldo_323]int
    TumbuhProv_aldo_323 [nProv_aldo_323]float64
)

// InputData: Mengisi data nama provinsi, populasi, dan angka pertumbuhan penduduk.
func InputData(prov_aldo_323 *NamaProv_aldo_323, pop_aldo_323 *PopProv_aldo_323,
tumbuh_aldo_323 *TumbuhProv_aldo_323) {
    fmt.Println("Masukkan data untuk setiap provinsi:")
    for i := 0; i < nProv_aldo_323; i++ {
        fmt.Printf("Data Provinsi ke-%d:\n", i+1)
        fmt.Print("Nama Provinsi: ")
        fmt.Scanln(&prov_aldo_323[i])
        fmt.Print("Populasi (dalam angka bulat): ")
        fmt.Scanln(&pop_aldo_323[i])
        fmt.Print("Angka Pertumbuhan (desimal): ")
        fmt.Scanln(&tumbuh_aldo_323[i])
    }
}

// ProvinsiTercepat: Mencari indeks provinsi dengan pertumbuhan penduduk tertinggi.
func ProvinsiTercepat(tumbuh_aldo_323 TumbuhProv_aldo_323) int {
    maxIndex_aldo_323 := 0
    for i := 1; i < nProv_aldo_323; i++ {
        if tumbuh_aldo_323[i] > tumbuh_aldo_323[maxIndex_aldo_323] {
            maxIndex_aldo_323 = i
        }
    }
    return maxIndex_aldo_323
}

// Prediksi: Menampilkan prediksi jumlah penduduk tahun depan untuk provinsi dengan
pertumbuhan > 2%.
func Prediksi(prov_aldo_323 NamaProv_aldo_323, pop_aldo_323 PopProv_aldo_323,
tumbuh_aldo_323 TumbuhProv_aldo_323) {
    fmt.Println("\nPrediksi Populasi Tahun Depan untuk Provinsi dengan Pertumbuhan > 2%:")
    for i := 0; i < nProv_aldo_323; i++ {
        if tumbuh_aldo_323[i] > 0.02 {
            prediksi_aldo_323 := float64(pop_aldo_323[i]) * (1 + tumbuh_aldo_323[i])
            fmt.Printf("Provinsi: %s, Populasi Tahun Depan: %.2f\n", prov_aldo_323[i],
prediksi_aldo_323)
        }
    }
}

```

```

// IndeksProvinsi: Mencari indeks provinsi berdasarkan nama.
func IndeksProvinsi(prov_aldo_323 NamaProv_aldo_323, nama_aldo_323 string) int {
    for i := 0; i < nProv_aldo_323; i++ {
        if strings.EqualFold(prov_aldo_323[i], nama_aldo_323) {
            return i
        }
    }
    return -1
}

func main() {
    var prov_aldo_323 NamaProv_aldo_323
    var pop_aldo_323 PopProv_aldo_323
    var tumbuh_aldo_323 TumbuhProv_aldo_323

    // Input data
    InputData(&prov_aldo_323, &pop_aldo_323, &tumbuh_aldo_323)

    // Provinsi dengan angka pertumbuhan tercepat
    indexTercepat_aldo_323 := ProvinsiTercepat(tumbuh_aldo_323)
    fmt.Printf("\nProvinsi dengan angka pertumbuhan tercepat: %s\n",
prov_aldo_323[indexTercepat_aldo_323])

    // Mencari indeks provinsi berdasarkan input pengguna
    var namaCari_aldo_323 string
    fmt.Print("\nMasukkan nama provinsi yang ingin dicari: ")
    fmt.Scanln(&namaCari_aldo_323)
    index_aldo_323 := IndeksProvinsi(prov_aldo_323, namaCari_aldo_323)
    if index_aldo_323 != -1 {
        fmt.Printf("Indeks provinsi %s: %d\n", namaCari_aldo_323, index_aldo_323+1)
    } else {
        fmt.Println("Provinsi tidak ditemukan.")
    }

    // Prediksi populasi tahun depan
    Prediksi(prov_aldo_323, pop_aldo_323, tumbuh_aldo_323)
}

```

```

Populasi (dalam angka bulat): 932532
Angka Pertumbuhan (desimal): 12
Data Provinsi ke-32:
Nama Provinsi: sumatra
Populasi (dalam angka bulat): 10000
Angka Pertumbuhan (desimal): 89
Data Provinsi ke-33:
Nama Provinsi: sulawesi
Populasi (dalam angka bulat): 12
Angka Pertumbuhan (desimal): 90
Data Provinsi ke-34:
Nama Provinsi: papua
Populasi (dalam angka bulat): 7676
Angka Pertumbuhan (desimal): 120

Provinsi dengan angka pertumbuhan tercepat: papua

Masukkan nama provinsi yang ingin dicari: sumatra
Indeks provinsi sumatra: 32

Prediksi Populasi Tahun Depan untuk Provinsi dengan Pertumbuhan > 2%:
Provinsi: maluku, Populasi Tahun Depan: 12122916.00
Provinsi: sumatra, Populasi Tahun Depan: 900000.00
Provinsi: sulawesi, Populasi Tahun Depan: 1092.00
Provinsi: papua, Populasi Tahun Depan: 928796.00
PS C:\Users\USER\OneDrive\Desktop\Alpro 2> █

```

Nomor 4.

```

package main

import (
    "fmt"
    "sort"
)

const (
    MAX_MARKER_aldo_323 int = -5313541 // Penanda akhir data
)

func main() {
    var arrInt_aldo_323 []int // Array dinamis untuk menyimpan bilangan bulat
    var num_aldo_323 int      // Variabel untuk input bilangan

    fmt.Println("Masukkan bilangan bulat (akhiri dengan -5313541):")

    // Membaca input hingga mencapai nilai marker
    for {
        fmt.Scan(&num_aldo_323)
        if num_aldo_323 == MAX_MARKER_aldo_323 {
            break
        }
        if num_aldo_323 != 0 {
            arrInt_aldo_323 = append(arrInt_aldo_323, num_aldo_323) // Menyimpan bilangan bukan nol
        }
    }

    // Mengurutkan array dan menghitung median
    sort.Ints(arrInt_aldo_323) // Mengurutkan array
    median_aldo_323 := findMedian_aldo_323(arrInt_aldo_323)
}

```



```

// Output hasil
fmt.Println("Data yang tersimpan (terurut):", arrInt_aldo_323)
fmt.Printf("Median: %.0f\n", median_aldo_323)
}

// findMedian_aldo_323: Menghitung nilai median dari array terurut
func findMedian_aldo_323(arr_aldo_323 []int) float64 {
    n_aldo_323 := len(arr_aldo_323)
    if n_aldo_323 == 0 {
        return 0 // Jika array kosong, median adalah 0
    }
    if n_aldo_323%2 == 1 {
        // Jika jumlah elemen ganjil, median adalah elemen tengah
        return float64(arr_aldo_323[n_aldo_323/2])
    }
    // Jika jumlah elemen genap, median adalah rata-rata dua elemen tengah
    return float64(arr_aldo_323[n_aldo_323/2-1]+arr_aldo_323[n_aldo_323/2]) / 2
}

```

```

PS C:\Users\USER\OneDrive\Desktop\Alpro 2> go run "c:\Users\USER\OneDrive\Desktop\Alpro 2\Muhammad Ragiel Prastyo_2311102183_M
odul12\Unguided\Ujian\nomor4.go"
Masukkan bilangan bulat (akhiri dengan -5313541):
12
41
123
1231
112
2
1
1
12
12
3
1
2
3
12
3
131
31
1
-5313541
Data yang tersimpan (terurut): [1 1 1 1 2 2 3 3 3 12 12 12 12 31 41 112 123 131 1231]
Median: 12
PS C:\Users\USER\OneDrive\Desktop\Alpro 2>

```

Nomor 5.

```

package main

import (
    "fmt"
)

const NMAX_aldo_323 = 1000000 // Kapasitas maksimum array

// Struktur data untuk partai
type partai_aldo_323 struct {

```

```

    nama_aldo_323 int // Nama partai (diwakili angka)
    suara_aldo_323 int // Jumlah suara partai
}

// Fungsi utama
func main() {
    var p_aldo_323 []partai_aldo_323 // Array dinamis untuk menyimpan partai dan suara
    var namaPartai_aldo_323 int // Variabel untuk input nama partai

    fmt.Println("Masukkan nama partai (akhiri dengan -1):")

    // Input data hingga ditemukan tanda akhir (-1)
    for {
        fmt.Scan(&namaPartai_aldo_323)
        if namaPartai_aldo_323 == -1 {
            break // Keluar dari loop jika input adalah -1
        }

        // Cek apakah partai sudah ada dalam array
        index_aldo_323 := posisi_aldo_323(p_aldo_323, namaPartai_aldo_323)
        if index_aldo_323 == -1 {
            // Tambahkan partai baru jika belum ada
            p_aldo_323 = append(p_aldo_323, partai_aldo_323{nama_aldo_323:
namaPartai_aldo_323, suara_aldo_323: 1})
        } else {
            // Jika sudah ada, tambahkan jumlah suara
            p_aldo_323[index_aldo_323].suara_aldo_323++
        }
    }

    // Mengurutkan array berdasarkan jumlah suara (descending)
    insertionSort_aldo_323(p_aldo_323)

    // Tampilkan hasil
    fmt.Println("\nHasil penghitungan suara:")
    for _, partai_aldo_323 := range p_aldo_323 {
        fmt.Printf("Partai %d: %d suara\n", partai_aldo_323.nama_aldo_323,
partai_aldo_323.suara_aldo_323)
    }
}

// Fungsi posisi_aldo_323: Mencari indeks partai berdasarkan nama
func posisi_aldo_323(arr_aldo_323 []partai_aldo_323, nama_aldo_323 int) int {
    for i_aldo_323, partai_aldo_323 := range arr_aldo_323 {
        if partai_aldo_323.nama_aldo_323 == nama_aldo_323 {
            return i_aldo_323 // Kembalikan indeks jika ditemukan
        }
    }
    return -1 // Kembalikan -1 jika tidak ditemukan
}

```

```
// Fungsi insertionSort_aldo_323: Mengurutkan array partai berdasarkan jumlah suara
(descending)
func insertionSort_aldo_323(arr_aldo_323 []partai_aldo_323) {
    for i_aldo_323 := 1; i_aldo_323 < len(arr_aldo_323); i_aldo_323++ {
        temp_aldo_323 := arr_aldo_323[i_aldo_323]
        j_aldo_323 := i_aldo_323 - 1

        // Geser elemen ke kanan untuk menyisipkan elemen pada posisi yang tepat
        for j_aldo_323 >= 0 && arr_aldo_323[j_aldo_323].suara_aldo_323 <
temp_aldo_323.suara_aldo_323 {
            arr_aldo_323[j_aldo_323+1] = arr_aldo_323[j_aldo_323]
            j_aldo_323--
        }
        arr_aldo_323[j_aldo_323+1] = temp_aldo_323
    }
}
```

```
PS C:\Users\USER\OneDrive\Desktop\Alpro 2> go run "c:\Users\USER\OneDrive\Desktop\Alpro 2\Muhammad Ragiel Prastyo_2311102183_M
odul12\Unguided\Ujian\nomor5.go"
Masukkan nama partai (akhiri dengan -1):
perindo
12
12
313
1
231
31
-1

Hasil penghitungan suara:
Partai 0: 7 suara
Partai 12: 2 suara
Partai 313: 1 suara
Partai 1: 1 suara
Partai 231: 1 suara
Partai 31: 1 suara
PS C:\Users\USER\OneDrive\Desktop\Alpro 2> █
```