

NAMA : HAMID SABIRIN

NIM : 2311102129

KELAS : S1 IF-11-01

## 1. Soal 1

### Soal 1 (Voucher) TIPE I

[Nilai 40] Menjelang liburan lebaran tahun 2024 Universitas Telkom memberikan undian voucher Tel-U Gift kepada para mahasiswanya. Beragam hadiah yang bisa ditukarkan dari voucher tersebut, di antaranya adalah beasiswa BPP selama satu semester, laptop, dan peralatan belajar lengkap. Untuk itu diperlukan program untuk mengecek apakah voucher yang diberikan kepada mahasiswa valid atau tidak.

Voucher dikatakan valid, jika:

- Banyaknya digit dari nomor seri voucher adalah 5 atau 6.
- Perkalian digit pada dua digit pertama dan dua digit terakhir adalah sama.
- Digit yang berada di tengah (1 digit untuk panjang nomor seri 5, atau 3 digit untuk panjang nomor seri 6) adalah genap.

Masukan dan keluaran program adalah sebagai berikut:

**Masukan** terdiri dari beberapa baris. Baris pertama adalah bilangan bulat N yang menyatakan banyak mahasiswa. Selanjutnya diikuti sebanyak N baris yang menyatakan nomor seri dari masing-masing mahasiswa.

**Keluaran** berupa dua bilangan yang menyatakan banyaknya voucher valid dan tidak valid.

Kode

```
package main

import (
    "fmt"
)

func valid(voucher string) bool {
    panjang := len(voucher)
    if panjang != 5 && panjang != 6 {
        return false
    }

    duaawal := int(voucher[0]-'0') * int(voucher[1]-'0')
    duaakhir := int(voucher[panjang-2]-'0') * int(voucher[panjang-1]-'0')
    if duaawal != duaakhir {
        return false
    }

    if panjang == 5 {
```

```

        tengah := int(voucher[2] - '0')
        if tengah%2 != 0 {
            return false
        }
    } else if panjang == 6 {
        tengah1 := int(voucher[2] - '0')
        tengah2 := int(voucher[3] - '0')
        tengah3 := int(voucher[4] - '0')
        if tengah1%2 != 0 || tengah2%2 != 0 || tengah3%2 != 0 {
            return false
        }
    }

    return true
}

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

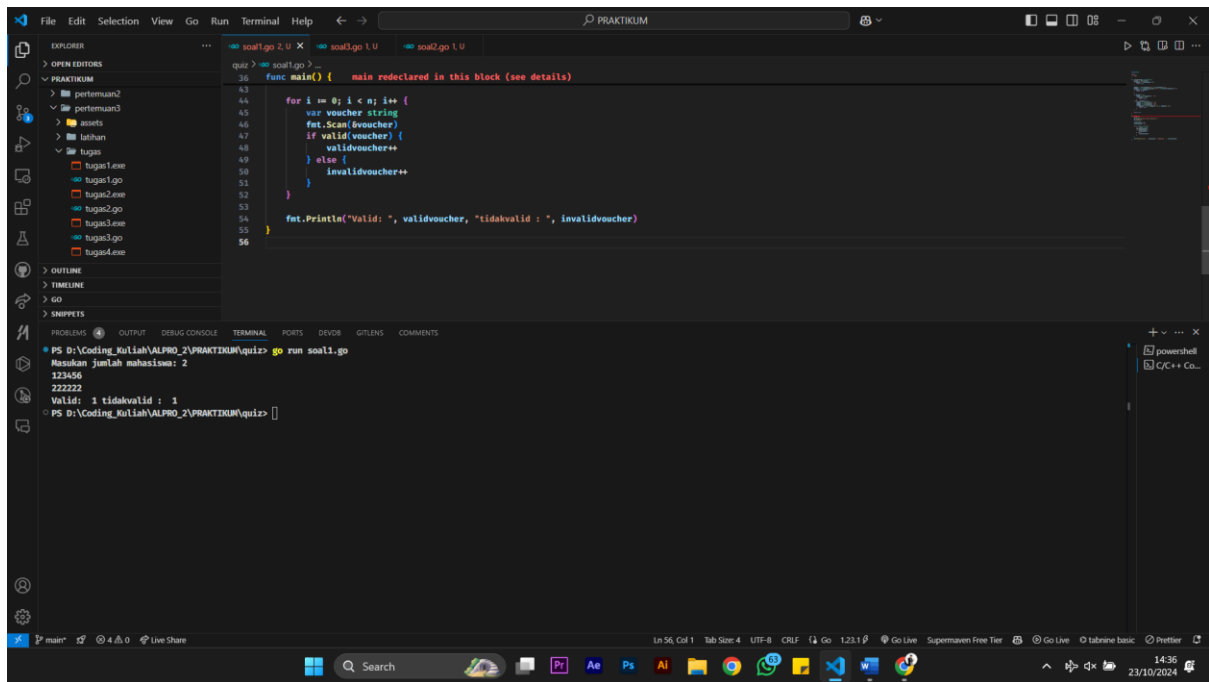
    validvoucher := 0
    invalidvoucher := 0

    for i := 0; i < n; i++ {
        var voucher string
        fmt.Scan(&voucher)
        if valid(voucher) {
            validvoucher++
        } else {
            invalidvoucher++
        }
    }

    fmt.Println(validvoucher, invalidvoucher)
}

```

Output



Kode

```
package main

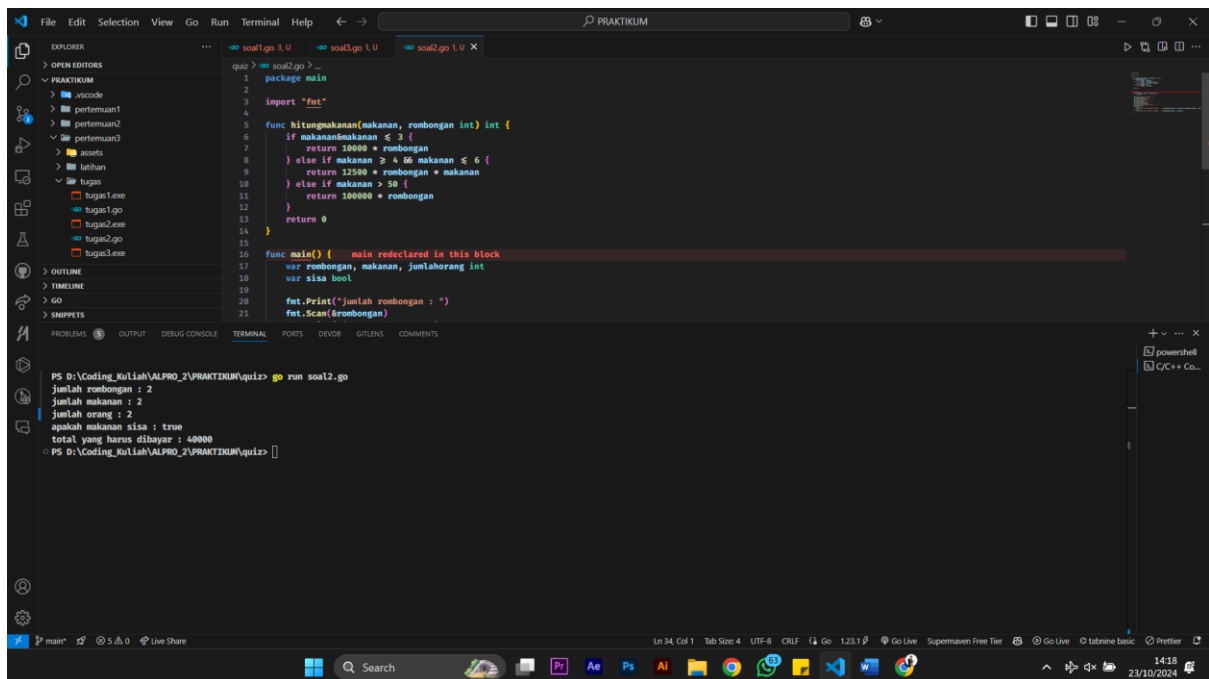
import "fmt"

func hitungmakanan(makanan, rombongan int) int {
    if makanan&makanan <= 3 {
        return 10000 * rombongan
    } else if makanan >= 4 && makanan <= 6 {
        return 12500 * rombongan * makanan
    } else if makanan > 50 {
        return 100000 * rombongan
    }
    return 0
}

func main() {
    var rombongan, makanan, jumlahorang int
    var sisa bool

    fmt.Print("jumlah rombongan : ")
    fmt.Scan(&rombongan)
    fmt.Print("jumlah makanan : ")
    fmt.Scan(&makanan)
    fmt.Print("jumlah orang : ")
    fmt.Scan(&jumlahorang)
    fmt.Print("apakah makanan sisa : ")
    fmt.Scan(&sisa)
    if sisa {
        fmt.Print("total yang harus dibayar : ", hitungmakanan(makanan,
rombongan)+hitungmakanan(makanan, rombongan))
    } else {
        fmt.Print("total yang harus dibayar : ", hitungmakanan(makanan,
rombongan))
    }
}
```

## Output



The screenshot shows a VS Code editor with a Go file named `soal2.go` and a terminal window displaying the program's output. The code defines a function `hitungMakanan` that calculates the total cost based on the number of people (`rombongan`) and the number of menu items (`makanan`). The terminal output shows the results of running the program with input values: 2 people, 2 menu items, and 2 people remaining.

```
1 package main
2
3 import "fmt"
4
5 func hitungMakanan(makanan, rombongan int) int {
6     if makanan <= 3 {
7         return 10000 * rombongan
8     } else if makanan >= 4 && makanan <= 6 {
9         return 2500 * rombongan * makanan
10    } else if makanan > 6 {
11        return 100000 * rombongan
12    }
13    return 0
14 }
15
16 func main() {
17     var rombongan, makanan, jumlahOrang int
18     var sisa bool
19     fmt.Println("jumlah rombongan : ")
20     fmt.Scan(&rombongan)
21 }
```

```
PS D:\Coding_Kuliah\ALPRO_2\PRAKTIKUM\quiz> go run soal2.go
jumlah rombongan : 2
jumlah makanan : 2
jumlah orang : 2
apakah makanan sisa : true
total yang harus dibayar : 40000
PS D:\Coding_Kuliah\ALPRO_2\PRAKTIKUM\quiz>
```

## Penjelasan

Pada kode tersebut merupakan kode untuk restoran yang dimana kita menginputkan jumlah rombongan, jumlah menu, jumlah orang dalam rombongan dan apakah makanan sisa atau tidak

Pada fungsi pertama yaitu kode hitung makanan ketika jumlah menu itu 1-3 maka harga nya 10000 \* rombongan, ketika 4-6 maka 2500 \* rombongan \* makanan dan ketika jumlah menu makanan itu lebih dari 50 maka dikenakan tarif 100000 \* rombongan dan ketika makanan sisa maka akan di kalikan jumlah rombongan dengan total tarifnya dan jika tidak sisa maka tidak kena denda.

### 3. Soal 3

[Nilai 20] Buatlah program bahasa Go untuk menjumlahkan bilangan-bilangan bulat positif kelipatan 4. Program yang dibuat harus menggunakan algoritma rekursif. Masukan dan keluaran program adalah sebagai berikut:

**Masukan** berupa serangkaian bilangan bulat dengan bilangan negatif sebagai sentinel.

**Keluaran** berupa jumlah bilangan bulat positif kelipatan 4.

Petunjuk: Buatlah solusi dalam bentuk prosedur iteratif, lalu gunakan konversi pola tail-end recursion untuk mendapatkan solusi bentuk rekursif.

Kode

```
package main

import (
    "fmt"
)

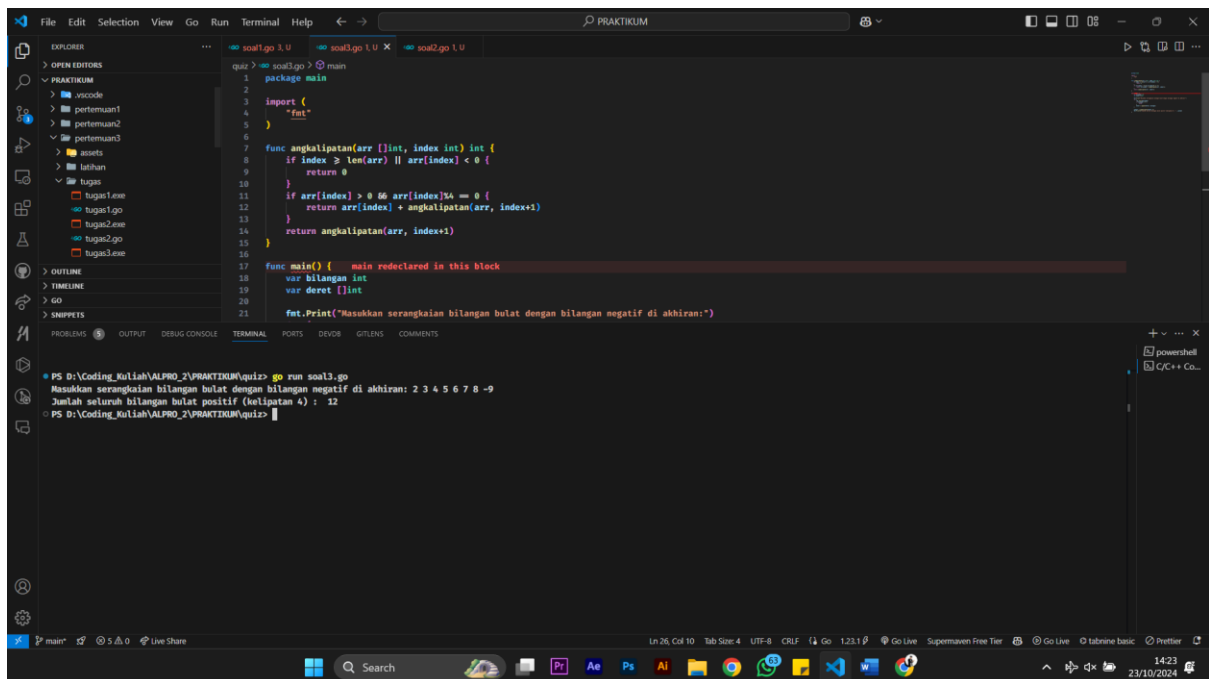
func angkalipatan(arr []int, index int) int {
    if index >= len(arr) || arr[index] < 0 {
        return 0
    }
    if arr[index] > 0 && arr[index]%4 == 0 {
        return arr[index] + angkalipatan(arr, index+1)
    }
    return angkalipatan(arr, index+1)
}

func main() {
    var bilangan int
    var deret []int

    fmt.Print("Masukkan serangkaian bilangan bulat dengan bilangan negatif di akhiran:")
    for {
        fmt.Scan(&bilangan)
        if bilangan < 0 {
            break
        }
        deret = append(deret, bilangan)
    }

    jumlah := angkalipatan(deret, 0)
    fmt.Println("Jumlah seluruh bilangan bulat positif (kelipatan 4) : ", jumlah)
}
```

## Output



```
1 package main
2
3 import (
4     "fmt"
5 )
6
7 func angkakelipatan(arr []int, index int) int {
8     if index >= len(arr) || arr[index] < 0 {
9         return 0
10    }
11    if arr[index] > 0 && arr[index]%4 == 0 {
12        return arr[index] + angkakelipatan(arr, index+1)
13    }
14    return angkakelipatan(arr, index+1)
15 }
16
17 func main() {
18     var bilangan int
19     var deret []int
20     fmt.Println("Masukkan serangkaian bilangan bulat dengan bilangan negatif di akhiran:")
21 }
```

PS D:\Coding\_Kuliah\ALPRO\_2\PRAKTIKUM\quiz> go run soal3.go  
Masukkan serangkaian bilangan bulat dengan bilangan negatif di akhiran: 2 3 4 5 6 7 8 -9  
Jumlah seluruh bilangan bulat positif (kelipatan 4) : 12  
PS D:\Coding\_Kuliah\ALPRO\_2\PRAKTIKUM\quiz>

## Penjelasan

Pada fungsi angkakelipatan kita melakukan sebuah percabangan atau pengecekan inut apakah input kurang dari 0 atau bukan dan apakah urutan inputan itu kurang dari 0 dan apakah habis dibagi 4 atau tidak jika iya maka akan di jumlahkan dengan angka kelipatan 4 selanjutnya dan pada fungsi utama kita menginputkan inputan integer sebanyak mungkin dan di akhiri dengan negatif untuk akhirnya dan jumlah kelipaan 4 itu akan di jumlahkan semua menggunakan fungsi angkakelipatan.