**LAPORAN PRAKTIKUM  
PEMROGRAMAN BERORIENTASI OBJEK**

**MODUL II**

**REVIEW STRUKTUR KONTROL**



Oleh:

AULIA RADIX PUTRA WINARKO

2311102056

S1IF-11-02

**S1 TEKNIK INRMATIKA**

**INSTITUT TEKNOLOGI TLKOM PURWOKERTO**

**2024**

1. **DASAR TEORI**
2. **Stuktur Program Go**

Dalam kerangka program yang ditulis dalam bahasa pemrograman Go, program utama selalu mempunyai dua komponen berikut:

* **package main** merupakan penanda bahwa file ini berisi program utama.
* **func main()** berisi kode utama dari sebuah program Go.

Komentar, bukan bagian dari kode program, dan dapat ditulis di mana saja di dalam program:

* Satu baris teks yang diawali dengan garis miring ganda ('//’') s.d. akhir baris, atau.
* Beberapa baris telts yang dimulai dengan pasangan karakter ‘/\*’ dan dialthin dengan ‘\*/’.

1. **Struktur Kontrol Dan Perulangan**

Go hanya mempunyai kata kunci for untuk semua jenis perulangan yang kita pelajari dalam notasi algoritma. Dua bentuk yang kita gunakan di sini adalah struktur while-loop dan repeat-until.

1. **Sruktur Kontrol Percabangan**

Untukanalisa kasus, bahasa Go mendukung dua bentuk percabangan, yaitu if-else dan switch-case.

* **Bentuk if-else**

Berikut ini bentuk-bentuk If-else yang mungkin dilakukan dalam bahasa Go. Semua bentuk di bawah merupakan satu instruksi if-else-endif saja (hanya satu endif). Bentuk If-else yang bersarang (dengan beberapa endif) dapat dibentuk dengan komposisi beberapa If-else-endif tersebut.

* **Bentuk switch-case**

Dalam bahasa Go ada dua variasi bentuk switch-case. Bentuk yang biasa digunakan adalah ekspresi ditulis pada perintah switch dan nilai ditulis dalam setiap label case-nya. Bentuk yang kedua mempunyal switch tanpa ekspresi, tetapi setiap case boleh berisi ekspresi boolean. Tentunya bentuk yang kedua lebih bersifat umum, dan merupakan penyederhanaan bentuk (atau alias dari) susunan suatu if-elself-..-else-endif.

1. **GUIDED**

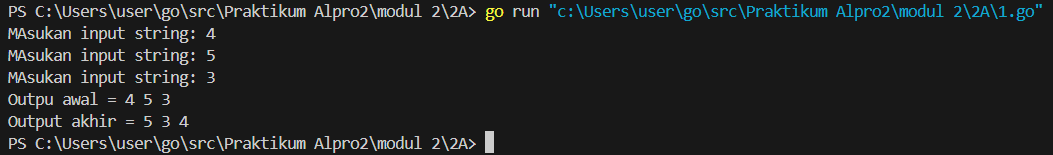
Source Code + Screenshot hasil program beserta penjelasan

**Latihan Modul 2A**

1. **Source Code**

|  |
| --- |
| package main  import "fmt"  func main() {      var (          satu, dua, tiga string          temp string      )      fmt.Print ("MAsukan input string: ")      fmt.Scanln (&satu)      fmt.Print ("MAsukan input string: ")      fmt.Scanln (&dua)      fmt.Print ("MAsukan input string: ")      fmt.Scanln (&tiga)      fmt.Println ("Outpu awal = " + satu + " " + dua + " " + tiga)      temp = satu      satu = dua      dua = tiga      tiga = temp      fmt.Println ("Output akhir = " + satu + " " + dua + " " + tiga)  } |

**Output**

****

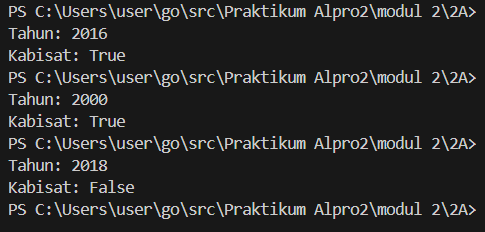
**Keterangan**

Program diatas berfungsi untuk menukar inputan yang dimana nilai satu disimpan pada temp ( sebagai penampung sementara ), nilai dua dipindah ke satu, nilai tiga dipindah ke dua, dan nilai temp dipindah ke tiga. Kemudaian dioutputkan nilai yang sudah ditukar .

1. **Source Code**

|  |
| --- |
| package main  import "fmt"  func main() {      var tahun int        fmt.Print ("Tahun: ")      fmt.Scanln(&tahun)      if tahun % 4 == 0 {          if tahun % 100 == 0 {              if tahun % 400 == 0{                  fmt.Print("Kabisat: True")              }else {                  fmt.Print("Kabisat: False")              }          } else {              fmt.Print("Kabisat: True")          }      } else {          fmt.Print("Kabisat: False")      }  } |

**Output**



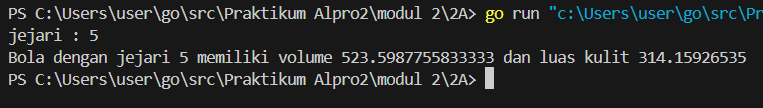
**Keterangan**

Program diata berfungsi untuk mengcek sebuah tahun meruoakan kabisat atau tidak, dengan cara membagi nilai tahun yang telah diinputkan dengan 4, jika hasil pembagian mempunyai sisa maka tahun yang diinputkan bukan tahun kabisat, jika tahun yang diinputkan habis dibagi 4 maka lanjut ketahap selanjutnya dengan dibagi 100 jika hasil pembagian mempunyai sisa maka tahun yang diinputkan berupa tahun kabisat, jika tahun yang diinputkan habis dibagi 100 maka lanjut ketahap selanjutnya dengan dibagi 400 jika hasil pembagian mempunyai sisa maka tahun yang diinputkan bukan tahun kabisat, jika tahun yang diinputkan habis dibagi 400 maka tahun kabisat.

1. **Source Code**

|  |
| --- |
| package main  import "fmt"  func main() {  var (jari, luas , volume float64)  fmt.Print ("jejari : ")  fmt.Scanln(&jari)  volume = (4.0/3.0) \* (3.1415926535) \* jari \* jari \* jari  luas = 4 \* (3.1415926535) \* jari \* jari  fmt.Print("Bola dengan jejari " ,jari, " memiliki volume " ,volume, " dan luas kulit ", luas)  } |

**Output**



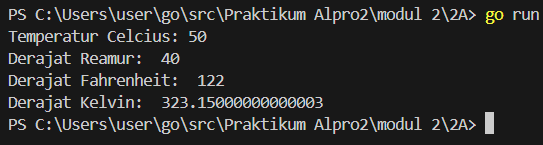
**Keterangan**

Program diatas digunakan untuk menghitung luas dan volume Bola,dengan cara mengimputkan jejari kemudian dioprasikan pada rumus volume dan luas yang telah dibuat dan di outputkan luas dan volume bola.

1. **Source Code**

|  |
| --- |
| package main  import "fmt"  func main(){      var celcius, reamur, kelvin, fahrenheit float64      fmt.Print("Temperatur Celcius: ")      fmt.Scanln(&celcius)      fahrenheit = celcius \* 9 / 5 + 32      reamur = celcius / 5 \* 4      kelvin = (fahrenheit + 459.67) / 9 \* 5      fmt.Println("Derajat Reamur: ", reamur)      fmt.Println("Derajat Fahrenheit: ", fahrenheit)      fmt.Println("Derajat Kelvin: ", kelvin)  } |

**Output**



**Keterangan**

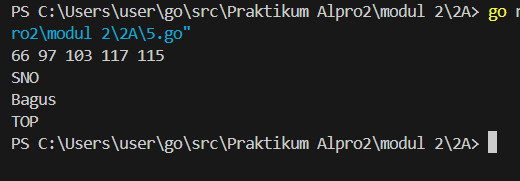
Program diatas dapat digunakan untuk mengkonversi suhu celcius

Ke reamur, kelvin, Fahrenheit. Dengan cara mengimputkan suhu celcius kemudian dikonversi menggunakan rumus yang telas dibuat dan di outputkan.

1. **Source Code**

|  |
| --- |
| package main  import "fmt"  func main() {      var nums [5]int      var chars [3]rune      for i := 0; i < 5; i++ {          fmt.Scan(&nums[i])      }      fmt.Scanln()      for i := 0; i < 3; i++ {          fmt.Scanf("%c", &chars[i])      }      for i := 0; i < 5; i++ {          fmt.Printf("%c", nums[i])      }      fmt.Println()      for i := 0; i < 3; i++ {          fmt.Printf("%c", chars[i]+1)      }  } |

**Output**



**Keterangan**

Program di atas menggabungkan pengolahan input bilangan bulat dan karakter dari pengguna, lalu menampilkan representasi karakter dari bilangan bulat (berdasarkan kode ASCII) dan menampilkan karakter yang dinaikkan 1 berdasarkan nilai ASCII-nya.

1. **UNGUIDED**

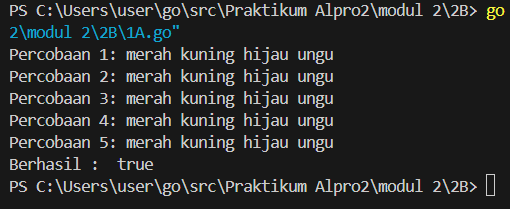
Source Code + Screenshot hasil program beserta penjelasan

**Latihan Modul 2B**

1. **Source Code**

|  |
| --- |
| package main  import "fmt"  func main() {      var warna [4]string      var urutan\_warna = [4]string{"merah", "kuning", "hijau", "ungu"}      var berhasil bool      berhasil = true      for i := 0; i < 5; i++ {          fmt.Print("Percobaan ", i+1, ": ")          for i := 0; i < 4; i++ {              fmt.Scan(&warna [i])          }          for i := 0; i < 4 ; i++ {              if warna[i] != urutan\_warna [i] {                  berhasil = false              }          }      }      fmt.Println("Berhasil : ", berhasil)  } |

**Output**



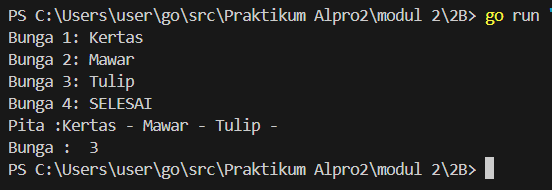
**Keterangan**

Program diatas meminta user untuk memasukkan urutan warna dalam beberapa percobaan, kemudian mengecek apakah urutan yang dimasukkan benar atau tidak. Jika user memasukkan urutan yang salah dalam satu percobaan, variabel berhasil diubah menjadi false, dan program akan mencetak hasil akhir setelah 5 percobaan.

1. **Source Code**

|  |
| --- |
| package main  import ("fmt")  func main() {      var n int      var bunga [20] string      n = 0      for {          fmt.Print ("Bunga ", n + 1, ": ")          fmt.Scan (&bunga[n])          if bunga [n] == "SELESAI"{              break          }          n++      }      fmt.Print ("Pita :")      for i := 0; i < n; i++ {          fmt.Print (bunga [i], " - ")      }      fmt.Println ("\nBunga : ", n)  } |

**Output**



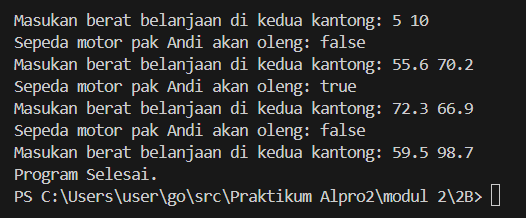
**Keterangan**

Program ini meminta user untuk memasukkan nama bunga, menyimpannya dalam array, dan berhenti jika user memasukkan kata "SELESAI". Setelah itu, program mencetak semua bunga yang dimasukkan dengan format yang ditentukan (memisahkan bunga dengan " - ") dan menghitung jumlah bunga yang telah diinput.

1. **Source Code**

|  |
| --- |
| package main  import "fmt"  func main(){      var oleng bool      var kanan, kiri float64      for {          oleng = false          fmt.Print ("\nMasukan berat belanjaan di kedua kantong: ")          fmt.Scanln (&kanan, &kiri)          if kiri > 0 && kanan >0 {              if kiri > kanan && kiri - kanan >= 9 {                  oleng = true              }              if kanan > kiri && kanan - kiri >= 9 {                  oleng = true              }          }          if kiri + kanan > 150 {              fmt.Print ("Program Selesai.")              break          }else {              fmt.Print ("Sepeda motor pak Andi akan oleng: ", oleng)          }      }  } |

**Output**

****

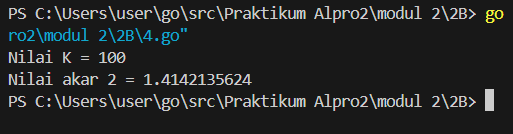
**Keterangan**

Program di atas memeriksa apakah sepeda motor Pak Andi akan oleng berdasarkan selisih berat belanjaan di kedua kantong. Jika perbedaan berat antara kantong kiri dan kanan mencapai 9 kg atau lebih, motor dianggap oleng. Program berhenti ketika total berat belanjaan di kedua kantong melebihi 150 kg.

1. **Source Code**

|  |
| --- |
| package main  import "fmt"  func main() {      var k int      var result float64 = 1.0      fmt.Print("Nilai K = ")      fmt.Scanln(&k)      for i := 0; i < k; i++ {          result = 0.5 \* (result + 2/result)      }      fmt.Printf("Nilai akar 2 = %.10f\n", result)  } |

**Output**



**Keterangan**

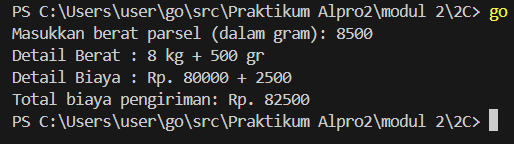
Program ini menggunakan metode Newton-Raphson untuk mendekati nilai √2 dengan iterasi. Nilai result dihitung secara bertahap melalui loop, dan semakin besar nilai k, semakin akurat hasil perhitungan.

**Latihan Modul 2B**

1. **Source Code**

|  |
| --- |
| package main  import "fmt"  func main() {      var berat int      var biayagr int      var biayakg int      var biayatotal int      const biayaPerKg = 10000      fmt.Print("Masukkan berat parsel (dalam gram): ")      fmt.Scanln(&berat)      kg := berat / 1000      Gram := berat % 1000      biayakg = kg \* biayaPerKg      if kg < 10 {          if Gram > 500 {          biayagr += Gram \* 15          } else {          biayagr += Gram \* 5          }      }      biayatotal = biayakg + biayagr      fmt.Println("Detail Berat :", kg,"kg +", Gram,"gr" )      fmt.Println("Detail Biaya : Rp.",biayakg, "+", biayagr )      fmt.Println("Total biaya pengiriman: Rp.",biayatotal)  } |

**Output**

****

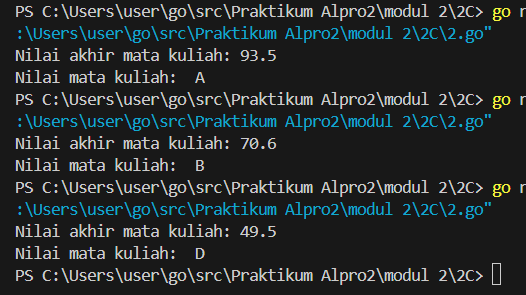
**Keterangan**

Program di atsa menghitung biaya pengiriman parsel berdasarkan berat dalam kilogram dan gram dengan tarif per kilogram yang tetap dan tarif tambahan untuk berat gram di bawah 1 kilogram.

1. **Source Code**

|  |
| --- |
| package main  import "fmt"  func main() {      var nam float64      var nmk string      fmt.Print("Nilai akhir mata kuliah: ")      fmt.Scanln(&nam)      if nam > 80 {          nmk = ("A")      } else if nam > 72.5 {          nmk = "AB"      } else if nam > 65 {          nmk = "B"      } else if nam > 57.5 {          nmk = "BC"      } else if nam > 50 {          nmk = "C"      } else if nam > 40 {          nmk = "D"      } else if nam <= 40 {          nmk = "E"      }      fmt.Println("Nilai mata kuliah: ", nmk)  } |

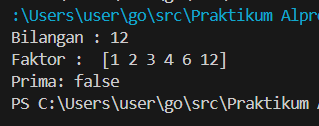
1. Jika nam diberikan adalah 80.1 dan user mengginputkan nilai 80 maka akan muncul output AB. Dan tidak sesuai spesifikasi soal.
2. Kesalahan pada program contah adalah tidak adanya else if setelah if. Jika kondisi dalam if pertama tidak terpenuhi, program akan memeriksa kondisi dalam else if berikutnya, dan begitu seterusnya. Program ini bekerja dengan cara meminta input nilai akhir dari pengguna, lalu menentukan grade berdasarkan rentang nilai tertentu.



1. **Source Code**

|  |
| --- |
| package main  import "fmt"  func main(){      var b int      var f [] int      fmt.Print ("Bilangan : ")      fmt.Scan(&b)      if b > 1 {          for i := 1; i <= b; i++ {              if b % i == 0{                  f = append(f, i)              }          }      }      fmt.Println("Faktor : ", f)      if len(f) == 2 {          fmt.Println("Prima: true")      } else {          fmt.Println ("Prima: false")      }  } |

**Output**



**Keterangan**

Program di atas menghitung faktor-faktor dari sebuah bilangan dan menentukan apakah bilangan tersebut adalah bilangan prima atau bukan. Jika jumlah faktor hanya dua, maka bilangan tersebut adalah prima.