

LAPORAN PRAKTIKUM
Algoritma Pemrograman

MODUL 2
I/O, TIPE DATA & VARIABEL



Disusun oleh:

ERIC SETIAWAN

109082500197

S1IF-13-04

PROGRAM STUDI S1 INFORMATIKA
FAKULTAS INFORMATIKA
TELKOM UNIVERSITY PURWOKERTO

2025

LATIHAN KELAS – GUIDED

1. Guided 1 Source Code

```
package main

import "fmt"

func main() {

    var a, b, c, d, e int

    var hasil int

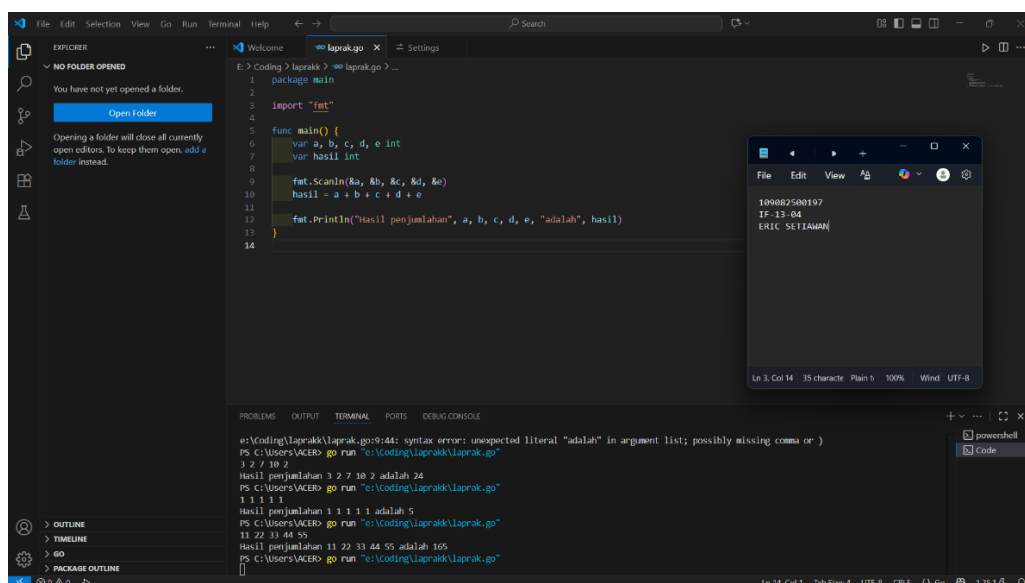
    fmt.Scanln(&a, &b, &c, &d, &e)

    hasil = a + b + c + d + e

    fmt.Println("Hasil penjumlahan", a, b, c, d, e,
        "adalah", hasil)

}
```

Screenshoot program



Deskripsi program

- 1) package main
 - Jadi setiap file Go harus dimulai dengan package.
 - "main" adalah package khusus di Go yang menandakan program utama.
 - Saat menjalankan go run atau go build, Go akan mencari fungsi main() dalam package main sebagai titik masuk eksekusi.
- 2) import "fmt"
 - fmt adalah paket standar Go untuk (input/output).
 - Fungsi-fungsi penting yang dipakai di sini:
 - fmt.Scanln() untuk membaca input dari user melalui keyboard.
 - fmt.Println() untuk mencetak output ke layar dengan newline otomatis.
- 3) func main() { ... }
 - main() adalah fungsi utama.
 - Program Go selalu mulai eksekusi dari fungsi ini.
 - Tidak boleh ada parameter atau return di main().
- 4) var a, b, c, d, e int
 - Mendeklarasikan 5 buah variabel integer (int).
 - Default nilai awal (zero value) dari int di Go adalah 0.
 - Jadi sebelum ada input, semua a, b, c, d, e = 0.
- 5) var hasil int
 - Variabel hasil juga bertipe integer.
 - Akan dipakai untuk menyimpan jumlah dari a+b+c+d+e.
- 6) fmt.Scanln(&a, &b, &c, &d, &e)
 - Scanln membaca input dari keyboard (stdin).
 - Tanda & berarti kita mengirimkan alamat variabel (pointer), bukan nilainya.
 - Jadi nilai yang dimasukkan kita akan disimpan langsung ke variabel a, b, c, d, e.
 - Contoh misal kita ngetik 1 2 3 4 5 maka a=1 b=2 c=3 d=4 e=5
- 7) hasil = a + b + c + d + e
 - Melakukan operasi aritmetika penjumlahan kelima angka yang sudah diinput.
 - Nilai totalnya disimpan ke variabel hasil.
 - Contohnya jika input 1 2 3 4 5 maka nanti muncul hasil = 15.
- 8) fmt.Println("Hasil penjumlahan", a, b, c, d, e, "adalah", hasil)
 - Mencetak output ke layar.
 - Println otomatis menambahkan spasi antar argumen, jadi tidak perlu menambahkan +.
 - Jika input 1 2 3 4 5, maka outputnya hasil penjumlahan dari 1 2 3 4 5 adalah 15.

2. Guided 2

Source Code

```
package main

import "fmt"

func main() {

    var x, fx float64

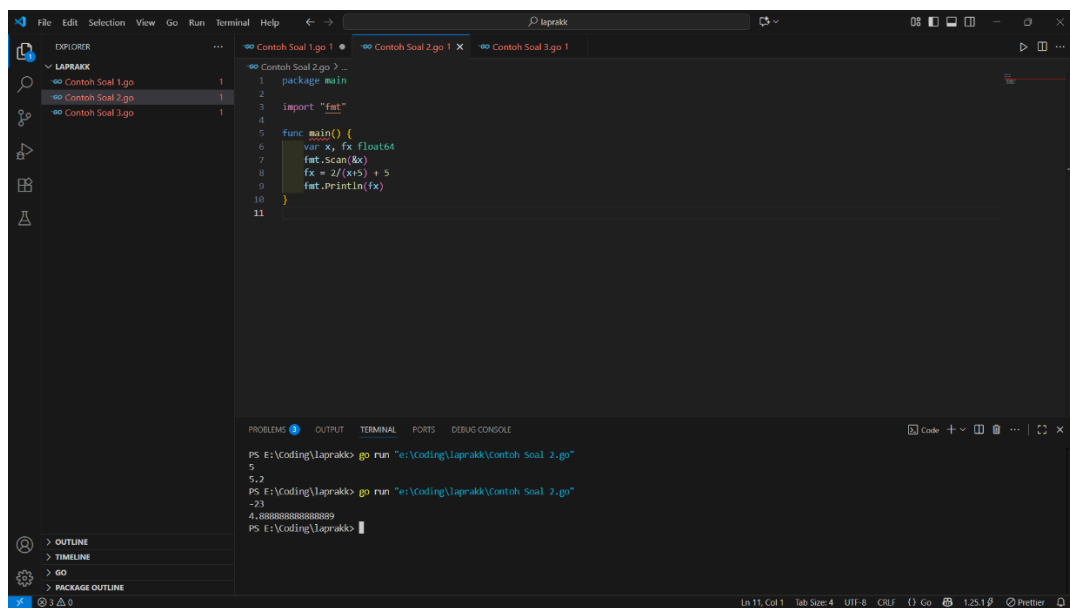
    fmt.Scan(&x)

    fx = 2/(x+5) + 5

    fmt.Println(fx)

}
```

Screenshoot program



Deskripsi program

1) package main

- Jadi setiap file Go harus dimulai dengan package.
- “main” adalah package khusus di Go yang menandakan program utama.
- Saat menjalankan go run atau go build, Go akan mencari fungsi main() dalam package main sebagai titik masuk eksekusi.

- 2) `import "fmt"`
 - Mengimpor paket standar `fmt` untuk input/output.
 - Dipakai di sini untuk:
 - `fmt.Scan(&x)` untuk membaca input dari user.
 - `fmt.Println(fx)` untuk mencetak hasil perhitungan ke layar.
- 3) `func main()`
 - Fungsi `main()` adalah titik masuk program.
 - Semua kode yang ada di dalamnya akan dijalankan saat program dipanggil.
- 4) `var x, fx float64`
 - Mendeklarasikan dua variabel:
 - `x` untuk menampung input user (bilangan real).
 - `fx` untuk menyimpan hasil perhitungan.
 - `float64` adalah tipe bilangan pecahan dengan presisi ganda (64-bit).
 - Menggunakan `float64` penting karena operasi $2/(x+5)$ bisa menghasilkan bilangan desimal (bukan integer bulat).
- 5) `fmt.Scan(&x)`
 - `Scan` membaca input dari user.
 - `&x` berarti alamat memori variabel `x`, agar nilai yang diinput user disimpan langsung ke `x`.
 - Contoh: jika user mengetik 3, maka `x = 3`.
- 6) `fx = 2/(x+5) + 5`
 - $$f(x) = \frac{2}{x+5} + 5$$
 - `x + 5` untuk menambahkan 5 ke nilai input.
 - $2/(x+5)$ untuk menghitung pecahan 2 dibagi hasil tadi.
 - `... + 5` untuk hasil pembagian ditambah 5.
 - Hasilnya disimpan ke variabel `fx`.
- 7) `fmt.Println(fx)`
 - Menampilkan hasil perhitungan ke layar.
 - Karena `fx` bertipe `float64`, hasil yang dicetak bisa berupa bilangan pecahan.

3. Guided 3

Source Code

```
package main

import "fmt"

func main() {

    var c1, c2, c3, c4, c5 byte

    var b1, b2, b3 int


    fmt.Scan(&c1, &c2, &c3, &c4, &c5)

    fmt.Scanf("%c", &b1)

    fmt.Scanf("%c", &b2)

    fmt.Scanf("%c", &b3)


    fmt.Printf("%c%c%c%c%c", c1, c2, c3, c4, c5)

    fmt.Printf("%c%c%c", b1+1, b2+1, b3+1)


}
```

Screenshoot program

```
1 package main
2
3 import "fmt"
4
5 func main() {
6     var c1, c2, c3, c4, c5 byte
7     var b1, b2, b3 int
8
9     fmt.Scan(&c1, &c2, &c3, &c4, &c5)
10    fmt.Scanf("%c", &b1)
11    fmt.Scanf("%c", &b2)
12    fmt.Scanf("%c", &b3)
13
14    fmt.Printf("%c%c%c%c%c", c1, c2, c3, c4, c5)
15    fmt.Printf("%c%c%c", b1+1, b2+1, b3+1)
16
17 }
18
```

PROBLEMS OUTPUT TERMINAL PORTS DEBUG CONSOLE

```
PS E:\Coding\LapraKK> go run "e:\Coding\LapraKK\Contoh Soal 3.go"
Contoh Soal 3.go:11: expected 'package', found 'EOF'
PS E:\Coding\LapraKK> go run "e:\Coding\LapraKK\Contoh Soal 3.go"
Contoh Soal 3.go:11: expected 'package', found 'EOF'
PS E:\Coding\LapraKK> go run "e:\Coding\LapraKK\Contoh Soal 3.go"
66 97 103 117 115 SHD
BagasTOR
PS E:\Coding\LapraKK>
```

Deskripsi program

- 1) func main()
 - Fungsi main() adalah titik awal eksekusi program.
 - Semua kode di dalam blok ini akan dijalankan saat program dipanggil.
- 2) var c1, c2, c3, c4, c5 byte
 - Mendeklarasikan 5 variabel (c1 s/d c5) bertipe byte.
 - byte di Go adalah alias dari uint8, yaitu bilangan 8-bit (0–255).
 - Biasanya dipakai untuk menyimpan karakter ASCII.
- 3) var b1, b2, b3 int
 - Mendeklarasikan 3 variabel bertipe int.
 - Nantinya dipakai untuk menyimpan input karakter lain, tetapi dengan tipe integer.
- 4) fmt.Scan(&c1, &c2, &c3, &c4, &c5)
 - Membaca 5 buah input sekaligus dari user.
 - Karena tipenya byte, biasanya input berupa karakter.
 - Contoh: jika user mengetik "Hello" c1 = 'H' c2 = 'e' c3 = 'l' c4 = 'l' c5 = 'o'
- 5) fmt.Scanf("%c", &b1)
 - Membaca 1 karakter dari input, sesuai format %c.
 - Input disimpan ke variabel integer (b1, b2, b3).
 - %c artinya input dianggap sebagai karakter ASCII, tapi karena disimpan di int, maka nilainya adalah kode ASCII dari karakter tersebut.
 - Misalnya input 'A', maka b1 = 65 (karena ASCII A = 65).
- 6) fmt.Printf("%c%c%c%c%c", c1, c2, c3, c4, c5)
 - Menampilkan kembali 5 karakter yang dimasukkan di awal.
 - %c artinya cetak dalam bentuk karakter ASCII.
 - Jadi jika input H e l l o, output pertama Adalah Hello
- 7) fmt.Printf("%c%c%c", b1+1, b2+1, b3+1)
 - Mencetak 3 karakter berikutnya dari input b1, b2, b3.
 - Karena %c dipakai, angka yang dicetak akan diubah ke karakter ASCII.

TUGAS

1. Tugas 1

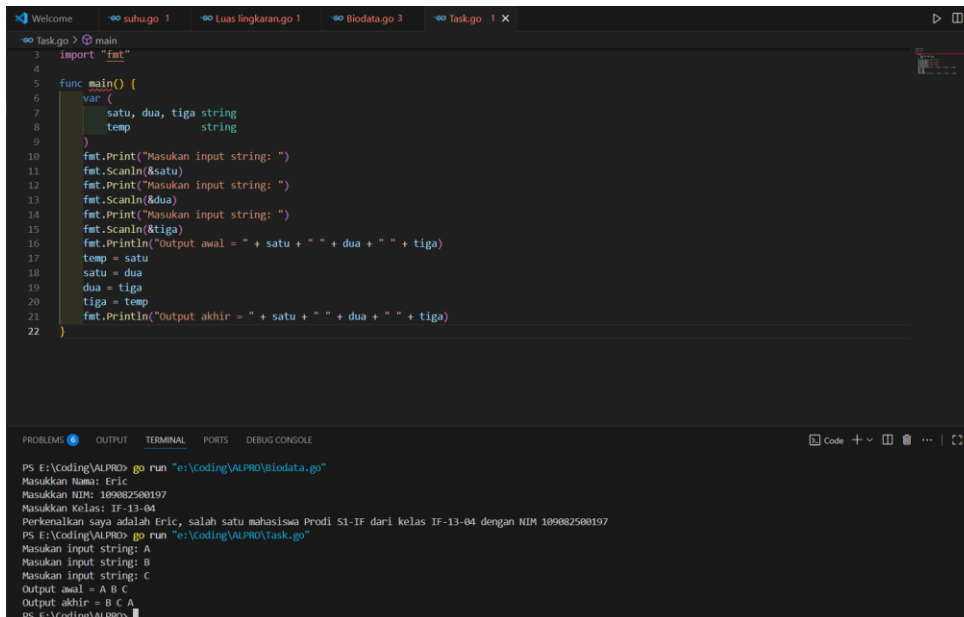
Source code

```
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("Output awal = " + satu + " " + dua + " " + tiga)
    temp = satu
    satu = dua
    dua = tiga
    tiga = temp
    fmt.Println("Output akhir = " + satu + " " + dua + " " + tiga)
}
```


Screenshoot program



```
Task.go > main
3 import "fmt"
4
5 func main() {
6     var (
7         satu, dua, tiga string
8         temp      string
9     )
10    fmt.Print("Masukan input string: ")
11    fmt.Scanln(&satu)
12    fmt.Print("Masukan input string: ")
13    fmt.Scanln(&dua)
14    fmt.Print("Masukan input string: ")
15    fmt.Scanln(&tiga)
16    fmt.Println("Output awal = " + satu + " " + dua + " " + tiga)
17    temp = satu
18    satu = dua
19    dua = tiga
20    tiga = temp
21    fmt.Println("Output akhir = " + satu + " " + dua + " " + tiga)
22 }
```

PROBLEMS OUTPUT TERMINAL PORTS DEBUG CONSOLE

```
PS E:\Coding\VALPRO> go run "e:\Coding\VALPRO\biodata.go"
Masukkan Nama: Eric
Masukkan NIM: 109082500197
Masukkan Kelas: IF-13-04
Perkenalkan saya adalah Eric, salah satu mahasiswa Prodi SI-IF dari kelas IF-13-04 dengan NIM 109082500197
PS E:\Coding\VALPRO> go run "e:\Coding\VALPRO\Task.go"
Masukan input string: A
Masukan input string: B
Masukan input string: C
Output awal = A B C
Output akhir = B C A
PS E:\Coding\VALPRO>
```

Deskripsi program

1) import "fmt"

- Mengimpor paket fmt (formatting) dari Go standard library.
- Fungsi yang digunakan dari fmt:
- fmt.Print untuk menampilkan teks ke layar tanpa newline.
- fmt.Scanln untuk membaca input dari user.
- fmt.Println untuk mencetak teks ke layar dengan newline otomatis.

2) func main()

- Fungsi main() adalah entry point program.
- Semua kode yang ada di dalam kurung kurawal {} inilah yang akan dijalankan ketika program dieksekusi.

3) var (satu, dua, tiga string temp string)

- mendeklarasikan variabel:
- satu, dua, tiga bertipe string, digunakan untuk menyimpan input dari user.
- temp variabel sementara bertipe string, digunakan untuk membantu proses pertukaran nilai (swapping).

2. Tugas 2

Source code

```
package main

import "fmt"

func main() {
```

```

var nama, nim, kelas string

fmt.Print("Masukkan Nama: ")

fmt.Scanln(&nama)

fmt.Print("Masukkan NIM: ")

fmt.Scanln(&nim)

fmt.Print("Masukkan Kelas: ")

fmt.Scanln(&kelas)

fmt.Println("Perkenalkan saya adalah", nama+",
salah satu mahasiswa Prodi S1-IF dari kelas
"+kelas+" dengan NIM "+nim)
}

```

Screenshoot program

The screenshot shows a Go program in VS Code. The code defines a `main` function that prompts the user for their name, NIM, and class, and then prints a formatted introduction. The terminal output shows the program being run, with the user inputting 'Eric', '109082500197', and 'IF-13-04', resulting in a formatted output line.

```

1 package main
2
3 import "fmt"
4
5 func main() {
6     var nama, nim, kelas string
7
8     fmt.Print("Masukkan Nama: ")
9     fmt.Scanln(&nama)
10
11    fmt.Print("Masukkan NIM: ")
12    fmt.Scanln(&nim)
13
14    fmt.Print("Masukkan Kelas: ")
15    fmt.Scanln(&kelas)
16
17    fmt.Println("Perkenalkan saya adalah", nama+",
18               salah satu mahasiswa Prodi S1-IF dari kelas "+kelas+" dengan NIM "+nim)
19 }

```

```

PS E:\Coding\VALPRO> go run "e:\Coding\VALPRO\Biodata.go"
Masukkan Nama: Eric
Masukkan NIM: 109082500197
Masukkan Kelas: IF-13-04
Perkenalkan saya adalah Eric, salah satu mahasiswa Prodi S1-IF dari kelas IF-13-04 dengan NIM 109082500197
PS E:\Coding\VALPRO>

```

Deskripsi program

1) import "fmt"

- Mengimpor paket `fmt`, yang berfungsi untuk input/output.
- `fmt.Print` untuk menampilkan teks ke layar tanpa newline.
- `fmt.Scanln` untuk membaca input dari keyboard.
- `fmt.Println` untuk menampilkan teks ke layar dengan newline otomatis.

2) func main()

- Titik masuk eksekusi program.
- Semua kode yang ada di dalamnya akan dijalankan.

3) Input data

- Menampilkan pesan "Masukkan Nama: ".
- `fmt.Scanln(&nama)` membaca input dari user dan menyimpannya ke variabel `nama`.
- Digunakan agar input disimpan langsung ke alamat memori variabel (pass by reference).

3. Tugas 3

Source code

```
package main

import (
    "fmt"
    "math"
)

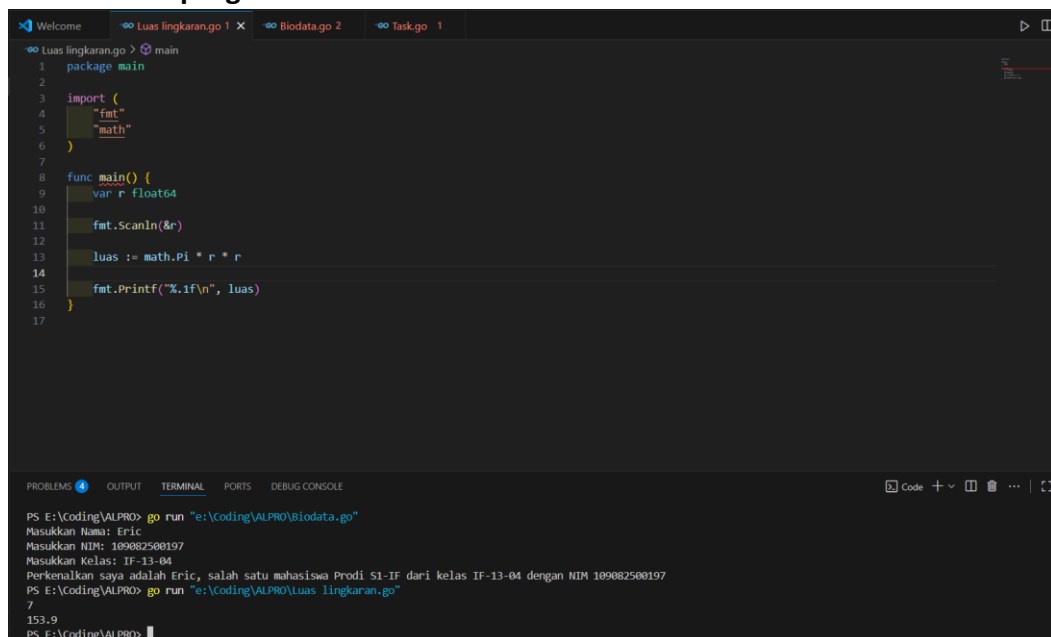
func main() {
    var r float64

    fmt.Scanln(&r)

    luas := math.Pi * r * r

    fmt.Printf("%.1f\n", luas)
}
```

Screenshoot program



```
PS E:\Coding\ALPRO> go run "e:\Coding\ALPRO\Biodata.go"
Masukkan Nama: Eric
Masukkan NIM: 109082500197
Perkenalkan saya adalah Eric, salah satu mahasiswa Prodi S1-IF dari kelas IF-13-04 dengan NIM 109082500197
PS E:\Coding\ALPRO> go run "e:\Coding\ALPRO\Luas lingkaran.go"
7
153.9
PS E:\Coding\ALPRO>
```

Deskripsi program

1) var r float64

- Mendeklarasikan variabel r bertipe float64.
- r digunakan untuk menyimpan nilai jari-jari lingkaran.
- Dipakai float64 karena perhitungan melibatkan bilangan riil (bukan integer).

2) fmt.Scanln(&r)

- Membaca input dari user (angka jari-jari lingkaran) dan menyimpannya ke variabel r.
- Tanda & menunjukkan alamat variabel, supaya nilai input disimpan langsung ke r.

3) fmt.Printf("%.1f\n", luas)

- Menampilkan hasil luas ke layar.
- %.1f berarti format float dengan 1 angka di belakang koma.
- \n untuk membuat baris baru setelah hasil ditampilkan.

4. Tugas 4

Source Code

```
package main

import (
    "fmt"
)

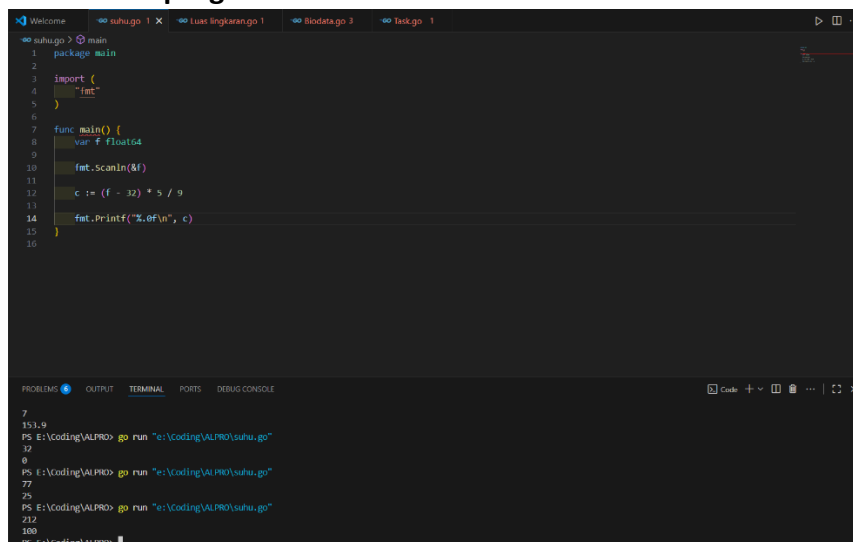
func main() {
    var f float64

    fmt.Scanln(&f)

    c := (f - 32) * 5 / 9

    fmt.Printf("%.0f\n", c)
}
```

Screenshot program



Deskripsi program

- 1) `fmt.Scanln(&f)`
 - Membaca input dari user (angka suhu Fahrenheit).
 - Tanda `&f` artinya kita memberikan alamat memori variabel `f`, sehingga nilai input disimpan langsung ke variabel itu.
- 2) `c := (f - 32) * 5 / 9`
 - Rumus konversi suhu Fahrenheit ke Celcius
- 3) `fmt.Printf("%.0f\n", c)`
 - Menampilkan hasil konversi.
 - `%.0f` artinya cetak bilangan float tanpa angka desimal (dibulatkan).
 - `\n` untuk membuat baris baru setelah hasil.

Tugas Pendahuluan

1. Jelaskan perbedaan `fmt.Print()`, `fmt.Println()`, dan `fmt.Printf()` di Go!
2. Jelaskan penggunaan tipe data `int`, `float64`, `bool`, dan `string` di Go!
3. Bagaimana Cara mendeklarasikan variabel dengan kata kunci `var` dan dengan acara singkat? Berikan contoh! Hint `“:=”`
4. Apa perbedaan antara operator `==` dan `=` dalam bahasa Go?
5. Buatlah program Go sederhana untuk meminta input nama kalian, lalu menampilkan nama.

- 1) Perbedaan `fmt.Print()`, `fmt.Println()`, dan `fmt.Printf()` di Go
`fmt.Print()`, Mencetak teks ke layar tanpa menambahkan spasi otomatis antar argumen dan tanpa newline di akhir.

Contoh:

Input : `fmt.Print("Hello") fmt.Print("World")`

Output : HelloWorld

`fmt.Println()`, Mencetak teks ke layar dengan spasi otomatis antar argumen dan menambahkan newline di akhir.

Contoh:

Input : `fmt.Println("Hello") fmt.Println("World")`

Output : Hello

World

`fmt.Printf()`, Digunakan untuk mencetak teks dengan format tertentu. Menggunakan format specifier seperti `%d` (integer), `%f` (float), `%s` (string).

Contoh:

`name := "eric"`

`age := 19`

`fmt.Printf("Nama saya %s dan umur saya %d\n", name, age)`

Output : Nama saya eric dan umur saya 19

- 2) Penggunaan tipe data `int`, `float64`, `bool`, dan `string` di Go
`Int` Bilangan bulat (positif/negatif). Contoh : `var umur int = 21`

float64 Bilangan pecahan dengan presisi tinggi. Contoh: var pi float64 = 3.14159

bool Nilai logika: true atau false. Contoh: var isStudent bool = true

string Teks atau kumpulan karakter. Contoh: var nama string = "Yura"

- 3) Cara mendeklarasikan variabel dengan kata kunci var dan cara singkat (:=)

Dengan var : var umur int = 19

var nama string = "eric"

Dengan cara singkat (:=) : Go bisa otomatis mengenali tipe data dari nilai awal.

umur := 19

nama := "eric"

- 4) Perbedaan operator == dan = dalam bahasa Go

==

Operator perbandingan (apakah dua nilai sama).

Hasilnya true atau false.

=

Operator penugasan (assignment).

Digunakan untuk menyimpan nilai ke dalam variabel.

- 5) Program Go sederhana untuk meminta input nama, lalu menampilkan nama

```
6) package main
7)
8) import "fmt"
9)
10) func main() {
11)     var nama string
12)
13)     fmt.Print("Masukkan nama kalian: ")
14)     fmt.Scanln(&nama)
15)
16)     fmt.Println("Halo,", nama)
17) }
```