

LAPORAN TUGAS PRAKTIKUM KE-4
PEMROGRAMAN BERBASIS WEB
KELAS A

Disusun Untuk Memenuhi Tugas Mata Kuliah Prak. Pemrograman Berbasis Web



Disusun oleh:

Arsya Yan Duribta

4522210117

Dosen Pengampu:

Adi Wahyu Pribadi, S.Si., M.Kom.

Program Studi Teknik Informatika
Fakultas Teknik Universitas Pancasila
2023/2024

Link Repository Github:

<https://github.com/Arsyayd11/pbw/tree/master/praktikum/pertemuan04/tugas>

Tugas 1

Lengkapi codingan dibawah ini agar run sesuai dengan perintah :

Tambahkan berupa function agar program dapat run sesuai dengan perintah (recursion function) dibawah ini :

```
package main

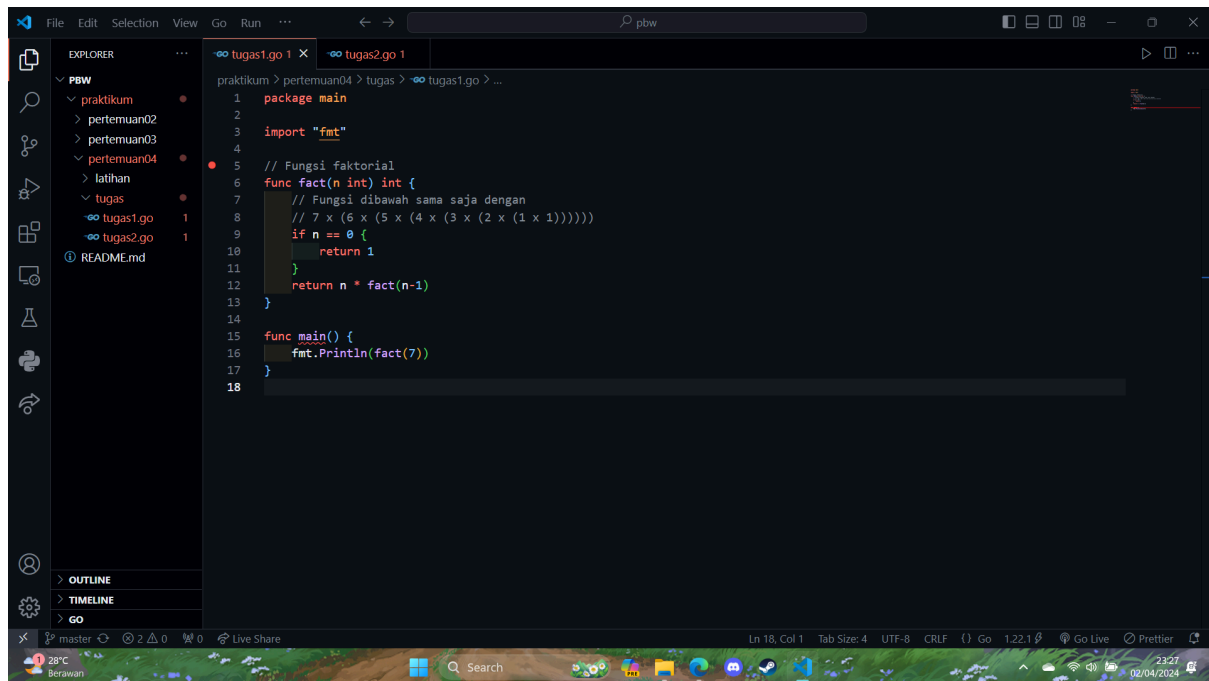
import "fmt"

func main() {
    // Fungsi dibawah sama saja dengan
    // 7 x (6 x (5 x (4 x (3 x (2 x (1 x 1))))))
    fmt.Println(fact(7))
}
```

Perbaikan yang dilakukan:

1. Definisi Fungsi “fact”: Dalam kode asli, fungsi “fact” belum didefinisikan. Fungsi ini seharusnya menghitung faktorial dari suatu bilangan.
2. Implementasi Fungsi “fact”: Fungsi “fact” diimplementasikan sebagai fungsi rekursif. Fungsi rekursif adalah fungsi yang memanggil dirinya sendiri dalam definisinya. Dalam hal ini, “fact(n)” didefinisikan sebagai “n * fact(n-1)”. Ini berarti bahwa untuk menghitung faktorial dari “n”, kita mengalikan “n” dengan faktorial dari “n-1”.

Implementasi setelah diperbaiki:

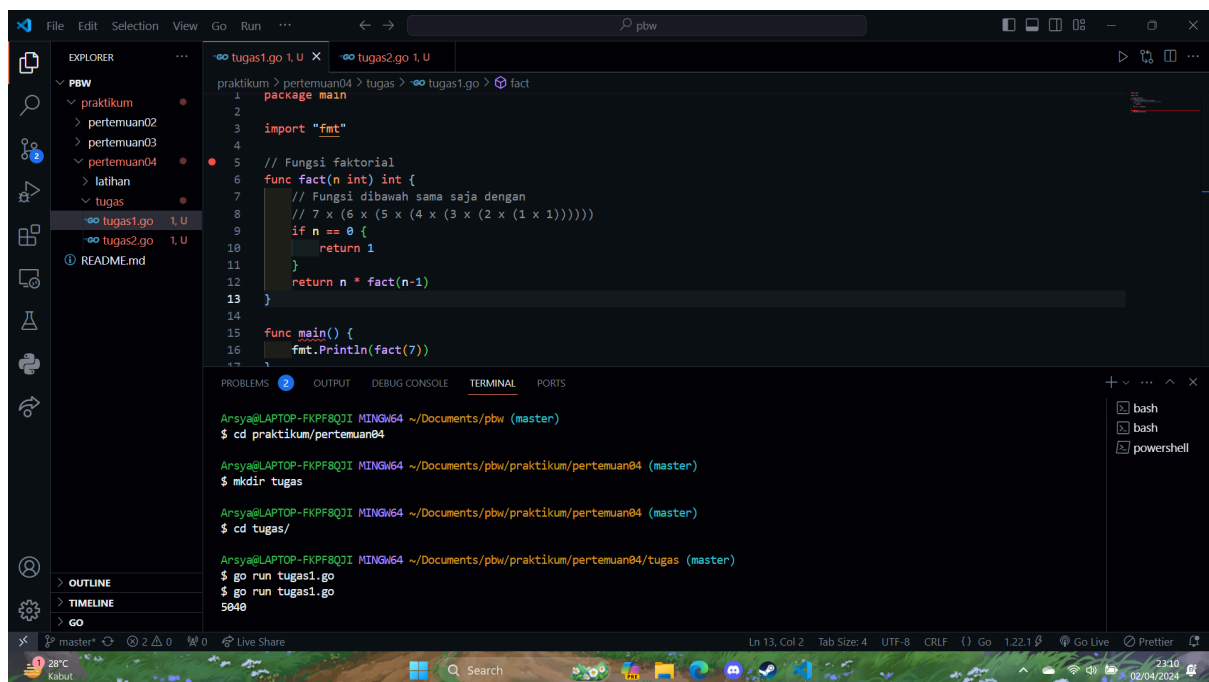


The screenshot shows the Visual Studio Code editor with a Go file named `tugas1.go`. The code implements a factorial function using recursion. The Explorer sidebar on the left shows a project structure with folders `praktikum`, `pertemuan02`, `pertemuan03`, `pertemuan04`, `latihan`, and `tugas`. The `tugas` folder contains `tugas1.go` and `tugas2.go`. The `tugas1.go` file is open in the editor, showing the following code:

```
1 package main
2
3 import "fmt"
4
5 // Fungsi faktorial
6 func fact(n int) int {
7     // Fungsi dibawah sama saja dengan
8     // 7 x (6 x (5 x (4 x (3 x (2 x (1 x 1))))))
9     if n == 0 {
10         return 1
11     }
12     return n * fact(n-1)
13 }
14
15 func main() {
16     fmt.Println(fact(7))
17 }
18
```

The status bar at the bottom indicates the file is at line 18, column 1, with a tab size of 4, UTF-8 encoding, and CRLF line endings. The system tray at the bottom shows the date and time as 23:27 on 02/04/2024.

Output ketika di run:



The screenshot shows the Visual Studio Code editor with the same Go file `tugas1.go` open. The terminal window at the bottom shows the output of the program. The terminal output is as follows:

```
Arsya@LAPTOP-FKPF8QJI MINGW64 ~/Documents/pbw (master)
$ cd praktikum/pertemuan04

Arsya@LAPTOP-FKPF8QJI MINGW64 ~/Documents/pbw/praktikum/pertemuan04 (master)
$ mkdir tugas

Arsya@LAPTOP-FKPF8QJI MINGW64 ~/Documents/pbw/praktikum/pertemuan04 (master)
$ cd tugas/

Arsya@LAPTOP-FKPF8QJI MINGW64 ~/Documents/pbw/praktikum/pertemuan04/tugas (master)
$ go run tugas1.go
5040
```

The terminal output shows the directory structure being created and the program being run, resulting in the output `5040`. The status bar at the bottom indicates the file is at line 13, column 2, with a tab size of 4, UTF-8 encoding, and CRLF line endings. The system tray at the bottom shows the date and time as 23:10 on 02/04/2024.

Tugas 2

Lengkapi codingan dibawah ini, agar dapat berjalan :

```
package main

import "fmt"

// Mengurutkan array dalam satu kali perulangan
func main() {
    fmt.Println("Bubble Short")

    var arrayNumber [20]int

    for i < len(arrayNumber) {

        if i != (len(arrayNumber) - 1) {
            if arrayNumber[i] > arrayNumber[i+1] {
                x := arrayNumber[i]
                arrayNumber[i] = arrayNumber[i+1]
                arrayNumber[i+1] = x
                i--
            }
        }

        if i > 0 {
            if arrayNumber[i] < arrayNumber[i-1] {
                x := arrayNumber[i]
                arrayNumber[i] = arrayNumber[i-1]
                arrayNumber[i-1] = x
                i--
            }
        }

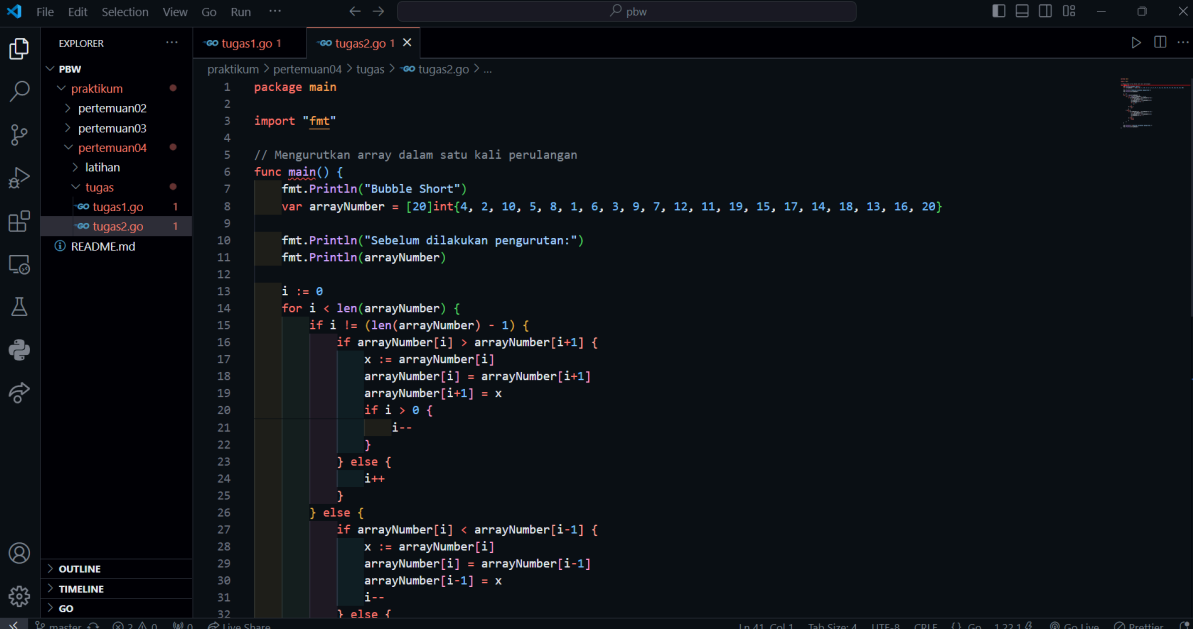
        i++
    }
}
```

```
    fmt.Println("Setelah dilakukan pengurutan.")
    fmt.Println(arrayNumber)
}
```

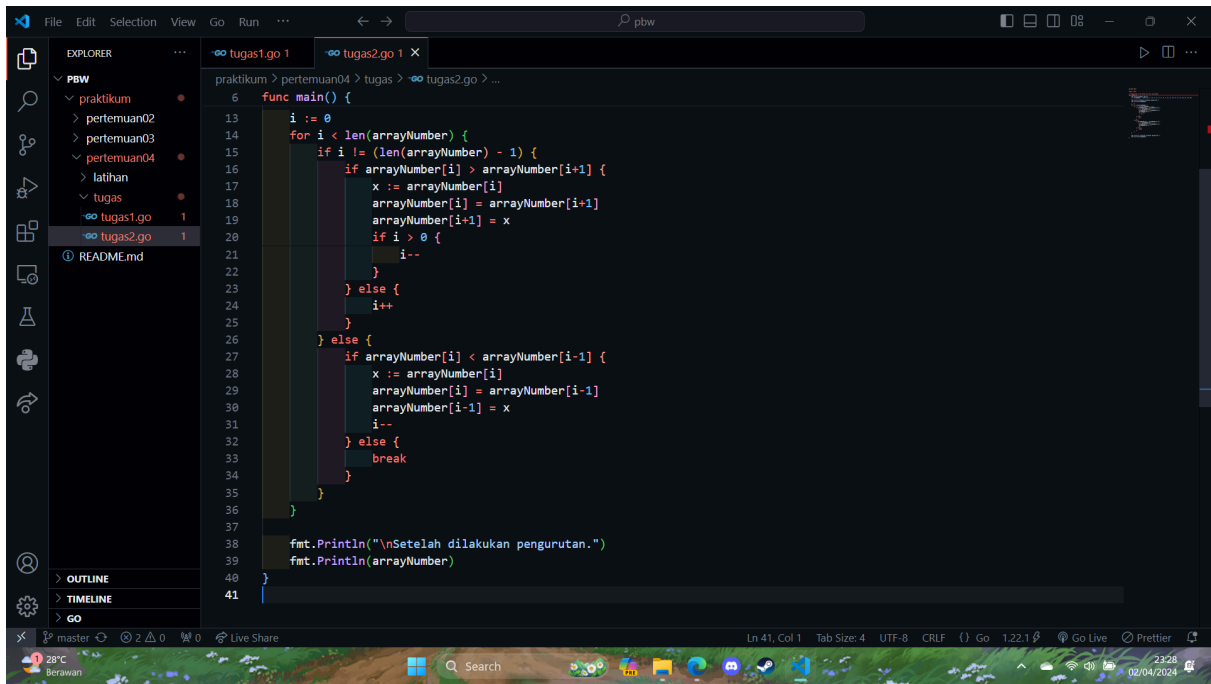
Perbaikan yang dilakukan:

1. Mengisi Variabel “arrayNumber”: variabel “arrayNubber” diinisialisasi tetapi tidak diisi dengan nilai apa pun, jadi jika menjalankan program ini, itu akan mengurutkan array kosong. Menambahkan isi array acak seperti berikut {4, 2, 10, 5, 8, 1, 6, 3, 9, 7, 12, 11, 19, 15, 17, 14, 18, 13, 16, 20}.
2. Inisialisasi Variabel “i”: Dalam kode asli, variabel “i” tidak diinisialisasi. Ini menyebabkan error saat mencoba menjalankan kode karena i tidak didefinisikan. Oleh karena itu, saya menambahkan “i := 0” sebelum loop for untuk menginisialisasi “i”.
3. Menambahkan Kondisi *else*: Dalam kode asli, jika elemen array tidak perlu ditukar yaitu, jika “arrayNumber[i] <= arrayNumber[i+1]”, tidak ada yang terjadi, dan loop *for* berlanjut ke iterasi berikutnya tanpa mengubah “i”. Ini berarti bahwa jika dua elemen sudah dalam urutan yang benar, “i” tidak akan pernah bertambah, dan loop akan berjalan selamanya. Untuk mengatasi ini, saya menambahkan blok *else* yang menambahkan “i” jika tidak ada tukar posisi yang terjadi.
4. Menambahkan *break*: Dalam kode asli, jika “i” mencapai akhir array yaitu, jika “i == len(arrayNumber) - 1”, loop akan terus berjalan tanpa henti, karena tidak ada yang mengubah “i”. Untuk mengatasi ini, saya menambahkan pernyataan *break* untuk keluar dari loop jika kita telah mencapai akhir array dan tidak ada lagi elemen yang perlu ditukar.

Implementasi setelah diperbaiki:

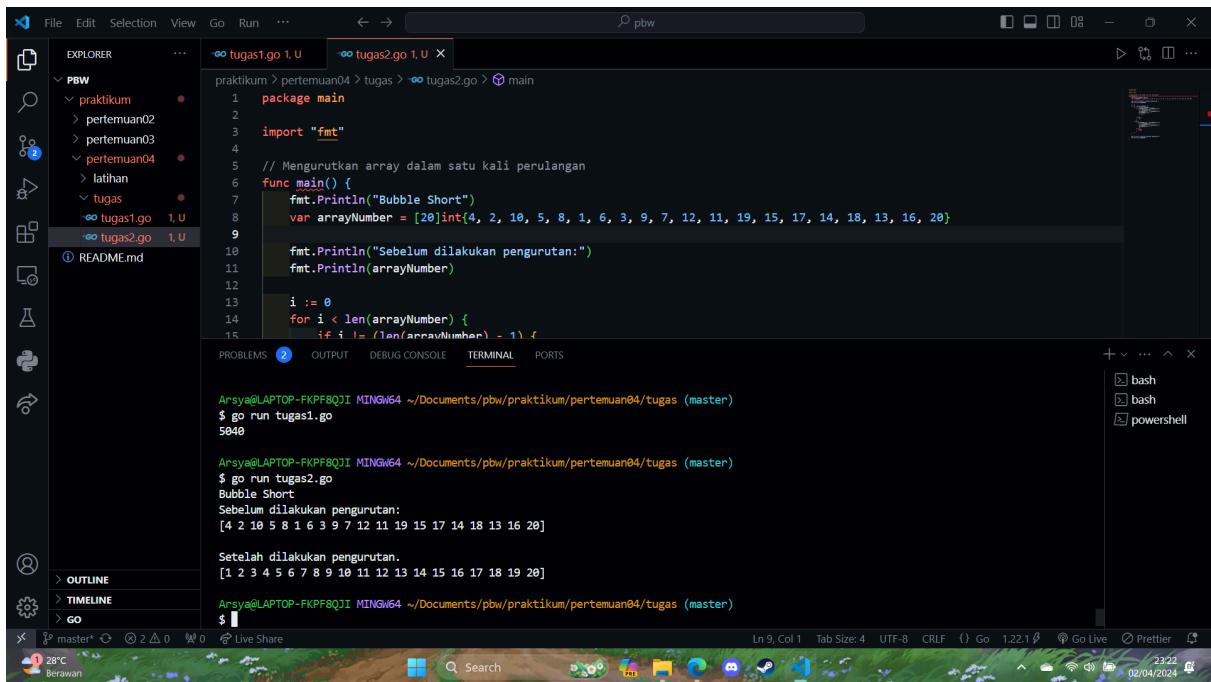


```
1 package main
2
3 import "fmt"
4
5 // Mengurutkan array dalam satu kali perulangan
6 func main() {
7     fmt.Println("Bubble Sort")
8     var arrayNumber = [20]int{4, 2, 10, 5, 8, 1, 6, 3, 9, 7, 12, 11, 19, 15, 17, 14, 18, 13, 16, 20}
9
10    fmt.Println("Sebelum dilakukan pengurutan:")
11    fmt.Println(arrayNumber)
12
13    i := 0
14    for i < len(arrayNumber) - 1 {
15        if i != (len(arrayNumber) - 1) {
16            if arrayNumber[i] > arrayNumber[i+1] {
17                x := arrayNumber[i]
18                arrayNumber[i] = arrayNumber[i+1]
19                arrayNumber[i+1] = x
20                if i > 0 {
21                    i--
22                }
23            } else {
24                i++
25            }
26        } else {
27            if arrayNumber[i] < arrayNumber[i-1] {
28                x := arrayNumber[i]
29                arrayNumber[i] = arrayNumber[i-1]
30                arrayNumber[i-1] = x
31                i--
32            } else {
33                break
34            }
35        }
36    }
37}
```



```
6 func main() {
13     i := 0
14     for i < len(arrayNumber) {
15         if i != (len(arrayNumber) - 1) {
16             if arrayNumber[i] > arrayNumber[i+1] {
17                 x := arrayNumber[i]
18                 arrayNumber[i] = arrayNumber[i+1]
19                 arrayNumber[i+1] = x
20                 if i > 0 {
21                     i--
22                 }
23             } else {
24                 i++
25             }
26         } else {
27             if arrayNumber[i] < arrayNumber[i-1] {
28                 x := arrayNumber[i]
29                 arrayNumber[i] = arrayNumber[i-1]
30                 arrayNumber[i-1] = x
31                 i--
32             } else {
33                 break
34             }
35         }
36     }
37
38     fmt.Println("\nSetelah dilakukan pengurutan.")
39     fmt.Println(arrayNumber)
40
41 }
```

Output ketika di run:



```
1 package main
2
3 import "fmt"
4
5 // Mengurutkan array dalam satu kali perulangan
6 func main() {
7     fmt.Println("Bubble Short")
8     var arrayNumber = [20]int{4, 2, 10, 5, 8, 1, 6, 3, 9, 7, 12, 11, 19, 15, 17, 14, 18, 13, 16, 20}
9
10    fmt.Println("Sebelum dilakukan pengurutan:")
11    fmt.Println(arrayNumber)
12
13    i := 0
14    for i < len(arrayNumber) {
15        if i != (len(arrayNumber) - 1) {
```

PROBLEMS OUTPUT DEBUG CONSOLE TERMINAL PORTS

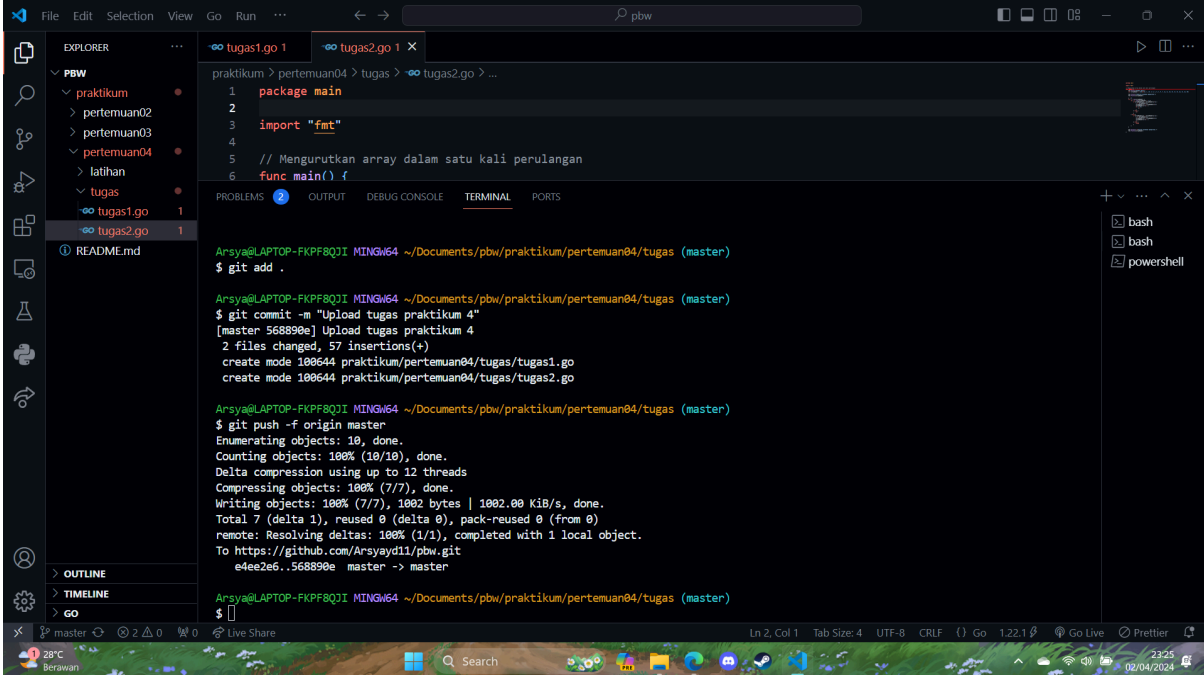
```
Arsya@LAPTOP-FKPF8QJI MINGW64 ~/Documents/pbw/praktikum/pertemuan04/tugas (master)
$ go run tugas1.go
5040

Arsya@LAPTOP-FKPF8QJI MINGW64 ~/Documents/pbw/praktikum/pertemuan04/tugas (master)
$ go run tugas2.go
Bubble Short
Sebelum dilakukan pengurutan:
[4 2 10 5 8 1 6 3 9 7 12 11 19 15 17 14 18 13 16 20]

Setelah dilakukan pengurutan.
[1 2 3 4 5 6 7 8 9 10 11 12 13 14 15 16 17 18 19 20]

Arsya@LAPTOP-FKPF8QJI MINGW64 ~/Documents/pbw/praktikum/pertemuan04/tugas (master)
$
```

Upload Ke Git



The screenshot shows the Visual Studio Code interface with a Go project. The Explorer pane on the left shows the project structure: `pbw` (parent), `praktikum` (child), and `pertemuan04` (child of `praktikum`). Inside `pertemuan04`, there are `tugas` (parent), `tugas1.go` (child), and `tugas2.go` (child). The main editor shows the content of `tugas1.go`, which is a Go program with a `main` function that prints "Hello, World!". The terminal pane at the bottom shows the following commands and output:

```
Arsya@LAPTOP-FKPF8QJI MINGW64 ~/Documents/pbw/praktikum/pertemuan04/tugas (master)
$ git add .
$ git commit -m "Upload tugas praktikum 4"
[master 568890e] Upload tugas praktikum 4
2 files changed, 67 insertions(+)
create mode 100644 praktikum/pertemuan04/tugas/tugas1.go
create mode 100644 praktikum/pertemuan04/tugas/tugas2.go

Arsya@LAPTOP-FKPF8QJI MINGW64 ~/Documents/pbw/praktikum/pertemuan04/tugas (master)
$ git push -f origin master
Enumerating objects: 10, done.
Counting objects: 100% (10/10), done.
Delta compression using up to 12 threads
Compressing objects: 100% (7/7), done.
Writing objects: 100% (7/7), 1002 bytes | 1002.00 KiB/s, done.
Total 7 (delta 1), reused 0 (delta 0), pack-reused 0 (from 0)
remote: Resolving deltas: 100% (1/1), completed with 1 local object.
To https://github.com/Arsyayd11/pbw.git
e4ee2e6..568890e master -> master

Arsya@LAPTOP-FKPF8QJI MINGW64 ~/Documents/pbw/praktikum/pertemuan04/tugas (master)
$
```

Kesimpulan

Dengan melakukan perbaikan pada kode program Golang, baik untuk fungsi faktorial maupun menggunakan metode Bubble Sort, kini kode tersebut bisa di run dan menjadi lebih efisien, jelas, dan terstruktur. Perbaikan ini meningkatkan keterbacaan dan pemahaman tentang algoritma yang diimplementasikan, serta memperbaiki masalah-masalah pada kode.

Link Repository Github:

<https://github.com/Arsyayd11/pbw/tree/master/praktikum/pertemuan04/tugas>