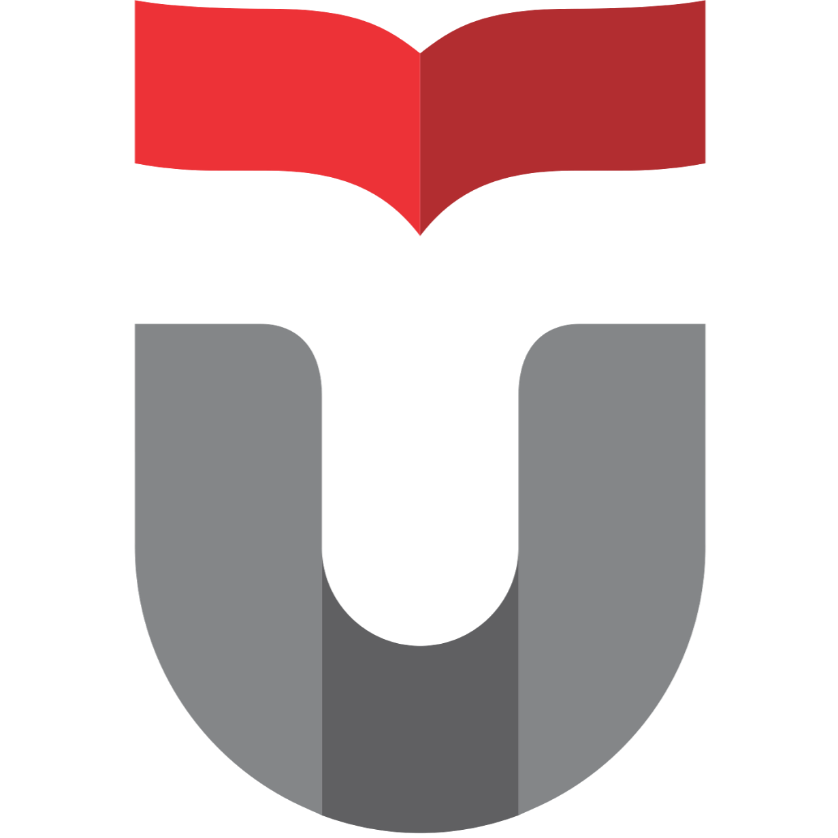
**LAPORAN PRAKTIKUM  
PEMROGRAMAN BERORIENTASI OBJEK**

**MODUL IV**

 **PROSEDUR**

Oleh:

AULIA RADIX PUTRA WINARKO

2311102056

S1IF-11-02

**S1 TEKNIK INRMATIKA**

**INSTITUT TEKNOLOGI TLKOM PURWOKERTO**

**2024**

1. **DASAR TEORI**
2. **Definisi Procedure**

Prosedur dapat dianggap sebagai potongan beberapa instruksi program menjadi suatu Instruksi baru yang dibuat untuk mengurangi kerumitan dari kode program yang kompleks pada suatu program yang besar. Prosedur akan menghasilkan suatu akibat atau efek langsung pada program ketika dipanggil pada program utama. Suatu subprogram dikatakan prosedur apabila:

1. Tidak ada deklarasi tipe nilai yang dikembalikan, dan
2. Tidak terdapat kata kunci return dalam badan subprogram.

Kedudukannya prosedur sama seperti instruksi dasar yang sudah ada sebelumnya (assignment) dan/atau instruksi yang berasal dari paket (fmt), seperti fmt.Scan dan fmt.Print. Karena itu selalu pilih nama prosedur yang berbentuk kata kerja atau sesuatu yang merepresentasikan proses sebagai nama dari prosedur. Contoh: cetak, hitungRerata, cariNilal, belok, mulal, ...

1. **Cara Pemanggilan Procedure**

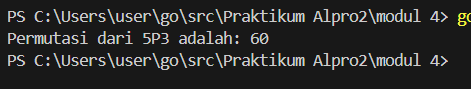
Seperti yang sudah dijelaskan sebelumnya, suatu prosedur hanya akan dieksekusi apabila dipanggil baik secara langsung atau tidak langsung oleh program utama. Tidak langsung di sini maksudnya adalah prosedur dipanggil oleh program utama melalu perantara subprogram yang lain.

Pemanggilan suatu prosedure cukup mudah, yaitu dengan hanya menuliskan nama beserta parameter atau argumen yang diminta dari suatu prosedur. Sebagai contoh prosedur cetakNFibo di atas dipanggil dengan menuliskan namanya, kemudian sebuah variabel atau nilai integer tertentu sebagai argument

1. **GUIDED**
2. **Source Code**

|  |
| --- |
| package main  import "fmt"  // Fungsi untuk menghitung faktorial  func factorial(n int) int {      if n == 0 {          return 1      }      result := 1      for i := 1; i <= n; i++ {          result \*= i      }      return result  }  // Prosedur untuk menghitung dan menampilkan permutasi  func permutasi(n, r int) {      hasilPermutasi := factorial(n) / factorial(n-r)      fmt.Printf("Permutasi dari %dP%d adalah: %d\n", n, r, hasilPermutasi)  }  func main() {      // Memanggil prosedur untuk menghitung dan menampilkan permutasi      n, r := 5, 3      permutasi(n, r)  } |

**Output**



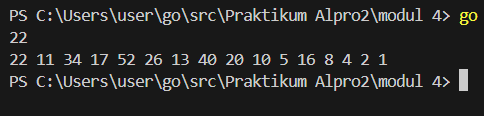
**Keterangan**

Kode diatas menghitung permutasi dari n yang diambil sebanyak r menggunakan faktorial, dan kemudian menampilkan hasilnya.

1. **UNGUIDED**
2. **Source Code**

|  |
| --- |
| package main  import "fmt"  func factorial(n int) {      fmt.Print(n, " ")      for {          if n%2==0 {              n = n/2              fmt.Print(n, " ")          } else if n==1 {              break          }else {              n = n\*3+1              fmt.Print(n, " ")          }      }  }  func main() {      var n int      fmt.Scan(&n)      if n<1000000 {          factorial(n)      }  } |

**Output**



**Keterangan**

Program di atas adalah program untuk menghitung permutasi dari dua bilangan. Di dalamnya terdapat fungsi faktorial dan prosedur permutasi. Fungsi faktorial berfungsi untuk menghitung nilai faktorial dengan melakukan perulangan sebanyak nilai n, di mana setiap iterasi, nilai i dikalikan dengan variabel result, dan hasil akhirnya dikembalikan sebagai nilai dari result.