

Nama: Mulia Akbar Nanda Pratama

Kelas: IF-12-01

NIM: 103112400034

LATIHAN 1

```
1  package main
2
3  import (
4      |   "fmt"
5  )
6
7  func main() {
8      |   var n int
9      |   fmt.Println("n: ")
10     |   fmt.Scan(&n)
11     |   nilai := 85
12
13
14     |   if nilai >= 90 {
15     |       |   fmt.Println("Nilai A")
16     |   } else if nilai >= 80 {
17     |       |   fmt.Println("Nilai B")
18     |   } else if nilai >= 70 {
19     |       |   fmt.Println("Nilai C")
20     |   } else if nilai >= 60 {
21     |       |   fmt.Println("Nilai D")
22     |   } else {
23     |       |   fmt.Println("Nilai E")
24     |   }
25
26     |   // 2. Struktur perulangan for (seperti while)
27     |   fmt.Println("\nContoh for sebagai while:")
28     |   counter := 1
29     |   for counter <= 5 {
30     |       |   fmt.Printf("Iterasi ke-%d\n", counter)
31     |       |   counter++
32     |   }
33 }
```

```

1 package main
2
3 // Fungsi dengan parameter dan return value
4 func hitungLuasLingkaran(jariJari float64) float64 {
5     return math.Pi * jariJari * jariJari // (1) Lengkapi rumus luas lingkaran
6 }
7
8 // Fungsi dengan multiple return values
9 func minMax(angka []int) (int, int) {
10     if len(angka) == 0 {
11         return 0, 0 // (2) Pastikan return value yang benar jika array kosong
12     }
13
14     min := angka[0]
15     max := angka[0]
16
17     for _, nilai := range angka {
18         if nilai < min {
19             min = nilai // (3) Lengkapi agar min selalu mendapat nilai terkecil
20         }
21         if nilai > max {
22             max = nilai // (4) Lengkapi agar max selalu mendapat nilai terbesar
23         }
24     }
25
26     return min, max
27 }
28

```

```

1  package main
2
3  import (
4      |   "fmt"
5  )
6
7  // Rekursif untuk menghitung faktorial
8  func faktorial(n int) int {
9      |   // Basis/kondisi penghentian rekursi
10     |   if n == 0 || n == 1 {
11     |       |   return 1
12     |   }
13     |   // Langkah rekursif
14     |   return n * faktorial(n-1) // (1) Lengkapi bagian ini
15 }
16
17 // Rekursif untuk menghitung bilangan Fibonacci
18 func fibonacci(n int) int {
19     |   if n <= 1 {
20     |       |   return n
21     |   }
22     |   return fibonacci(n-1) + fibonacci(n-2) // (2) Lengkapi bagian ini
23 }
24
25 // Rekursif untuk menghitung pangkat
26 func pangkat(base int, eksponen int) int {
27     |   if eksponen == 0 {
28     |       |   return 1
29     |   }
30     |   return _____ // (3) Lengkapi bagian ini
31 }
32
33 // Rekursif untuk mengecek palindrome
34 func isPalindrome(s string) bool {
35     |   if len(s) <= 1 {
36     |       |   return true
37     |   }
38     |   if s[0] != s[len(s)-1] {
39     |       |   return false
40     |   }
41     |   return _____ // (4) Lengkapi bagian ini
42 }
43
44 // Rekursif dengan helper function (untuk menghitung jumlah elemen array)
45 func sum(arr []int) int {
46     |   return _____ // (5) Lengkapi bagian ini
47 }
48
49 func sumHelper(arr []int, index int) int {
50     |   if index >= len(arr) {
51     |       |   return 0
52     |   }
53     |   return _____ // (6) Lengkapi bagian ini
54 }
55

```

```

53 |     return fibonacci(i-1) + fibonacci(i-2) // (6) Lengkapi bagian ini
54 | }
55 |
56 | func main() {
57 |     // Contoh penggunaan rekursif faktorial
58 |     fmt.Printf("Faktorial 5 = %d\n", faktorial(5))
59 |
60 |     // Contoh penggunaan rekursif fibonacci
61 |     fmt.Println("Deret Fibonacci:")
62 |     for i := 0; i < 10; i++ {
63 |         | fmt.Printf("%d ", fibonacci(i)) // (7) Lengkapi bagian ini
64 |     }
65 |     fmt.Println()
66 |
67 |     // Contoh penggunaan rekursif pangkat
68 |     fmt.Printf("2 pangkat 8 = %d\n", fibonacci(i)) // (8) Lengkapi bagian ini
69 |
70 |     // Contoh penggunaan rekursif palindrome
71 |     kata1 := "katak"
72 |     kata2 := "mobil"
73 |     fmt.Printf("Apakah '%s' palindrome? %t\n", kata1, (kata1)) // (9) Lengkapi bagian ini
74 |     fmt.Printf("Apakah '%s' palindrome? %t\n", kata2, (kata2)) // (10) Lengkapi bagian ini
75 |
76 |     // Contoh penggunaan rekursif dengan helper function
77 |     angka := []int{1, 2, 3, 4, 5}
78 |     fmt.Printf("Jumlah elemen array = %d\n", sum(angka))
79 | }
80 |

```