

TUGAS PRAKTIKUM INDIVIDU MODUL 4

Andrian Agusitinus Lumban Gaol

121450090

2023-12-05

TUGAS

1. Buatlah Tabel Perkalian sampai 10

```
# Fungsi untuk membuat tabel perkalian hingga suku ke-n
tabel_perkalian <- function(n) {
  # Menampilkan judul tabel
  cat("Tabel Perkalian", n, ":\n")

  # Loop untuk iterasi baris
  for (i in 1:n) {
    # Loop untuk iterasi kolom (hingga 10)
    for (j in 1:10) {
      # Menghitung hasil perkalian
      hasil <- i * j

      # Menampilkan hasil perkalian dalam format yang sesuai
      cat(i, "x", j, "=", hasil, "\t")
    }

    # Pindah ke baris berikutnya setelah satu baris selesai
    cat("\n")
  }
}
```

Memanggil fungsi untuk membuat tabel perkalian hingga suku ke-10

```
tabel_perkalian(10)
```

Tabel Perkalian 10 :

## 1 x 1 = 1	1 x 2 = 2	1 x 3 = 3	1 x 4 = 4	1 x 5 = 5	1 x 6 = 6	1
x 7 = 7	1 x 8 = 8	1 x 9 = 9	1 x 10 = 10			
## 2 x 1 = 2	2 x 2 = 4	2 x 3 = 6	2 x 4 = 8	2 x 5 = 10	2 x 6 = 12	2
x 7 = 14	2 x 8 = 16	2 x 9 = 18	2 x 10 = 20			
## 3 x 1 = 3	3 x 2 = 6	3 x 3 = 9	3 x 4 = 12	3 x 5 = 15	3 x 6 = 18	3
x 7 = 21	3 x 8 = 24	3 x 9 = 27	3 x 10 = 30			
## 4 x 1 = 4	4 x 2 = 8	4 x 3 = 12	4 x 4 = 16	4 x 5 = 20	4 x 6 = 24	4
x 7 = 28	4 x 8 = 32	4 x 9 = 36	4 x 10 = 40			
## 5 x 1 = 5	5 x 2 = 10	5 x 3 = 15	5 x 4 = 20	5 x 5 = 25	5 x 6 = 30	5
x 7 = 35	5 x 8 = 40	5 x 9 = 45	5 x 10 = 50			

```
## 6 x 1 = 6    6 x 2 = 12    6 x 3 = 18    6 x 4 = 24    6 x 5 = 30    6 x 6 = 36    6
x 7 = 42    6 x 8 = 48    6 x 9 = 54    6 x 10 = 60
## 7 x 1 = 7    7 x 2 = 14    7 x 3 = 21    7 x 4 = 28    7 x 5 = 35    7 x 6 = 42    7
x 7 = 49    7 x 8 = 56    7 x 9 = 63    7 x 10 = 70
## 8 x 1 = 8    8 x 2 = 16    8 x 3 = 24    8 x 4 = 32    8 x 5 = 40    8 x 6 = 48    8
x 7 = 56    8 x 8 = 64    8 x 9 = 72    8 x 10 = 80
## 9 x 1 = 9    9 x 2 = 18    9 x 3 = 27    9 x 4 = 36    9 x 5 = 45    9 x 6 = 54    9
x 7 = 63    9 x 8 = 72    9 x 9 = 81    9 x 10 = 90
## 10 x 1 = 10   10 x 2 = 20    10 x 3 = 30    10 x 4 = 40    10 x 5 = 50
10 x 6 = 60    10 x 7 = 70    10 x 8 = 80    10 x 9 = 90    10 x 10 = 100
```

2. Buatlah suatu fungsi yang menampilkan formulasi pendugaan nilai z (jumlah derat tak hingga) sebagai berikut

$$z = 1 + \frac{5}{6} + \frac{7}{11} + \frac{9}{18} + \dots$$

```
# Fungsi untuk menghitung nilai z
calculate_z <- function(n) {
  # Inisialisasi nilai awal z
  z <- 1

  # Loop untuk menghitung nilai z hingga suku ke-n
  for (i in 2:n) {
    # Penambahan suku ke-n sesuai pola
    z <- z + (2*i + 1)/(3*i + 5)
  }

  return(z)
}

# Menampilkan hasil perhitungan nilai z untuk suku ke-n (misal, n = 10)
n_value <- 10
result <- calculate_z(n_value)

# Menampilkan hasil
cat("Nilai z untuk suku ke-", n_value, "adalah:", result)

## Nilai z untuk suku ke- 10 adalah: 5.956055
```

3. Berapa nilai $f(7)$ berdasarkan fungsi rekursif berikut (tunjukkan urutan perhitungannya):

$$F(n) = \begin{cases} n + 1 & \text{jika } n = 1 \\ 2 + f(n - 1), & \text{jika } n = 2 \\ \sum_{k=2}^n f(k - 1) + f(n - k + 1) & \text{jika } n > 2 \end{cases}$$

```

# Fungsi rekursif f dengan parameter n
f <- function(n){
  # Kasus dasar: jika n = 1, kembalikan nilai (n + 1)
  if(n == 1){
    return(n + 1)
  }
  # Kasus khusus untuk n = 2
  else if(n == 2){
    # Kembalikan nilai 2 ditambah dengan hasil rekursif f(n-1)
    return(2 + f(n-1))
  }
  # Kasus umum untuk n > 2
  else{
    # Inisialisasi vektor kosong untuk elemen-elemen di sebelah kiri dan
    kanan
    vektor_kiri <- c()
    vektor_kanan <- c()

    # Loop untuk mengisi vektor_kiri dan vektor_kanan
    for (k in 2:n) {
      vektor_kiri <- append(vektor_kiri, f(k-1))
      vektor_kanan <- append(vektor_kanan, f(n-k+1))
    }

    # Kembalikan hasil penjumlahan elemen-elemen vektor_kiri dan vektor_kanan
    return(sum(vektor_kiri + vektor_kanan))
  }
}

# Memanggil fungsi f dengan parameter n = 7
result <- f(7)
cat("Hasil f(7) adalah:", result)

## Hasil f(7) adalah: 972

```