

Lembar Kerja Praktikum KOM120C Pemrograman 11: Functional Programming IV

PETUNJUK PRAKTIKUM

Review HOF

- HOF untuk pengurutan dalam list: `sorted`, `sortBy`, dan `sortWith`.
- HOF untuk tipe data string: `split`, `length`, `reverse`, dll.
- HOF `groupBy` untuk mengelompokkan list berdasarkan ketentuan tertentu, menghasilkan kembalian berupa map.
- Compiler Scala Online: <https://onecompiler.com/scala>

Struktur data dan Container pada Scala 1. Array

- Memiliki ukuran **tetap**

```
val myArray = Array(1, 2, 3, 4, 5)
val zeros = Array.fill(5)(0) // Array(0, 0, 0, 0, 0)
```

- **Mutable** (elemennya dapat diubah)

```
myArray(2) = 10
println(myArray) // Array(1,2,10,4,5)
```

- Setiap elemen **bertipe sama**
- Mendukung penggunaan **indeks** untuk mengakses elemen (indeks pertama nol)

```
val thirdElement = myArray(2)
println("Third element: " + thirdElement)
```

2. List

- **Immutable** (elemennya tidak bisa diubah)

```
val myList = List(1, 2, 3, 4, 5)
myList(2) = 10 // Error
```

- Setiap elemen **bertipe sama**
- Mendukung operasi menambah elemen list atau menggabungkan dua list

```
val list3 = list1 ++ list2
// atau
```

```
val list3 = list1 :+ list2
// atau
val list3 = list1.concat(list2)
```

- Mendukung penggunaan **head** dan **tail** untuk mengakses elemen (biasanya diolah dengan algoritma rekursi)

```
// Fungsi rekursif untuk menjumlahkan semua elemen
dalam list integer
def sumList(list: List[Int]): Int = list match {
  case Nil => 0 // Basis kasus: list kosong, jumlahnya
adalah 0
  case head::tail => head + sumList(tail) // Rekursi:
jumlahkan kepala list dengan jumlah dari sisa list
}

// Contoh penggunaan
val myList = List(1, 2, 3, 4, 5)
val total = sumList(myList)
println("Total sum: " + total)
```

3. Set

- **Immutable** (elemennya tidak bisa diubah)
- Elemennya harus **unik** (tidak duplikat)

```
val mySet = Set(1, 2, 3, 4, 5, 1, 2)
println(mySet) // HashSet(5, 1, 2, 3, 4)
```

4. Map

- Menyimpan pasangan **kunci-nilai**
- kunci harus **unik**
- **Immutable** (pasangan kunci-nilai tidak bisa diubah)
- Diakses berdasarkan **kunci**

```
val myMap = Map(("A" -> List("A1234BCD", "A6009KID"))
, ("AB" -> List("AB9999BA", "AB1234BCD"))
, ("B" -> "B2345AAA"))
// Menyimpan bentuk data (key -> value)
// myMap("A") = "A0987AAA"
// Error karena tidak dapat diubah (immutable)
// diakses berdasarkan key/kunci

println(myMap("A")) //List(A1234BCD, A6009KID)
println(myMap("B")) //B2345AAA
```

5. Tuple

- Setiap elemen bisa memiliki **tipe data berbeda**
- **Immutable** (elemennya tidak bisa diubah)

TUGAS

1. Diberikan bilangan bulat M, N. M baris berikutnya berisikan N bilangan bulat. Buat program untuk menghitung berapa banyak dari kelompok bilangan setiap baris yang tidak ada duplikatnya,

| Contoh Input |
|--|
| 3 5 1 5 7 5 1 2 3 13 3 4 1 4 6 8 10 |
| Contoh Output |
| 1 |
| Penjelasan |
| Kelompok bilangan baris pertama dan kedua memiliki duplikat, sedangkan kelompok bilangan ketiga tidak memiliki duplikat sehingga jumlah kelompok tidak memiliki duplikat ada 1 |

Kode: https://github.com/NSP-MO/CS60_Pempro/blob/main/pempro11/1.scala

```
import scala.io.StdIn.readLine

object Main {
  def main(args: Array[String]): Unit = {
    val Array(n, m) = readLine().split(" ").map(_.toInt)
    var count = 0
    for (i <- 0 until n) {
      val input = readLine().split(" ").map(_.toInt)
      if (input.distinct.length == input.length) count += 1
    }
    println(count)
  }
}
```

Output:

```

PS C:\Users\Hp\Documents\workspace\CS60\Semester4\Pempro> scala "c:\Us
Compiling project (Scala 3.6.4, JVM (21))
Compiled project (Scala 3.6.4, JVM (21))
3 5
1 5 7 5 1
2 3 13 3 4
1 4 6 8 10
1

```

2. Diberikan bilangan bulat M, N. M baris berikutnya berisikan N bilangan bulat yang merepresentasikan matriks berdimensi M*N. Buat program untuk mentranspose matriks tersebut.

Contoh Input

```

3 3
1 2 3
4 5 6
7 8 9

```

Contoh Output

```

1 4 7
2 5 8
3 6 9

```

Kode: https://github.com/NSP-MO/CS60_Pempro/blob/main/pempro11/1.scala

```

import scala.io.StdIn.readLine

object Main {
  def main(args: Array[String]): Unit = {
    val Array(n, m) = readLine().split(" ").map(_.toInt)
    val matrix = Array.ofDim[Int](n, m)
    for (i <- 0 until n) {
      matrix(i) = readLine().split(" ").map(_.toInt)
    }
    for (j <- 0 until m) {
      for (i <- 0 until n) {
        print(matrix(i)(j) + (if (i == n - 1) "\n" else " "))
      }
    }
  }
}

```

Output:

KOM120C

Pemrograman

```
PS C:\Users\Hp\Documents\workspace\CS60\Semester4> scala "c:\Users\Hp\Documents\workspace
3 3
1 2 3
4 5 6
7 8 9
1 4 7
2 5 8
3 6 9
```

3. Zantos ingin membuat sebuah sistem untuk mencari nilai IPK. terdapat input sebanyak N baris yang berisikan NIM, Nama mahasiswa, Nama mata Kuliah, dan Huruf Mutu yang dipisahkan dengan tanda koma plus spasi (,). Nilai Dari setiap huruf mutu adalah:

| | | |
|----|---|-----|
| A | : | 4.0 |
| AB | : | 3.5 |
| B | : | 3.0 |
| BC | : | 2.5 |
| C | : | 2.0 |
| D | : | 1.0 |
| E | : | 0 |

Anggap input sudah merupakan semua mata kuliah yang diambil oleh setiap mahasiswa. Buatlah program untuk mencari IPK mahasiswa yang ada di input (output IPK dibulatkan menjadi 2 angka dibelakang koma). Output berformat nama IPK dan diurutkan berdasarkan NIM. Dipastikan tidak ada input yang memiliki id dan mata kuliah yang sama.

Contoh Input

```
5
G0401221001, Rangga Rafif, Blockchain, A G0401221002,
Raja Iblis, Blockchain, AB
G0401221003, Zantos Zantoso, Forensik, B
G0401221001, Rangga Rafif, PWN, BC
G0401221001, Rangga Rafif, Kriptografi, A
```

Contoh Output

```
Rangga Rafif 3.50
Raja Iblis 3.50
Zantos Zantoso 3.0
```

Kode: https://github.com/NSP-MO/CS60_Pempro/blob/main/pempro11/1.scala

```
import scala.io.StdIn.readLine

def nilaiMutu(nilai: String): Double = nilai match {
  case "A"  => 4.0
  case "AB" => 3.5
```

```
case "B" => 3.0
case "BC" => 2.5
case "C" => 2.0
case "D" => 1.0
case "E" => 0.0
case _ => 0.0
}

object Main {
  def main(args: Array[String]): Unit = {
    val n = readLine().toInt

    val data = scala.collection.mutable.Map[String, (String,
List[Double])]()

    for (_ <- 0 until n) {
      val input = readLine().split(" ").map(_.trim)
      val nim = input(0)
      val nama = input(1)
      val nilai = nilaiMutu(input(3))

      if (data.contains(nim)) {
        val (listNama, listNilai) = data(nim)

        if (listNama != nama) {
          data(nim) = (listNama, nilai :: listNilai)
        }
        else {
          data(nim) = (nama, nilai :: listNilai)
        }
      } else {
        data(nim) = (nama, List(nilai))
      }
    }

    data.values.foreach { case (nama, nilaiList) =>
      val ipk = nilaiList.sum / nilaiList.size
      println(f"$nama $ipk%.2f")
    }
  }
}
```

Output:

KOM120C

Pemrograman

```
PS C:\Users\Hp\Documents\workspace\CS60\Semester4> scala "c:\Users\Hp\Documents\workspace\CS60\Semester4\Pempro\11\3.scala"
Compiling project (Scala 3.6.4, JVM (21))
Compiled project (Scala 3.6.4, JVM (21))
5
G0401221001, Rangga Rafif, Blockchain, A
G0401221002, Raja Iblis, Blockchain, AB
G0401221003, Zantos Zantoso, Forensik, B
G0401221001, Rangga Rafif, PWN, BC
G0401221001, Rangga Rafif, Kriptografi, A
Rangga Rafif 3.50
Raja Iblis 3.50
Zantos Zantoso 3.00
```

4. Tung Tung Tung Sahur sedang menghitung bunyi kentung tiap pos ronda, setiap pos ronda yang melakukan kentungan sahur ditulis berdasar nomor pos ronda semisal pos ronda 1 melakukan kentungan sahur maka ditulis 1, berikutnya jika pos ronda 2 melakukan kentungan sahur ditulis 2, ketika adzan subuh berkumandang maka ditulis -1, hitung frekuensi tiap kentungan dari setiap pos ronda dan urutkan berdasarkan nomor pos rondanya.

| Contoh Input |
|--|
| 1 2 2 3 1 1 2 3 2 3 4 5 6 8 6 6 7 8 1 5 4 -1 |
| Contoh Output |
| 4, 4, 3, 2, 2, 3, 1, 2 |

Kode: https://github.com/NSP-MO/CS60_Pempro/blob/main/pempro11/1.scala

```
import scala.io.StdIn.readLine

object Main {
  def main(args: Array[String]): Unit = {
    val n = readLine().split(" ").map(_.toInt)
    val count = Array.fill(8)(0)

    for (i <- n if i != -1) {
      if (i >= 1 && i <= 8) {
        count(i - 1) += 1
      }
    }
    println(count.mkString(", "))
  }
}
```

Output:

```

PS C:\Users\Hp\Documents\workspace\CS60\Semester4> scala "c:\
Compiling project (Scala 3.6.4, JVM (21))
Compiled project (Scala 3.6.4, JVM (21))
1 2 2 3 1 1 2 3 2 3 4 5 6 8 6 6 7 8 1 5 4 -1
4, 4, 3, 2, 2, 3, 1, 2

```

5. Diberikan sebuah array bilangan bulat A dengan panjang N, serta sebuah bilangan asli M. Buat array baru B berukuran M, di mana elemen ke-i dari array B didefinisikan sebagai:

$$B_i = \sum_{k=0}^{\infty} A[i + km] \quad \text{selama } i + km < n$$

| Contoh Input |
|---------------------|
| 10 4 |
| 9 2 5 8 8 2 3 5 1 3 |
| Contoh Output |
| 18 7 8 13 |

Kode: https://github.com/NSP-MO/CS60_Pempro/blob/main/pempro11/1.scala

```

import scala.io.StdIn.readLine
import scala.io.StdIn.readInt

object Main {
  def main(args: Array[String]): Unit = {
    val Array(n, m) = readLine().split(" ").map(_.toInt)
    val input = readLine().split(" ").map(_.toInt)

    val sum = for (i <- 0 until m) yield {
      (i until n by m).map(input(_)).sum
    }
    println(sum.mkString(" "))
  }
}

```

Output:

KOM120C

Pemrograman

```
PS C:\Users\Hp\Documents\workspace\CS60\Semester4> sca  
10 4  
9 2 5 8 8 2 3 5 1 3  
18 7 8 13
```