

1. Buat program untuk membaca input berupa N buah bilangan bulat, kemudian carilah nilai terkecil, terbesar dan jumlah dari semua bilangan yang ganjil (Gunakan High Order Function).

```
import scala.io.StdIn.readLine

object Main {
  def main(args: Array[String]): Unit = {
    val n = readLine().toInt
    val a = readLine().split(" ").map(_.toInt)
    val min = a.min
    val max = a.max
    val oddSum = a.filter(_ % 2 != 0).sum
    println(s"$min $max $oddSum")
  }
}
```

```
Compiling project (Scala 3.6.4, JVM (21))
Compiled project (Scala 3.6.4, JVM (21))
5
1 -1 -9 7 5
-9 7 3
```

2. Diberikan N bilangan bulat positif. Tentukan jumlah dari bilangan tersebut yang termasuk bilangan kuadrat.

```
import scala.io.StdIn.readLine

object Main {
  def main(args: Array[String]): Unit = {
    val n = scala.io.StdIn.readInt()
    val a = readLine().split(" ").map(_.toInt)

    val a2 = a.filter(x => {
      val sr = math.sqrt(x)
      sr == sr.toInt
    })
    println(a2.sum)
  }
}
```

```
5
2 9 15 25 10
34
```

3. Buatlah program untuk menghitung ragam dari nilai ujian mahasiswa. Perhatikan bahwa terdapat nilai tidak valid (di luar 0-100) yang harus dieliminasi. Pada Input, baris pertama berupa N yang mewakili banyaknya nilai mahasiswa dan baris kedua berupa N nilai mahasiswa (note: gunakan 3 HOF, print hingga 2 angka di belakang koma).

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

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

    val score = a.filter(x => x >= 0 && x <= 100)

    val mean = score.sum / score.length.toDouble

    val variant = (for(x <- score) yield {
      val diff = x - mean
      diff * diff
    }).sum / score.length.toDouble

    println(f"$variant%.2f")
  }
}
```

```
5
90 -9 80 45 105
372.22
```

4. Buat HOF bernama filterList yang menerima sebuah koleksi bilangan bulat dan sebuah fungsi (yang menerima bilangan bulat dan mengembalikan boolean) sebagai parameter. Fungsi filterList harus mengembalikan daftar baru yang hanya berisi elemen-elemen yang memenuhi kondisi yang diberikan. Implementasikan filter dasar berikut sebagai fungsi:
1. isEven: Mengembalikan true jika bilangan genap.
 2. isOdd: Mengembalikan true jika bilangan ganjil.
 3. isPositive: Mengembalikan true jika bilangan positif.
 4. isMult(factor): Mengembalikan true jika bilangan tersebut merupakan kelipatan dari factor.

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

object Main {
  def filterList(a: Int, operation: (Int) => Boolean): Boolean = {
    operation(a)
  }
  def isEven(a: Int): Boolean = {
    a % 2 == 0
  }
  def isOdd(a: Int): Boolean = {
    a % 2 != 0
  }
  def isPositive(a: Int): Boolean = {
    a > 0
  }
  def isMult(a: Int): Boolean = {
    val sr = math.sqrt(a)
    sr == sr.toInt
  }
  def main(args: Array[String]): Unit = {
    printf("Enter a number: ")
    val n = readInt()
    println("Is the number even: " + filterList(n, isEven))
    println("Is the number odd: " + filterList(n, isOdd))
    println("Is the number positive: " + filterList(n, isPositive))
    println("Is the number a perfect square: " + filterList(n, isMult))
  }
}
```

```
Compiling project (Scala 3.6.4, JVM (21))
Compiled project (Scala 3.6.4, JVM (21))
Enter a number: -9
Is the number even: false
Is the number odd: true
Is the number positive: false
Is the number a perfect square: false
```