

Лабораторная работа № 1 «Введение в функциональное программирование на языке Scala»

19 февраля 2024 г.

Сергей Виленский, ИУ9-62Б

Цель работы

Целью данной работы является ознакомление с программированием на языке Scala на основе чистых функций.

Индивидуальный вариант

Функция `partitionP: (List[Int], Int => Boolean) => (List[Int], List[Int])`, разделяющая элементы исходного списка на два списка в зависимости от того, удовлетворяют ли они предикату.

Реализация и тестирование

Работа в REPL-интерпретаторе Scala:

```
scala> val partitionP: (List[Int], Int => Boolean) => (List[Int], List[Int]) = {
  | case (Nil, f)      => (Nil, Nil)
  | case (x :: xs, f)  =>
  |   partitionP(xs, f) match {
  |     case (as, bs) if (f(x)) => (x :: as, bs)
  |     case (as, bs)          => (as, x :: bs)
  |   }
  | }
val partitionP: (List[Int], Int => Boolean) =>
(List[Int], List[Int]) = $Lambda$1202/0x0000000080118100@6bf0f70a

scala> val test =
  partitionP(List(0, 0, 1, -2, -5, 10, -3, 0, 1, -10, 1, -2), _ % 2 == 0)
```

```
val test: (List[Int], List[Int]) =  
  (List(0, 0, -2, 10, 0, -10, -2), List(1, -5, -3, 1, 1))
```

Вывод

В ходе выполнения лабораторной работы были повторены после курса основ информатики основы функционального программирования, а конкретно были использованы такие конструкции языка программирования Scala, как функции высшего порядка, конструкция шаблонов case и match-case. В результате выполнения лабораторной работы была написана функция разделения элементов входного списка на два выходных по удовлетворимости входному предикату.