

Федеральное государственное автономное
образовательное учреждение
высшего образования
«СИБИРСКИЙ ФЕДЕРАЛЬНЫЙ УНИВЕРСИТЕТ»

Институт космических и информационных технологий
институт

Кафедра «Информатика»
кафедра

ОТЧЕТ О ЛАБОРАТОРНОЙ РАБОТЕ №3

Коллекции
Тема

Преподаватель

подпись, дата

А. А. Чикизов

инициалы, фамилия

Студент

КИ19-16/16 031939175

номер группы, зачетной книжки

подпись, дата

А. Д. Непомнящий

инициалы, фамилия

Красноярск 2022

1 Задание

Все необходимые данные (многоэлементные структура данных) хранятся в списках case-классов. Необходимо на языке Scala определить функции для добавления, удаления и "модификации" элементов списков и определить функции, демонстрирующие корректное выполнение этих запросов и операций. Структура данных: предприятие; подразделение; прибыль/убыток за последний месяц (символ «+» или «-»). Создать два запроса, позволяющих определить, предприятие в целом работало с прибылью или с убытком, и узнать, каких подразделений больше прибыльных или убыточных.

Вариант 11.

2 Исходный код основного алгоритма

Исходный код программы приведен на листинге 1.

Листинг 1 — исходный код структур данных и кода для демонстрации

```
case class Company (name: String, divisions: Array[Profit]) {
  override def toString: String = "Company " + name + ", divisions' profit last
month:\n" + divisions.mkString(" ")
}
class CompanyList {
  private val list: ListBuffer[Company] = ListBuffer()

  def this(companies: ListBuffer[Company]) = {
    this()
    companies.foreach(x => add(x))
  }

  def add(company: Company): Unit = {
    list.append(company)
  }

  def deleteByName(name: String): Unit = {
    delete(findIndexByName(name))
  }

  def isProfitable(name: String): Boolean = {
    isProfitable(findByName(name))
  }

  def countProfitable(): Int = {
    list.count(x => isProfitable(x)) - list.count(x => !isProfitable(x))
  }
}
```

Окончание листинга 1

```
private def findIndexByName(name: String): Int = {
  list.indexWhere(x => x.name == name)
}

private def findByName(name: String): Company = {
  list.find(x => x.name == name).get
}

private def delete(index: Int): Unit = {
  list.remove(index)
}

private def isProfitable(company: Company): Boolean = {
  company.divisions.count(x => x == Gain) >= company.divisions.count(x => x ==
Loss)
}

def where(filter: Company => Boolean): CompanyList = {
  CompanyList(list.filter(filter))
}

override def toString: String = list.mkString("\n")
}
@main
def Main(): Unit =
  val list = CompanyList()
  list.add(Company("General motors", Array(Gain, Loss, Loss, Loss)))
  list.add(Company("Chrysler group", Array(Gain, Gain, Loss, Loss)))
  list.add(Company("Ford Motor", Array(Loss, Gain, Gain, Gain, Gain, Gain)))
  println(list.toString)
  println(list.isProfitable("General motors"))
  println(list.isProfitable("Chrysler group"))
  println(list.isProfitable("Ford Motor"))
  println(list.countProfitable())
  println(list.where(x => x.divisions.length > 4 ).toString())
```

3 Результаты

В результате работы программы был получен следующий результат, соответствующий ожидаемому.

```
Company General motors, divisions' profit last month:
Gain Loss Loss Loss
Company Chrysler group, divisions' profit last month:
Gain Gain Loss Loss
Company Ford Motor, divisions' profit last month:
Loss Gain Gain Gain Gain Gain
false
true
true
1
Company Ford Motor, divisions' profit last month:
Loss Gain Gain Gain Gain Gain
```