CSC 369 – Assignment 5

Question 1:

```
import scala.io.Source

object NumDivisble3 {
      def main (args: Array[String]) : Unit = {
            val lines = Source.fromFile("input.txt").getLines.toList
            val integerMap = lines.flatMap(_.split("\\s+"))
            val div3Map = integerMap.filter(Integer.parseInt(_)%3 == 0)
            div3Map.distinct.foreach(x => println(x + " appears " +
                  div3Map.count(_ == x) + " times"))
      }
}
```

Question 2:

```
import scala.io._
import org.apache.spark.SparkContext._

object EmployeeDepartment {
    def main(args: Array[String]) : Unit = {
        val emps = Source.fromFile("employees.txt").getLines.toList;
        val deps = Source.fromFile("departments.txt").getLines.toList;

        val rddEmps = sc.parallelize(emps)
        val rddDeps = sc.parallelize(deps)
        val rddCart = rddEmps.cartesian(rddDeps)

        val matches = rddCart.filter {x => x._1.split(",")(1) ==
           x._2.split(",")(0)}
        matches.collect.foreach(x => println(x._1.split(",")(0) + ", "
           + x._2.split(",")(1)))
    }
}
```

Question 3:

```
import scala.io._

object GPA {
    val gradeMap = Map("A" -> 4, "B" -> 3, "C" -> 2, "D" -> 1, "F" ->
0)
    def calcGPA(line:String) : Unit = {
        var grades = line.split(", ",3)(2).split(", ")
        val gradeInfo = grades.aggregate((0, 0)) (
            (x,y) => (x._1 + gradeMap(y.split(" ")(0)), x._2 + 1),
            (x,y) => (x._1 + y._1, x._2 + y._2))

        val lineDelim = line.split(",")
        println(lineDelim(0) + "," + lineDelim(1) + ", " +
           gradeInfo._1 * 1.0 / gradeInfo._2)
    }

    def main(args: Array[String]) : Unit = {
        val lines = Source.fromFile("input.txt").getLines.toList
        lines.foreach(calcGPA)
    }
}
```