Programming Assignment 3 – Java Arrays

Due date: Monday June 5, 2017 (by midnight)

Grading: 8 points

Write an application that runs 1,000,000 games of craps and answers the following questions:

1) How many games are won on the first roll, second roll, ..., twentieth roll, and after the twentieth roll?
2) How many games are lost on the first roll, second roll, ..., twentieth roll, and after the twentieth roll?
3) What are the chances of winning at craps? [Note: You should discover that craps is one of the fairest casino games. What do you suppose this means?]
4) What is the average length of a game of craps?

Below is a reproduced copy of the Craps.java from chapter 6

Do NOT copy/paste from *.pdf to *.java files (may not work correctly)

```
// Craps.java
// Craps class simulates the dice game craps.

import java.util.Random;


public class Craps
{
    // create random number generator for use in method rollDice
    private static final Random randomNumbers = new Random();
   // enumeration with constants that represent the game status
   private enum Status { CONTINUE, WON, LOST };

    // constants that represent common rolls of the dice
   private static final int SNAKE_EYES = 2;
   private static final int TREY = 3;
   private static final int SEVEN = 7;
   private static final int YO_LEVEN = 11;
   private static final int BOX_CARS = 12;

   // plays one game of craps
   public static void main( String[] args )
    {
      int myPoint = 0; // point if no win or loss on first roll
      Status gameStatus; // can contain CONTINUE, WON or LOST
      int sumOfDice = rollDice(); // first roll of the dice
```

```
      // determine game status and point based on first roll
      switch ( sumOfDice )
      {
         case SEVEN: // win with 7 on first roll
         case YO_LEVEN: // win with 11 on first roll
            gameStatus = Status.WON;
            break;
         case SNAKE_EYES: // lose with 2 on first roll
         case TREY: // lose with 3 on first roll
         case BOX_CARS: // lose with 12 on first roll
            gameStatus = Status.LOST;
            break;
         default: // did not win or lose, so remember point
            gameStatus = Status.CONTINUE; // game is not over
            myPoint = sumOfDice; // remember the point
            System.out.printf( "Point is %d\n", myPoint );
            break; // optional at end of switch

       } // end switch

      // while game is not complete
      while ( gameStatus == Status.CONTINUE ) // not WON or LOST
      {
         sumOfDice = rollDice(); // roll dice again

         // determine game status
         if ( sumOfDice == myPoint ) // win by making point
            gameStatus = Status.WON;
         else
            if ( sumOfDice == SEVEN ) // lose by rolling 7 before point
               gameStatus = Status.LOST;
      } // end while

      // display won or lost message
      if ( gameStatus == Status.WON )
         System.out.println( "Player wins" );
      else
         System.out.println( "Player loses" );
   } // end main

   // roll dice, calculate sum and display results
   public int rollDice()
   {
      // pick random die values
      int die1 = 1 + randomNumbers.nextInt( 6 ); // first die roll
      int die2 = 1 + randomNumbers.nextInt( 6 ); // second die roll

      int sum = die1 + die2; // sum of die values

      // display results of this roll
      System.out.printf( "Player rolled %d + %d = %d\n",
                         die1, die2, sum );
      return sum; // return sum of dice
   } // end method rollDice
} // end class Craps
```