HANDOUT 13.1

**Craps.java**

import java.util.Random;

/\*\*

\* Simulates the game of craps. A modified version of Owen Astrachan's

\* program, A Computer Science Tapestry, 1997, McGraw-Hill, p. 225-231.

\* Revised on 4/10/97, Cary Matsuoka.

\* Converted to Java on 6/2/2002, George Peck.

\* Modified 1/19/06 Jason Quesenberry and Nancy Quesenberry.

\*

\*/

public class Craps{

private Random myDie;

private int myGamesWon, myGamesToPlay;

private double myPercentWon;

/\*\*

\* Constructor for the Craps object

\*/

public Craps(){

myDie = new Random();

}

/\*\*

\* Plays craps the specified number of times and evaluates

\* the number of games won and winning percentage

\*

\* @param gamesToPlay Number of trials (games)

\*/

public void playGame(int gamesToPlay){

myGamesToPlay = gamesToPlay;

myGamesWon = 0;

for (int game = 0; game < myGamesToPlay; game++){

if (winGame()){

myGamesWon++;

}

}

myPercentWon = (double) myGamesWon / myGamesToPlay \* 100;

}

/\*\*

\* Description of the Method

\*

\* @return Total number of games won

\*/

public int gamesWon(){

return myGamesWon;

}

/\*\*

\* Description of the Method

\*

\* @return Winning percentage

\*/

public double percentWon(){

return myPercentWon;

}

/\*\*

\* Gets the point attribute of the Craps object

\*

\* @param point Value to match, 2 <= point <= 12

\* @return true if point is matched, false if 7 is rolled first

\*/

private boolean getPoint(int point){// precondition: 2 <= point <= 12

int sum;

do{

sum = rollDice();

} while ((sum != point) && (sum != 7));// <--- application of DeMorgan's Laws

/\*

if (sum == point)

System.out.println("point is matched, player won");

else

System.out.println("crapped out");

\*/

return (sum == point);

}

/\*\*

\* Determines if a single game of craps is won or lost

\*

\* @return true if a single game of craps is won, otherwise false

\*/

private boolean winGame(){

int point = rollDice();

switch (point){

case 7:

case 11:

// System.out.println("7 or 11 rolled, win");

return true;

case 2:

case 3:

case 12:

// System.out.println("2, 3,or 12 rolled, lost");

return false;

default:

return getPoint(point);

}

}

/\*\*

\* Simulates the rolling of two dice

\*

\* @return sum of rolling two dice

\*/

private int rollDice(){

int first = myDie.nextInt(6) + 1;

int second = myDie.nextInt(6) + 1;

int sum = first + second;

return sum;

}

}