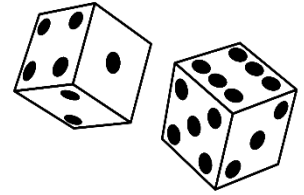


Assignment 04 – DiceWar2 (A Game for Two Players Involving Two Dice) (Due: Friday April 17, 2020)

You are probably familiar with games that involve rolling dice. For this assignment, you are to complete a Java application program, named `DiceWar2`, that plays a game of the same name. This version of the game, which was invented just the other day, involves two players who play rounds in which they each roll a die.



To play `DiceWar2`, the players must each have a die and some starting number of "chips" (or matchsticks, money, or whatever) which will serve as their "balance". The game ends when only one player has a balance greater than zero, at which point they are declared the "winner". Of course, this means that the loser has lost all their original balance to the winner.

Each round consists of the players rolling their respective dice. The player whose die has the highest face value is then "paid" by the other player an amount equal to the difference between this highest face value and the other player's die face value. If there are multiple highest face values then zero is paid, which means that that round resulted in no change to the players balances. The game continues through subsequent rounds until a winner is determined.

Of course, if a player's balance is insufficient to pay the full difference between face values then that player will only be able to pay what they have, bringing their balance to zero and meaning they lose at that point.

Consider the sample execution of the game shown to the right. You'll note that the names and initial balances of the two players are communicated to the program through "Run Arguments". The details of doing this are already included in the given version of `DiceWar2.java` provided to you, as shown below.

```

public class DiceWar2 {
    public static void main(String[] args) {
        System.out.println("DiceWar2:");
        if(args.length == 4) {
            String name1 = args[0];
            int balance1 = Integer.parseInt(args[1]);
            System.out.println(name1 + ":\t" + balance1);

            String name2 = args[2];
            int balance2 = Integer.parseInt(args[3]);
            System.out.println(name2 + ":\t" + balance2);

            // Stubbed !!!
        } else {
            System.out.println("---->ERROR: missing arguments");
        }
    }
}

```

Run Arguments: DaVinci 17 Einstein 17

Compile Messages | jGRASP Messages | Run I/O | Interactions

End | Clear | Help

Line:19 Col:1 Code:125 Top:1 OVSBLK

To activate the *Run Arguments* text box (within which you type the run argument values), you must first select the *Build* menu and check the *Run Arguments* option.

```

c:\C134>java DiceWar2 PMJ 10 Other 10
DiceWar2:
PMJ:    10
Other:  10

PMJ     rolled [5]
Other   rolled [3]
PMJ:    12
Other:   8

PMJ     rolled [2]
Other   rolled [6]
PMJ:    8
Other:  12

PMJ     rolled [4]
Other   rolled [1]
PMJ:    11
Other:   9

PMJ     rolled [2]
Other   rolled [6]
PMJ:    7
Other:  13

PMJ     rolled [1]
Other   rolled [6]
PMJ:    2
Other:  18

PMJ     rolled [6]
Other   rolled [5]
PMJ:    3
Other:  17

PMJ     rolled [2]
Other   rolled [2]
PMJ:    3
Other:  17

PMJ     rolled [3]
Other   rolled [2]
PMJ:    4
Other:  16

PMJ     rolled [1]
Other   rolled [2]
PMJ:    3
Other:  17

PMJ     rolled [1]
Other   rolled [5]
PMJ:    0
Other:  20

Other Won!!!

```

This assignment provides you an opportunity to utilize much of the previous things we have covered, and hopefully you have learned, this semester. Included among these are decomposition, modularization, variables, expressions, looping and decisions. However, the primary motivation (or goal) of this assignment is to provide an opportunity for you to make use of "instance classes", and thus objects, which means that you are required to use the given `Die.java` class in the development of this program.

Thus, you are given the files `DiceWar2.java` and `Die.java` for your use in this assignment. There is no need for you to modify and you should not modify `Die.java` in any way. Of course, `DiceWar2.java` is incomplete and it is your job to complete it so that it fully performs the tasks of playing the game as described and as shown in the given illustrations.

Be sure to fully comment the source program, by inserting the required comment block at the beginning and by also providing appropriate and useful comments for each method and important instruction sequences. When you are done you are to submit the `DiceWar2.java` file to the AS04 folder on the CMPS134 CWS. Do not submit the `Die.java` file!

Good luck,
P.M.J.