

CS150 Assignment 6

Craps!

Date assigned: Friday, October 26, 2012

Date due: Friday, November 2, 2012, 9:15am (35 points)

For this assignment, you will write a program that plays *craps*. If you aren't familiar with the game, here are the basics. Craps is played by rolling a pair of dice and using the sum of the pair as the roll's value. If one die shows 4 and the other shows 5, you have rolled a 9. On the first roll, if the value is 7 or 11, you've rolled a *natural* and win. If you roll a 2, 3, or 12 on the first roll, you *crap out* or lose. If you roll anything else, you establish a *point*. The object is to keep rolling until you roll the *point* again. However, if you roll a 7 before your *point*, you lose. If you reach the *point* without first rolling a 7, you win. You will not include betting for this assignment.

Since you will need to randomly roll the dice, you will need a pseudo random number generator. The random number generator is seeded using the current time. You must include the following line of code in your program:

```
srand(static_cast<unsigned> (time(static_cast<time_t *>(NULL))));
```

Along with adding `#include <ctime>` to your preprocessor directives.

Once you have seeded your random number generator, you can generate an integer between 1 and 6 by using the following:

```
int random = rand() % 6 + 1;
```

Minimally, you must use these three functions in your program:

1. `void drawHeading ()` – displays the heading.
2. `void drawDiceFace (int face)` – draws the die face that is passed into the parameter.
3. `int rollDice ()` – rolls two random dice, displays them to the screen, and returns the roll value, which is the sum of the two dice faces.

You may add other functions if you like.

After displaying every roll, your program must pause then clear the screen after the user enters a key. You do this by including:

```
system ("pause");  
system ("cls");
```

Sample Runs:

Run 1:

Screen 1

```
*****
*           Craps           *
*****

You rolled: 8
-----
|o o|
| o |
|o o|
-----
-----
|o  |
| o |
|  o|
-----

You must roll a 8 to win. If you roll a 7, you lose.

Press any key to continue . . .
```

Screen 2

```
*****
*           Craps           *
*****

You rolled: 7
-----
|  |
| o |
|  |
-----
-----
|o o|
|o o|
|o o|
-----

Sorry, you rolled a 7. You crapped out!

Press any key to continue . . .
```

Run 2:

Screen 1

```
*****
*           Craps           *
*****

You rolled: 7
-----
|o o|
| o |
|o o|
-----
-----
|o  |
|   |
|  o|
-----

Congratulations! You rolled a 7. You win!

Press any key to continue . . .
```

Run 3:

Screen 1

```
*****
*           Craps           *
*****

You rolled: 6
-----
|o  |
| o |
|  o|
-----
-----
|o  |
| o |
|  o|
-----

You must roll a 6 to win. If you roll a 7, you lose.

Press any key to continue . . .
```

Screen 2

```
*****
*           Craps           *
*****

You rolled: 8
-----
|o  |
|   |
|  o|
-----
-----
|o o|
|o o|
|o o|
-----

You must roll a 6 to win. If you roll a 7, you lose.

Press any key to continue . . .
```

Screen 3

```
*****
*           Craps           *
*****

You rolled: 9
-----
|o  |
| o |
|  o|
-----
-----
|o o|
|o o|
|o o|
-----

You must roll a 6 to win. If you roll a 7, you lose.

Press any key to continue . . .
```

Screen 4

```
*****
*           Craps           *
*****

You rolled: 10
-----
| o o |
|  o  |
| o o |
-----
-----
| o o |
|  o  |
| o o |
-----

You must roll a 6 to win. If you roll a 7, you lose.
Press any key to continue . . .
```

Screen 5

```
*****
*           Craps           *
*****

You rolled: 6
-----
|   |
|  o |
|   |
-----
-----
| o o |
|  o  |
| o o |
-----

Congratulations! You win!
Press any key to continue . . .
```

To complete this assignment you must submit the following:

1. An electronic copy of your program on Turing

- a. Add a new project named 06_Craps to your previously created assignment solution called PUNetID-Assignments. It is *vital* that you name your project correctly!
- b. Type your program (fully documented/commented) into the project. You must follow the coding standards!
- c. Pay attention to the example output! Your program's output must look **exactly** like the example output! The spacing and newlines in your output must match exactly.
- d. Make sure that your program compiles and runs correctly. If you get any errors, double check that you typed everything correctly.
- e. Make sure that your program does not produce any warnings.
- f. Once you are sure that the program works correctly it is time to submit your program. You do this by logging on to Turing and placing your complete solution folder in the **CS150-02 Drop** folder. This solution folder must contain six projects.
- g. The program must be in the drop folder by 9:15am on the day that it is due. Anything submitted after that will be considered late.

2. A hard copy of your program

- a. The hard copy must be placed on the instructor's desk by 9:15am on the day that it is due.
- b. The hard copy must be printed in color, double-sided, and stapled if necessary.
- c. Your tab size must be set to 2 and you must not go past column 80 in your output.

Good luck! And remember, if you have any problems, come and see straight away. ☺