For this assignment, you will write a complete C++ program that plays the game
*UnderOver*. If you aren’t familiar with the game, here are the basics. *UnderOver* 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. Before rolling the dice, you must
place a bet and decide if you think the roll will be under 7, over 7, or exactly 7. The
payout for a correct bet is:
  - Under 7 pays 1-1
  - Over 7 pays 1-1
  - Exactly 7 pays 4-1

So for instance, if you bet $1 on under and win, then you gain a dollar as well as getting
your dollar back. If you bet on exactly 7 and win, you make $4.

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 *once* as the first executable statement in main.

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

Also add `#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:

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

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

1. `void drawHeading ();` - displays the heading
2. `void drawDieFace (int face);` - draws the die face that
   is passed into the parameter as follows:
   ```cpp
   ______
   |o o|
   |
   | o |
   |
   ______
   3. `int rollDie ();` - rolls one random die and returns
   the roll value
4. void drawDiceFace (int face1, int face2); - calls drawDieFace twice showing a special output as seen on the screen as follows:

```
-----
|   |
| o |
|   |
-----
-----
|   |
| o |
|   |
-----
Dice Roll (1 , 1)
```

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:

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

Sample Run:
Note1: Each user input is to be error checked as shown in the previous examples.

Note2: Allow either uppercase or lowercase input.
To complete this assignment you must submit the following:

1. An electronic copy of your program on Grace
   a. Add a new project named 05_UnderOver to your previously created assignment solution called PUNetIDAAssignments. It is vital that you name your project correctly!
   b. Type your program (fully documented/commented) into the project. We are now commenting each function in a program. 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 Grace and placing your complete solution folder in the CS150-01 Drop folder. This solution folder must contain six projects.
   g. The program must be in the drop folder by 1:00pm 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 1:00pm 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. 😊