Quest 11

200 EXP

CPSC121 SI

Erin: Hello programmer!

Erin: Today we will be going over generating random numbers.

Erin: Generating random numbers can be very useful for many applications. Just look at any program that deals with chance. If you are ever rolling dice, drawing cards, or spinning a wheel, there needs to an element of randomness.

Erin: Random numbers are generated using the ctime library, but why do we need time when it comes to random numbers?

Erin: Random numbers are generated as a sequence of numbers based on some given value. There is a function called srand() which takes in a seed value to determine which sequence of random numbers to generate. If you pass the same seed to srand() then you can expect the same sequence of random numbers.

Erin: Of course, this sequence of numbers is not defined by you, but if the same sequence of random numbers are generated each time, then they are no longer random are they?

Erin: That's where time comes into the equation. Time is always changing, so if you use this as your seed, then the sequence of random numbers will always be changing as well.

Erin: Here's an example of how to generate a random number.

```
#include<ctime>
void main()
{
   unsigned int seed = time(0); //Returns the system time in seconds
   srand(seed);  //sets the seed of rand to the system time
   int randomnum = rand() % 10; //generates a random number from 0 - 9
}
```

Erin: Notice how there is a modulus 10 after the rand()? This takes whatever random number generated and uses the modulus to generate only numbers in a 10 integer range. You can not only determine integer range, but which integers by shifting the numbers around.

```
rand() % 10 + 1; //generates a random number from 1 – 10
rand() % 3 + 5; //generates a random number from 5 – 7
```

Erin: Now you should be able to figure out how to generate a random number and set the range of those numbers.

Erin: Today's quest will be creating a game. There are two topic that you may do, but only one is necessary to pass this quest.

Quest 11a:

Create a game where you play with dice rolls. Dice rolls can return values of 1-6. You must create a game with you as player 1, and player 2 is Rex. Each player rolls 1 die. Whoever rolls the higher value wins a point. If you both roll the same value, then neither player gets a point. First player to reach 5 points wins the game. I would like to see what each player rolls and the current score after each roll.

Quest 11b:

Create a guessing game where a random number in the range of 1 – 100 is generated. The player must guess which number was generated. The player will make a guess, and the program shall tell the player if they guessed correctly, or if the generated number is higher or lower than the current guess. The player gets 6 tries to guess the number. If they do not guess the number by 6^{th} try, they lose the game.