

# **Project1**

## **High Low Game**

**CIS 5 – 44187**  
**Dokyoung Shin**

## **Introduction**

This program is high low game.

The high low game is very simple and enjoyable game to do for everyone.

This is the number guessing game. The Player deposits an initial amount to start playing game. The computer will choose the random number between 1 to 100, and the player will guess the number to win the ten times of their betting money. If the player bet on wrong number they will lose their betting money.

This number guessing game is great for fun and entertainment.

## **Summary**

Project size: about 124 lines

First, I created a simple high low game which is guessing the number between 1 to 100. After that, I added the limited time to try, so only 10 times to figure it out what is number computer picked.

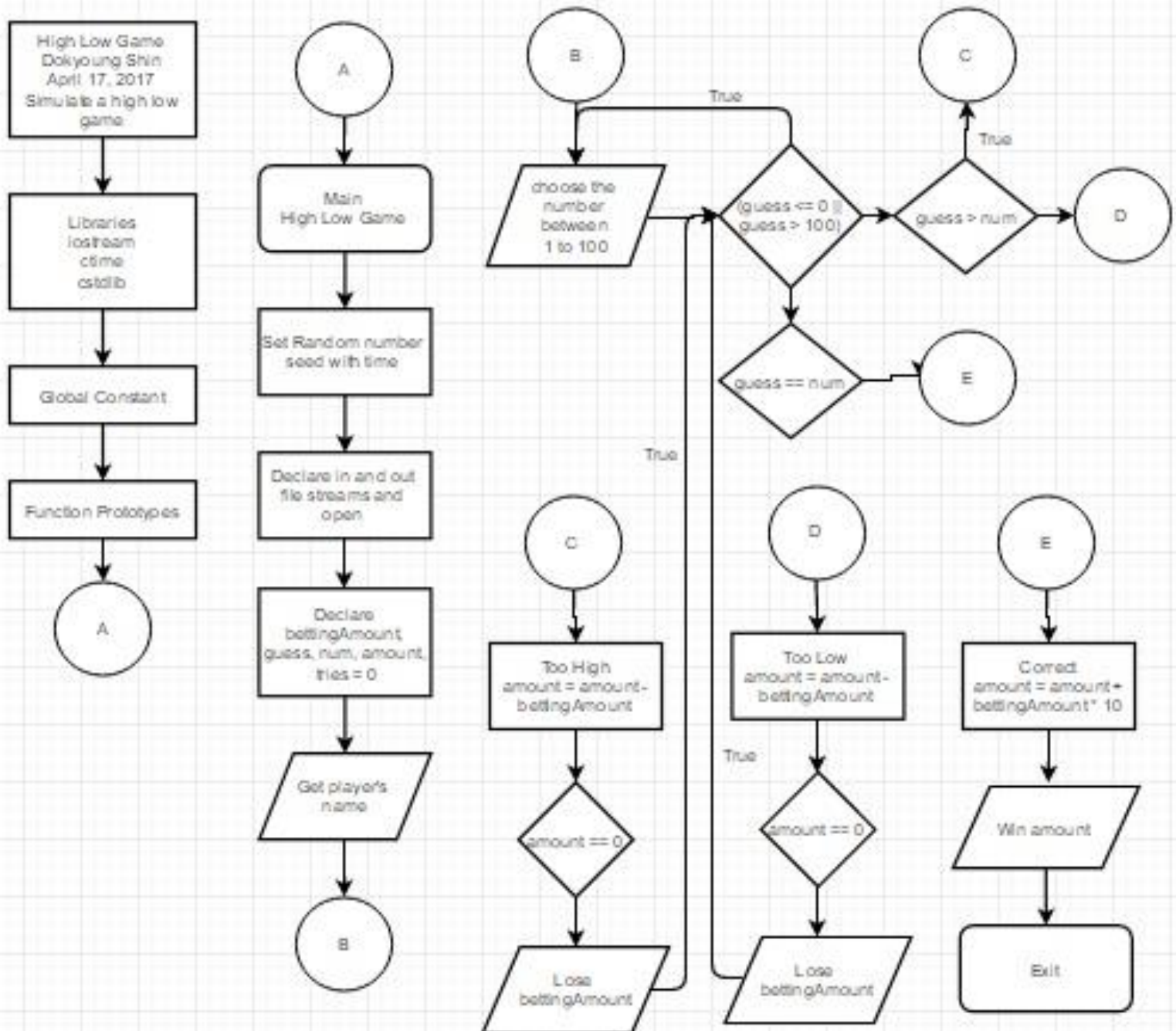
This project includes many concepts that we learned from the chapters in the textbook. Even though this game program is very simple, I think it has many possibilities to be extended for second project.

For the second project, I can combine with deck of card. For example, computer will choose one of card and I should guess not only number, but also color, odd or even, shape.

I realized that C++ has numerous functions and it can make many different ways to create the programs. It was very interesting time to learn and figure it out.

## Flow Chart

### High Low Game Version 5



# **Pseudo Code**

/\*

File: main.cpp

Author: Shin Dokyoung

Created on April 17,2017

Purpose: Simulate a high-low game.

\*/

//system Libraries

//input - Output Library

//random numbers

//string Functioning

//time for rand

//namespace std of system libraries

//user Libraries

//global Constants

//such as PI, Vc, -> Math/Science values

//as well as conversions from system of units to

//another

//function Prototypes

//main

//declare Variables

//string the playername

//int for player's balance amount

//int for player's betting amount

//int for player's number guess

//set tries equal to 0

```
//int for computer generated number
//char for Answer

//input values
//seed the random generator
//get the player's name
//explain betting system
//get player's betting amount
//get player's number
//will hold the randomly generated integer between 1 to 100

//output values
//if guess <= 0 or guess > 100, try again
//if guess > num, Too high
//if guess < num, Too low
//if guess is equal to number, player win

//Exit!!
```

# Cross Reference for Project 1

## Where in Code

Chapter	Section	Topic	Line number
2	2	cout	Used throughout entire project
	3	libraries	iostream, cstdlib, string, ctime
	4	variables/literals	Lines 31-44
	5	Identifiers	Break, string, if, while, char, float, cin, etc....
	6	Integers	Follows my variables
	7	Characters	Lines 38, 62, 122
	8	Strings	Lines 27
	9	Floats No Doubles	
	10	Bools	Lines 71
	11	Sizeof *****	Achieved
	12	Variables 7 characters or less	Keep this consistent throughout project
	13	Scope ***** No Global Variables	Achieved
	14	Arithmetic operators	+ - * / %
	15	Comments 20%+	Used throughout, better displayed in pseudo code
	16	Named Constants	
	17	Programming Style ***** Emulate	Achieved
3	1	cin	Lines 41, 57, 70, 117
	2	Math Expression	Lines 66, 83, 95, 108
	3	Mixing data types *****	Demonstrated through ratio usage
	4	Overflow/Underflow *****	
	5	Type Casting	Line 337, 8, 40, 46 – 52, 56, 60, 69, 73, 81, 87, 93, 99
	6	Multiple assignment *****	Line 69
	7	Formatting output	Line 113
	8	Strings	Lines 27

	9	Math Library	
	1 0	Hand tracing *****	Achieved
4	1	Relational Operators	Lines 55, 59, 62, 63, 68, 72, 75, 80, 86, 89, 90, 92, 98, 101, 103, 104, 109, 118
	2	if	Lines 58, 71, 79, 85, 97
	4	If-else	Lines 91
	5	Nesting	Most of the project is different nested statements. Primarily demonstrated in the menu function.
	6	If-else-if	Lines 85, 97
	7	Flags *****	
	8	Logical operators	
	1 1	Validating user input	Lines 41, 57, 70
	1 2	Conditional Operator	Lines 41, 57, 70, 117
	1 4	Switch	
5	1	Increment/Decrement	Lines 78
	2	While	
	5	Do-while	Lines 42-118, 54-63, 67-110
	6	For loop	Lines 140-176
	1 1	Files input/output both	
	1 2	No breaks in loops *****	
***** Not r	require	show	

# **Program**

/\*

File: main.cpp

Author: Shin Dokyoung

Created on April 17, 2017, 8:15 PM

Purpose: Simulate a high-low game.

\*/

//System Libraries

#include <iostream> // Needed to use Input - Output Library

#include <string> // Needed to use strings

#include <cstdlib> // Needed to use random numbers

#include <ctime> // Needed to use time for rand

using namespace std;

//User Libraries

//Global Constants

//Such as PI, Vc, -> Math/Science values

//as well as conversions from system of units to

//another

//Function Prototypes

//Executable code begins here!!!

int main(int argc, char\*\* argv) {

    //Declare Variables

    string playerName;



```

int amount; // player's balance amount

int bettingAmount; // player's betting amount

int guess; // player's number guess

int tries = 0;

int num; // computer generated number

char Answer;


//Input values

srand(time(0)); // Seed the random generator

cout << "\n\nWELCOME TO THE HIGH-LOW GAME\n\n";

cout << "Enter your name : ";

getline(cin, playerName);

cout << "\nEnter deposit amount to play game: $";

cin >> amount;

do

{


//Process by mapping inputs to outputs

    cout << "\n\t\tRULES OF THE GAME\n";

    cout << "\t1. Guess any number between 1 to 100.\n";

    cout << "\t2. If you win, you will get 10 times of money you bet.\n";

    cout << "\t3. If you bet on wrong number, you will lose your betting"

        " amount.\n";

    cout << "\tGood Luck! Let's start the game.\n\n";

    cout << "\n\nYour current balance is $" << amount << ".\n";

    // Get player's betting amount

    do

    {

```

```
cout << playerName << ", Enter the money to bet: $";  
cin >> bettingAmount;  
if (bettingAmount > amount)  
{  
    cout << "Your betting amount is more than your current balance."  
        "\n" << "Try again.\n";  
}  
}while (bettingAmount > amount);
```

//Get player's numbers

```
num = rand()% 100 + 1; //will hold the randomly generated integer between 1 to 100  
do  
{  
    cout << "Enter a guess between 1 and 100: ";  
    cin >> guess;  
    if (guess <= 0 || guess > 100)  
    {  
        cout << "Please check the number!! It should be between 1 to 100."  
            ".\n" << "Try again.\n";  
    }  
}
```

//Output values

```
tries++;  
if (guess > num)  
{  
    cout << "Too high! Bad Luck this time!!\n\n";  
    cout << "You lost $" << bettingAmount << ".\n";  
    amount = amount - bettingAmount;
```

```

    cout << "Your balance amount is $" << amount << ".\n\n";
    if (amount == 0)
    {
        cout << "Sorry. You have no money to play.";
        break;
    }
}
else if (guess < num)
{
    cout << "Too low! Bad Luck this time!!\n\n";
    cout << "Your lost $" << bettingAmount << ".\n";
    amount = amount - bettingAmount;
    cout << "Your balance amount is $" << amount << ".\n\n";
    if (amount == 0)
    {
        cout << "Sorry. You have no money to play.";
        break;
    }
}
else
{
    cout << "\nCorrect! You got it in " << tries << " guesses!\n";
    cout << "Congratulation! You won $" << bettingAmount * 10 <<
        ".\n\n";
    amount = amount + bettingAmount * 10;
}
}while (guess != num);

```

```
//Random number between 1 and 100

cout << "\n\nThe winning number was " << num << ".\n";

cout << "\n" << playerName << ", You have $" << amount << ".\n";


cout << "\n\nDo you want to play again? [Y/N]";

cin >> Answer;

}while(Answer == 'Y' || Answer == 'y');


cout << "\n\nThanks for playing game. Your balance amount is $" << amount <<
    ".\n\n";

//Exit stage right!

return 0;

}
```