CPP 程式設計題

命題者：SCY

題目名稱(中文/英文)：GuessNumber

主要測試觀念：

Basics		Functions	
■	C++ BASICS 1	□	SEPARATE COMPILATION AND NAMESPACES
□	FLOW OF CONTROL	□	STREAMS AND FILE I/O
■	FUNCTION BASICS	□	RECURSION
□	PARAMETERS AND OVERLOADING	■	INHERITANCE
□	ARRAYS	□	POLYMORPHISM AND VIRTUAL FUNCTIONS
■	STRUCTURES AND CLASSES	□	TEMPLATES
□	CONSTRUCTORS AND OTHER TOOLS	□	LINKED DATA STRUCTURES
□	OPERATOR OVERLOADING, FRIENDS, AND REFERENCES	□	EXCEPTION HANDLING
□	STRINGS	□	STANDARD TEMPLATE LIBRARY
□	POINTERS AND DYNAMIC ARRAYS	□	PATTERNS AND UML

題目說明：

Listed below is code to play a guessing game in which two players attempt to guess a number. Your task is to extend the program with objects that represent either a human player or a computer player.

// you can modify the following codes to let your "guess" more smarter

```
bool checkForWin(int guess, int answer)
{
   if (answer == guess)
   {
        cout << "You're right! You win!" << endl;
        return true;
   }
   else if (answer < guess)
        cout << "Your guess is too high." << endl;
   else
        cout << "Your guess is too low." << endl;
   return false;
}

void play(Player &player1, Player &player2)
{
   int answer = 0, guess = 0;
   answer = rand() % 100;
   bool win = false;


   while (!win)
   {
     cout << "Player 1's turn to guess." << endl;
     guess = player1.getGuess();
     win = checkForWin(guess, answer);
     if (win) return;

     cout << "Player 2's turn to guess." << endl;
```

```
    guess = player2.getGuess();
    win = checkForWin(guess, answer);
   }
}
```

The play function takes as input two Player objects.
Define the Player class with a virtual function named getGuess(). The implementation of Player::getGuess() can simply return 0.
Next, define a class named HumanPlayer derived from Player. The implementation of HumanPlayer::getGuess() should prompt the user to enter a number and return the value entered from the keyboard.
Next, define a class named ComputerPlayer derived from Player. The implementation of ComputerPlayer::getGuess() should be in ***smarter*** **way (i.e., not randomly select a number from 0 to 100).**
Finally, construct a main function that invokes play(Player &player1, Player &player2)with two instances of a HumanPlayer (human vs. human), an instance of a HumanPlayer and ComputerPlayer (human vs. computer), and two instances of ComputerPlayer (computer vs. computer).

Note that please use this following code snippets as your main()

```
// Main
int main()
{
	HumanPlayer playerH1, playerH2;
	ComputerPlayer playerC1, playerC2;

	play(playerH1, playerH2);
	play(playerH1, playerC1);
	play(playerC1, playerC2);

	return 0;
}
```

輸入說明：當你是 HumanPlayer 時，輸入猜測的數字，當是 ComputerPlayer 時則要由電腦自動輸入猜測的數字 in *smarter* way (i.e., not randomly select a number from 0 to 100.)

輸出說明：
太小輸出 Your guess is too low.
太大輸出 Your guess is too high.
猜中數字輸出 You’ re right! You win!

IO範例 ：

```
Player 1's turn to guess.
50
Your guess is too high.
0~50
Player 2's turn to guess.
25
Your guess is too high.
0~25
Player 1's turn to guess.
12
Your guess is too low.
12~25
Player 2's turn to guess.
18
Your guess is too low.
18~25
Player 1's turn to guess.
23
Your guess is too high.
18~23
Player 2's turn to guess.
100
you guess out of range.
20
Your guess is too high.
18~20
Player 1's turn to guess.
19
You're right! You win!
Player 1's turn to guess.
50
Your guess is too low.
50~100
Player 2's turn to guess.
computer guess 57
Your guess is too high.
50~57
Player 1's turn to guess.
53
Your guess is too high.
50~53
Player 2's turn to guess.
computer guess 51
You're right! You win!
Player 1's turn to guess.
computer guess 3
Your guess is too low.
3~100
Player 2's turn to guess.
computer guess 56
Your guess is too high.
3~56
Player 1's turn to guess.
computer guess 46
Your guess is too high.
3~46
Player 2's turn to guess.
computer guess 45
Your guess is too high.
3~45
Player 1's turn to guess.
computer guess 6
Your guess is too low.
6~45
Player 2's turn to guess.
computer guess 42
Your guess is too high.
6~42
Player 1's turn to guess.
computer guess 24
You're right! You win!
```

附屬資料：

☑解答程式：GuessNumber.cpp(檔名)

☑測試資料：

☐易，僅需用到基礎程式設計語法與結構

☐中，需用到多項程式設計語法與結構

■難，需用到多項程式結構或較為複雜之資料型態或結構

解題時間：30 分鐘。

其他註記：

main 使用 input_main

除了輸出要求以外的 output 為方便 debug 的輸出，可有可無