COEN 70: Formal Specification and Advanced Data Structures

Winter 2015

Lab 7: Polymorphism – Guessing game

We are going to implement the guessing game. This game should support two players and each player can be human or the computer. The game starts by generating a random answer and the players take turn in guessing the answer. The player with the correct guess wins the game. Upon winning, the game displays the number of guesses the winning player took to win the game.

For the computer player, instead of randomly guess the answer, we are going to program it to play the same way we humans do. In order to do this, we are going to provide the computer player the ability to remember the opponent's guess and the comparison result of that guess with the correct answer. We are going to use polymorphism, therefore the base class has to following interface:

```
class Player {
      public:
            Player();
            virtual ~Player();
            virtual int getGuess();
            // Returns the players next guess, an integer between
            // 0 and 99 (inclusive)
            virtual void lastWasTooHigh(bool tooHigh);
            // Called to inform the player that their last guess
            // was too high (if the tooHigh argument is true) or
            // too low (if the tooHigh argument is false)
            virtual void opponentsGuess(int guess, bool tooHigh);
            // Called to tell the player what the opponent's last
            // guess was, and whether it was too high or too low.
            virtual void reset();
            // Called to reset the player at the end of the game.
            // Any stored state about the last guess should be cleared.
};
```

Implement the `HumanPlayer` and `ComputerPlayer` classes that derive from the above `Player` class. Additionally, implement the game engine `play()` and properly test the program.