Carlos R. Ocasio Rodriguez

CSCI 301-61

**Test One Project**

1. Goal: Create a guessing game where the user enters numbers until they guess the right number.
2. Considerations:
   1. Print a welcoming message for the user.
   2. Ask the user to type in a numerical guess for the magic number.
   3. Tell the user if the guess is too high, too low, or just right.
   4. Give the user the option to exit the game or to try again if he/she wishes.
   5. Quit with a closing message if the user guesses the magic number or types the quitting signal.
3. Strategy:
   1. Print a greeting to the user.
   2. Prompt the user to enter a value as the max number to be guessed. This will determine the range of numbers to be guessed
   3. Generate a random number using the input set as the max value by the user.
   4. Prompt the user to guess the “magical” number.
   5. Determine whether the user is correct or not.
   6. If the user is correct display a winner message, followed by a goodbye (quitting) message and then exit the program. If the guess is incorrect prompt the user whether the guess was too high or too low and ask the user if they want to try again or quit the game.
   7. If the user wishes to keep playing loop back to step e, until the player wins or gives up.
4. Test Results:

|  |  |  |  |
| --- | --- | --- | --- |
| Test | Expected Result | Actual Result | Comments |
| 1. Run the program. | Program runs without errors. | Program is running without errors. | Program is running as expected. |
| 1. Generate a random number based on the players entered max number. | Program will generate a random number between 1 and the players entered number. | The program is correctly generating random numbers between 1 and the players enter max number. | The program is generating the proper random number that will be used as a guess. |
| 1. Determine whether the player’s guess is too high or too low. | A message will notify the user when the guess was too high.  A message will notify the user when the guess was too low. | The program outputs the appropriate message based on the player’s guess. | This test will allow the player to guess the number much faster. |
| 1. Allow user to try again if the guess was incorrect. | The player should be prompted to try to guess the game for as long as he/she wishes to keep playing. | The game allows players to keep trying for as long as they wish.  To exit the game the player needs to enter 0 when prompted. | The game ends when the player guesses the number correctly or when the user gives up.  The user can attempt to guess the number indefinitely. |
| 1. Display winner message. | A message congratulating the player will appear after the player guesses the number correctly. | The program determines when the player guessed the number correctly and displays the winner message as expected. | The program displays the winner message as expected. |
| 1. Exit the program only when the player wins, or the player give up. | The program should only close when the player wins, or the player gives up. | The program continues to run until the player wins, or the player gives up. | The program closes itself as expected. |