Project 2: Hangman Game

Collaboration Policy: This assignment **must** be completed **individually**. You may discuss high-level ideas with other students, **but you should not share code**.

Description

The goal of this homework assignment is to create hangman-like game with one player playing against a computer.

First the player picks the length of the word she or he would like to use. Then the computer picks a random word of the appropriate length from a list (use the list provided in the starter file - do not make your own). If the computer does not have a word of the specified length in the list, it will ask the user to pick the length again, until a word of the correct length is found.

This time, instead of counting guesses, the player begins with a number of *lives* (or number of tries they can use before the game ends) that is equal to the **length of the word + 4**. The player then guesses a letter that she or he thinks appear in the secret word. The computer checks whether the letter appears in the secret word. If the letter exists, the computer reveals *all* of the instances of that letter within the secret word. If the letter does not exist, the computer decreases the player's lives by one. If the player tried to guess a letter that was guessed before, the computer will announce this **and take a life away**.

The computer always displays the progress that the player has made towards guessing the word. The game continues until either all of the letters have been guessed (and the player wins) or all of the player's lives are used up (and the player loses). In either case, the computer should announce this event and the game should end.

Here is an example of the game (remember this is just an example):

Computer: How long of a word do you want to play with?
Player: 20

Computer: Sorry I don’t know any words of length 20.
Computer: How long of a word do you want to play with?
Player: 8

Computer: Guess a letter.
Computer: _ _ _ _ _ _ _ _
Player: r

Computer: Yes, r is in the secret word.
Computer: Guess a letter.
Computer: _ _ r _ _ _ _ _

Player: r
Computer: You have already guessed the letter r.
Computer: Guess a letter.
Computer: _ _ r _ _ _ _ _

Player: a
Computer: Yes, a is in the secret word.
Computer: Guess a letter.
Computer: _ _ r _ _ a _ _

Player: L
Computer: Yes, l is in the secret word.
Computer: Guess a letter.
Computer: _ _ r _ _ a l l

Player: h
Computer: No, h is NOT in the secret word.
Computer: Guess a letter.
Computer: _ _ r _ _ a l l

Player: f
Computer: Yes, f is in the secret word.
Computer: Guess a letter.
Computer: f _ r _ _ a l l

Player: e
Computer: Yes, e is in the secret word.
Computer: Guess a letter.
Computer: f _ r e _ a l l

Player: i
Computer: Yes, i is in the secret word.
Computer: Guess a letter.
Computer: f i r e _ a l l

Player: b
Computer: Yes, b is in the secret word.
Computer: Congratulations! You figured out the secret word fireball in 8 guesses and had 10 lives left!

Starting Materials

Load this starter file which includes a word bank of 1000 words.

Bonus

If you want credit for the bonus, you must add a note to the dropbox indicating that you completed it.

Graphics

Add the ability to display the letters on the screen using sprites. Try to make it so that the word shown is always centered, regardless of the length. Account for word lengths up to and including 11 letters.

Submission Guidelines

The following must be placed in the D2L dropbox. Add a comment to the dropbox if you completed the extra credit.

Project file

This is the file with all your Snap! code. Name this file as Proj2 with your first and last name, such as: Proj2_MaryMosman.xml

Readme file

The readme file is a separate file that can be shared along side the code so to give context and extra information about the project. Write a file that has 2 sections:

- The first section - Summary, should describe what the program is and does. In this case, it should describe the game (in your own words, not mine), and how to play it.

- The second section - Learning, should describe what you found challenging (or not) about the assignment and how you worked through the challenges and solved the problem. Perhaps you tried something and it didn't work, then you tried a different approach. Perhaps you didn't know where to start and talked it through with a friend. Perhaps the problem was easy because you've done something similar before. The only "wrong" answer here is not talk about how you solved the problem.

Tips

- Remember the saving and testing tips from Project 1! Save often and back it up in multiple locations.
- Start by putting thoughts down on paper. Break the problem into smaller parts before you start writing the code.
- Code a little bit at a time (block by block) and test as you go.
- Be sure to handle the situation when the computer does not know any of the words that are of the length chosen by the user.
- Don't forget to deduct a life when the user guesses a letter she has already guessed.
- Use good naming for variables and blocks. It should be obvious what things are used for based on what they are called. Avoid names like x which give no indication of meaning, and make sure names are not misleading.
- Make sure to read through your code when you are all done. Make sure it makes sense. Look for places where there is a group of code doing a task that could be pulled out into its own block.
- Have someone else (a friend or family member) play your game and give you feedback.