Test Specification
Gravilex, or The Captain's Mistress
Keith Walsh
CS333 Fall 2009

1. Module Overview

The test module consists of the program and two plain text files. The first text file will contain a dictionary of available words; the seconf text file will contain data that allows the game to begin a round from various points in the game.

The text files will test that the round goes as expected.

1.1 Inputs:

File 1 (testNN.words): a list of words, each separated by a space or end of line, that can be used a scorable words in the game.

File 2 (testNN.data): 7 lines that represent the current state of the game, one line with the letters available to be played by the player, one line with the letters available to the computer, and one line with the current score (player space computer).

File 2 (testNN.in): Contains the input for human and computer player moves to automate cin processes

1.2 Outputs:

Each round consists of the following:

1. The current grid is printed.
2. The letters remaining to the human player are printed.
3. The human player is prompted to choose a letter and a column.
4. The updated grid, reflecting the human's choice, is printed.
5. The human is asked to list words that he/she wishes to claim points for. The points awarded for that turn are printed.
6. The letters remaining to the computer player are printed.
7. The computer announces the letter and column it has chosen.
8. The updated grid, reflecting the computer's choice, is printed.
9. The computer lists the words that it claims points for. The points awarded for that turn are printed.
10. The total scores for both players are printed.
11. ~~If the grid is now completely filled, print a line with "Game over." followed by a line with one of "I win!", "You win!", or "We have tied!", as appropriate.~~ Not in spec, so out of scope

2. Test Data

2.1 Representative Input

1. The example from the assignment.

2.2 Functional Coverage

1. letter dropped into empty grid
    a. test1
2. computers first play follows logic (is first consonant, adjacent, left)
    a. test1
3. letter dropped on top of another letter
    a. test 1 (showing human player dropping letter on top of first play)
4. word extended as a power word
    a. test1 (showing computer claiming add on to you)
5. word running from left to right
    a. test1 (showing computer claiming left-right word)
6. word running from down to up
    a. test1 (showing human claiming down-up word)
7. word running from right to left
    a. test2 (human makes ride, claims as not in dictionary)
8. word running from up to down
    a. test2 (human makes ace)
9. word running diagonally, southeast
    a. test3 (human claims pickle)
10. word running diagonally, northeast
    a. test3 (human claims truck)
11. word running diagonally, northwest
    a. test3 (computer claims shuck)
12. word running diagonally, southwest
    a. test4 (computer claims door)
13. last space in grid filled, game terminates
    a. test3 (final move by computer)

2.3 Boundary Values

1. word played found in play, not in word file
    a. test2 (player makes ride)
2. word not found on grid or play
    a. test2 (player claims men)
3. word found on grid, not in play
    a. test2 (player claims door)
4. player tries letter not in list

    a. test2 (player tries z)
5. player tries to drop on a full column
    a. test3 (human tries to drop in col 3)

2.4 Special Cases:

1. Word claimed wraps around grid
    a. test4 (human claims lead)