SCRABBLE

with STUFF

A JAVA GAME BY HIZAL CELIK FOR 15-214

Reflection and changes on HW4a:

I argue against the comment made about the Blank tile in the domain model: I don't mean the Blank tile as an empty space, I mean it as the wildcard tile used in actual Scrabble games. Another comment is that, for the system diagram, I had included the word "optional" signifying that the player could indeed directly end their turn upon its' start, without doing anything. The TA mentioned that I had not drawn that, but it is there. These files are now in HW4a-b folder in HW/4

Object design-wise, I have implemented a Move object to store the data on the current move including scores, special tile effects to invoke later (if score related) and letters in the move.

I also added a dictionary class, and have greatly, greatly increased the runtime of the getMoves() operation because of it. I changed the words.txt file to one of Scrabble's official dictionary, which has approx.. 200,000 more words than the given text file. As a result, word searches became very slow. So when the dictionary class is initiated, it saves the dictionary to an arraylist of arraylists, each storing every word that shares the first two letters (lists for words that begin with "aa", "ab", "ac"… "zz"). Then every time the isWord() method is called, it skips directly to the correct part of the dictionary that shares the first two letters. The dictionary class also has separate lists that save the history of searched words, and which ones were legal, this way reducing the time even more for repeated searches. As a result, when typing letters into the GUI, the searches are performed near-instantaneously and is able to keep up with the player's typing speed. These changes were like the golden step of this whole thing because the experience was quite anticlimactic for having to wait a few seconds

every time a key was pressed. I wouldn't want to make the game search on an enter press, that's not as cool looking.

A "Word" is still an ArrayList<LetterTile> and there is no "Word" object but there is no need as I use "Move" as a "Word" as well, just ignoring score info. I consider a word as a move, that's why. getMoves() returns a list of possible Moves.

My special tiles all share the exact same interface so there is plenty of reuse amongst them, and their activate() method is kept to an efficient and minimal code that works effectively. And without much variable dependency.

Implementing the FlipTile special was simple enough, just adding it to the list in the initSpecials() function in Game, and changing the activate(), name and description parameters. One issue was I saved a reversed copy of lettertiles and tried to sync the coordinates with those in the current Move, but I forgot that both lists were referring to the same LetterTile objects, thus changing them together. I fixed that by changing one to an arraylist of locations instead.

SpecialTile.activate() returns a string, which is used in the GUI as a notification that a special tile was triggered, what happened, and who placed it. This was the best way I could think of, of notifiying the GUI (or textUI).

Throughout my coding these last few weeks I have kept finding situational bugs in my getMoves() algorithm, particularily surrounding the blank tiles or their existence even if they are not part of the move. I have squished every bug I have seen so far and hopefully none others have existed. Of the game I played until completion with a friend, I fixed all the bugs we came across. Also I won, with a score of 315. Not bad.

I really, really recommend playing the combination of "Quick Game" and "Casual" mode, especially with a friend or two. It is extremely fun, especially for you TAs probably after dryly testing all the other scrabble games. It's fun to use and type with, seeing all those options float around as you type or move the mouse. And whats more fun is when you can't think of anything, start typing randomly and discover new words that you swear aren't really words until you check the dictionary again ("Palla"?). It's a fun thing to laugh at and connect with friends. Aside from that, the Casual mode really isn't much of a challenge, but is a welcome difference to the blind-and-challenging gameplay of regular scrabble ("Hard" mode).