Jennifer Be
INFO W18 Summer 2016
Project 1: Minesweeper

INTRODUCTION:
First of all, creating this project took a lot of patience, planning, and algorithmic thinking. In the beginning, my fears were that I would not have enough lines to fill the requirements, but surely I was able to because Minesweeper is more complex than meets the eye.

PLANNING:
Probably, I spent too much time trying to piece together what classes were necessary, without really mapping out all of the functions that would be needed. I didn't think much about classes, but thankfully after the revisal I had a more clear direction to create this.

ALGORITHM:
Figuring out how the boxes would open was probably the hardest part. I had to figure out how to calculate what boxes would open (because if you click on a box that has no mines next to it, the boxes around it also open up, and if another box if clear, it's neighbors open, and so forth).

My first idea was to figure out all of the sets of blanks and the corresponding boxes that needed to be opened, and create unique sets. I would have a dictionary that would store: key (the position), and value (being a set of all the positions that needed to be opened). That turned out to be a mess because it was hard for me to figure out a way to morph all of the sets, and I was looking at the grid linearly. I would have to go through the grid numerous times to get complete sets.

I ended up choosing to implement an algorithm that checks all of the neighbors, and if the neighbor is blank, check their neighbors, and continue. This method worked really well because I utilized the **set** data structure. I initialized a list that acted like a queue –I would add neighbors to the list, and pop them (and do all of the appropriate checking, and adding their neighbors to queues). If the items have been "seen" already—if they were in the set—I would continue popping the list. This went on until the list was empty.

One way I tried to make my program faster is by setting up the Square class, which saved information about the squares so that the game would be faster.

CONCLUSION:
All in all, this was a very fun and challenging assignment! I would like to clean up the game to have more functions, and maybe implement a minesweeper solver (using Artificial Intelligence, which I took a class on).