Pacman Mini-Assignment

Due Date: Bring a hard copy to section

Instructions

- Please read the handout extra carefully and play around with the Pacman demo before you delve into these questions. We recommend completing this mini-assignment before you begin coding.
- Carefully think through the discussion questions and bring in your printed thoughts/responses to them. Be prepared to explain and discuss the tradeoffs of your design choices during discussion session.
- As the name suggests, a “Final Project” is a lot more involved and bigger in size and complexity than your previous projects, so please give these questions a lot of careful thought. Trust us, that will save you time while coding.
- A reminder that the diagrams can be drawn in the digital program of your choice or by hand.

Written Questions:

1. What data structure(s) will you use in this assignment? What type of objects will these data structures hold? How will you use them? Be specific.

2. Create the following diagrams. Diagrams can be drawn in the digital program of your choice or by hand.
   a. A complete inheritance diagram. Include any interfaces you will write or use.
   b. A logical containment diagram for your program. All classes used in your proposed design should be modeled in this diagram, including JavaFX classes. Think about the purpose of each class as you design your program and be prepared to discuss it.

3. Provide pseudocode for the ghost’s breadth-first-search target-finding algorithm, and explain in what the pseudocode is doing in two to three sentences. (This is a more complicated algorithm then we have seen before with some helper methods, so this will likely be a substantial amount of pseudocode. Your pseudocode should translate almost line by line to Java, so be sure to include all your variable initializations and correct types. Write your helper methods in your pseudocode. Note: Do not just regurgitate the pseudocode we give you in the handout! Think critically and write detailed pseudocode).
Make sure you have a good understanding of the major components of your project!

Questions to Think About:

You do not need to type up your answers to these questions, but be sure to give them some thought and be prepared to discuss them in your design discussion.

1. How do ghosts know when to leave the pen? How do they leave in the correct order? What data structure could we use to accomplish this?

2. What is the purpose of a “Collidable” interface? What objects would implement this interface? What tangible ~polymorphic~ benefits does this allow us to have when storing objects?

3. Write up a plan for coding Pacman incrementally. Pacman is an even larger project than Tetris, and should be coded in small, manageable steps. See earlier handouts for examples of a plan for coding incrementally -- we highly recommend actually writing this out before you begin coding!