

Assignment 2: An Amazing Maze (20% of course total) Marking Guide

Can be done either individually or in a pair (recommended).

Total = [40] marks. Use the formula $\text{Quiz_completed?}(\text{Quiz_mark} + \text{Program_mark}) : (0.5 * \text{Program_mark})$.

Quiz_mark

[5] Quiz: Online quiz.

Program_mark has three parts [6]+[24]+[5]

Only the first [6] will be considered on Program_mark if cannot compile (communicate this to students).

[6] Design

- [2] CRC cards: Reasonable break-out of classes, high-level responsibilities, and collaborators.
 - Progressive -0.5 for things like unreadable writings, low-level responsibilities, classes that do nothing (e.g., no collaborators nor collaborated by anyone).
- [3] UML class diagram: Clear OOD.
 - Progressive -0.5 for things like a class responsible for more than one thing or knowing everything, incorrect encapsulation with members being non-private, not meaningful or missing class relationships (and multiplicity when appropriate), too many or too few member fields/functions listed.
- [1] OOD explained: Good description of how two non-trivial tasks are completed (1 each).

[24]

- [9] Able to generate maze correctly. Comprised of the following:
 - [2] No internal 2 x 2 square of cells
 - [2] No internal 2 x 2 square of walls
 - [2] No internal corner with a wall
 - [2] Has path to every cell
 - [1] Has multiple paths to some of the cells
- [5] Correct user input handling. Comprised of the following:
 - [1] WASD works
 - [3] Special keys work
 - [1] Handles invalid inputs
- [10] Correct game behaviour. Comprised of the following:
 - [2] Correct revealing of the cells as the hero moves
 - [2] Correct movements of the hero/monsters (e.g., no going through walls)
 - [2] Correct display of the power (e.g., 1 each time at a randomized location)
 - [2] Correct handling of winning condition, including cheat mode
 - [2] Correct handling of losing condition

[5] Code Quality and Style Guide

- Reasonable object-oriented structure
 - must separate packages for UI and game logic.

- Very minor violations of style guide have no penalty
 - (e.g., having "int myCount=0;" (spacing wrong) once)
- Lose a few marks for consistent problems
 - (like always getting the spacing wrong).
- Larger penalties possible for horrific code (0 on Program_mark)

Some specifics to check with suggested deductions (to Program_mark)

- -3 Poor/missing JavaDoc comment on a class (not needed on methods and fields)
- -5 Incorrect indentation, brackets, or spacing (use IDE's reformat if needed).
- -5 Poor intention revealing class, method/variable names.
 - (e.g., making everything public or static, undecipherable names)
- -3 Bad display of information

Forward to instructor if...

- Material is suspiciously similar to another submission or code posted online.