Brendan, Danyaal, Peter, Stephen

Description of the game:

The user/player will control a character in the arcade-style 2D game via keyboard input. A tick elapses with every keyboard input. An enemy will move towards the direction that would make them the closest to the main character every tick and can only move up, down, left, or right. Enemies cannot trigger reward or punishment tiles. No moving entities can go through walls or barriers.

The player has a primary goal and two optional secondary goals. The primary goal of the player is to reach the end tile of the maze after interacting with all checkpoints (regular rewards), though the player may choose to also collect bonus rewards to maximize their score before they end the game by leaving through the exit, or opt to finish the game as quickly as possible to minimize the time taken to finish the game. When the main character interacts with a punishment tile, they lose score.

The player can lose the game in multiple ways: stepping on enough punishment tiles to make their score negative or by being on the same tile as an enemy. Since the player loses if their score becomes negative, the player will start with a preset positive score.

The main character initially is set inside a wall tile. We designate this tile as the entry tile and when the main character steps off the entry tile into the maze, the entry tile will close, preventing the user from escaping back through the entry tile. The entry tile and exit tile will be generated such that they cannot be the same tile, and can only be generated from a wall tile. Only when all checkpoints have been interacted with, will the main character be permitted to exit the maze through the exit tile.

Plan for upcoming phases:

Coming out of Phase 1, we wish to use the evolutionary model of programming while adopting certain agile methods to ensure that our progress is tracked and we are working according to schedule; in each phase, we may be slated to develop new features, so to be able to incrementally develop and deliver a working prototype in each phase is a valuable asset. The use of Jira Software will facilitate communication among group members and use of a Kanban board will be effective to provide insight on everyone's progress.. In this way, we are using a hybrid development style, combining the tracking process of Kanban with the development life cycle of the evolutionary model.

We will use a Java GUI (possibly Java Swing) to draw the board, represented as a two-dimensional array. The game objects are either tile and tile modifiers (stationary) or the two minimum non-stationary entities: the player and a moving enemy. There must be checks in place to ensure that non-stationary entities cannot move into wall or barrier tiles, as well as collision detection to check whether non-stationary entities are within the same cell. The enemy's movement will be determined by a function of the player's keyboard input and current main character's position such that the enemy is always moving towards the actor.