

CMPT 276 - Assignment 3: Code Review Report

Michael Plunkett, mplunket@sfu.ca, 301396899

Salman Rafiei, salman_rafiei@sfu.ca, 301435217

April 1, 2023, 9:06pm

Refactors:

Game Panel:

GamePanel::changeGameStates has repeated code

- Package repeated code into new method: GamePanel::setupLevel
- changeGameStates calls setupLevel, passing through the level number
- Also automated the setting of keysNeeded
 - Used to be hardcoded for each level
 - Added numKeys variable to ItemFactory which is calculated when items are spawned
 - Now, in setupLevel, keysNeeded is set to numKeys (exactly the total amount of keys in the current level)
- Commit address: 905fe63e1938263842042cdc2149117f7170e279

Skeleton class

Instance variables have no access modifiers (all defaulted to public):

- Made all variables (ArrayList<String> path, int pathIndex, int nextX, and int nextY) private
- No getters or setters required as they are only used by the Skeleton class
- Commit address: 6d49da589c956e467edf1d58f8361528a9d6d7aa

Bat class

Bats change direction after collision with walls or other entities

- Previously, bats attempted to change directions every 50 ticks
- Added condition to Bat::setAction to check if its collision is on. If so, it will attempt to change directions immediately instead of waiting for the next 50 ticks to pass.
- Long delay between collision and changing direction removed
- Commit address: 808929d480bcd2790cd8ed9661fd9a70cbc21653

Items:

Item instance variables are all public:

- Change all variables to private
 - BufferedImage image
 - String name
 - boolean collision
 - int worldX, worldY,
 - Rectangle hitBox
 - public int hitBoxDefaultX, hitBoxDefaultY,
- Create getter and setter methods for each variable in the Item class

- Change name of the static method `Key::getImage` -> `getSprite` to allow for the creation of `Item::getImage` method as the getter method for image
- This naming convention is also more intuitive and descriptive, as it follows suit with the `setupSprite` methods
- Change all accesses and modifications of these instance variables to call the respective getters and setters
 - Changes made to `Door`, `Key`, `Potion`, `Spikes`, `CollisionChecker`, and `Player` classes
- Commit address: 1e1672b4b7bcff88e632983255802555ff930eff

Generalized the static `getSprite` method

- Originally only existed for `Key` and `Player` classes
 - Used to draw the player sprite on the title screen, and the key sprite on the in game key count stat
- Created `getSprite` methods for each subclass of `Item`, so each item sprite is readily accessible if needed
- (currently only needed for `Player` class and `Key` class, but this makes it easier to build on the game if we wanted more interesting UI)
- Commit address: 1e1672b4b7bcff88e632983255802555ff930eff

BonusReward instance variables are public

- Change `birthTime` and `lifeTime` to private
- Create getters for each and replace accesses with calls to the getter function in the `GamePanel` class
- Create protected setters (only subclasses should be able to set this variable) and implement them in the constructor of `Potion`
- Commit address: a487dd3c4ddfe81612a46cb2a8b940cc424cef11

KeyInputs:

KeyInputs instance variables are public

- Change `upPressed`, `downPressed`, `leftPressed`, `rightPressed`, and `checkDrawTime` to private
- Create getters for each and replace accesses with calls to the getter function in `Player` class
- Commit address: 9351299c764f8441f5424a06c9c4c49680b57298

Fixing Potions

Potions occasionally throws `ConcurrentModification` exceptions

- Instead of using an `ArrayList` to store the game's temporary items, use a `Vector` object which is implicitly synchronized (all of its methods being locked on itself)
- Remove the locks we had in place previously
- Keep the synchronized block over the for-each loop in `GamePanel::paintComponent` which iterates over the vector of temporary items, but locking it over the vector itself instead. This is necessary because this loop accesses the components of the vector,

and calls `Item::draw()` on each element. Since `draw` is not a method of the `Vector` class, it is not implicitly locked on templates.

- All edits were made in GamePanel
- Commit address: 7588aba63ac3a3072777308ddd67d5b40477a8a6

Move the code which handles despawning of potions from the `run()` loop to the `update()` loop

- This is cleaner and more intuitive, as we are grouping all of the game state updates into the `update()` loop. This is the intended purpose of the `update()` loop.
- All edits were made in GamePanel
- Commit address: 7588aba63ac3a3072777308ddd67d5b40477a8a6