Dungeon Simulator Design Doc

By Roland Jiang and Allen Wang

Outline (Steps I took)

- Setup randomly moving dots
- Setup start and end sprite
- Setup start/stop button for timer
- Added walls and boards (boundaries)
- Assigned dots fitness based on how close they got
- Made population of AI dots have breed method
- Tested to make sure dots learn to get closer
- Messed with numbers and fitness to improve algorithm
- Added enemies so the dots can die
- Worked out how enemies move back and forth
- Added the selection panel with a grid of text fields
- Used mouselistener to track when player clicks on sprite
- Adjusted the appearance of the selection menu
- Added button panel for adding sprites
- Displayed information text at the top
- Added a toggleable setting to only see best AI dot
- Reworked population code to make population settings customizable
- Setup algorithm menu with a grid of text fields
- Made sure algorithm works when player makes changes to the values
- Implemented try catches to make sure player doesn't input bad values
- Added pop ups to make experience better
- Designed a way to save all sprites as letters and numbers into a long string
- Created a way to export the current sprites into a code
- Made an import box where the code can be used to recreate that unique level
- Worked on key listeners
- Implemented player sprite and player testing mode
- Created two levels using the app

Classes

- Vector
    - X and Y
- Sprite
    - Abstract class for all sprites to be drawn
    - X, y, width, height
    - isInBounds(x,y,width,height)
        - For sprite collision using rectangle boundary boxes
- Goal extends Sprite (for graphics)

    - Just a square
- Start extends Sprite (for graphics)
    - Just a square
- Rectangle extends Sprite (for graphics)
    - Just a rectangle
- Board
    - Arraylist of rectangles
    - Inbounds(Sprite)
        - For checking if dots are in “board”
- Enemy
    - Two points (x1, y1, x2, y2)
    - Speed and vector
    - setSpeed
        - Changes speed and calculates vector to travel the path between points
- AI
    - ArrayList of vectors called directions
    - Steps
    - AI(AI bestDot)
    - AI(int step, List vectors)
        - If the list of vectors is less than step we generated randomly the remaining moves
    - Mutate()
        - Randomly alters certain steps
    - getCurrentStep()
        - Gets the correct step from directions
- Population
    - Arraylist of AI
    - All the customizable settings like bestFreq, mutateFreq, maxSize
    - isDone()
        - Returns if all AIs are dead
    - setSpawn()
        - Moves all AI to spawn
    - selectBest()
        - Get best AI based on distance from goal
    - breed()
        - Setup next generation using frequencies and creating new AI objects
- Player extends Dot (for graphics)
    - Just a square
- DisplayBoard
    - paintcomponent()
        - Draws everything based on settings
- App
    - run()

    - buttonPanels
    - keyListeners
        - For controlling player
    - mouseListeners
        - For repositioning and selecting sprites
    - Sliders
    - Panels with grid layout
    - Buttons
    - Checkboxes
    - Text Fields
    - Text Areas
    - Timer
        - Controls all AI, enemies, and players
        - Updates info text

Customization Menus

- Main
    - Add wall
    - Add sentry
    - Add spinner
    - Add board
- Information display
    - Show generation number
    - Show number of dots alive
    - Show number of dots finished
    - Show smallest distance
    - Show least number of steps
- App Player
    - Start/Stop
    - Speed of player (timer)
    - Show all or just best
    - Reset All (same as startup)
    - Delete All sprites
    - Toggle player mode
- AI
    - Population (10-500)
    - Mutation Freq
    - Random Freq
    - Best Freq
    - Top Percent
    - Beginning increment

    - Increment
    - Reset to default
    - Apply changes button (also resets generation)
- Wall
    - Move (Click to update X and Y)
    - Adjust Width and Height (Sliders or dialog)
    - Delete
    - Copy
- Sentry
    - Move Coord 1 (Click to update X and Y)
    - Move Coord 2 (Click to update X and Y)
    - Adjust Speed (slider)
    - Delete
    - Copy
- Spinner
    - Move Center (Click to update X and Y)
    - Adjust Radius (slider)
    - Adjust Speed (slider)
    - Delete
    - Copy
- Board (can’t be dragged)
    - Move (Click to update X and Y)
    - Adjust Width and Height (Sliders or dialog)
    - Delete
    - Copy
- Start/Goal
    - Move (Click to update X and Y)