Bugs And Apples

CS-170-01
Andrew Hoang
May 11, 2022

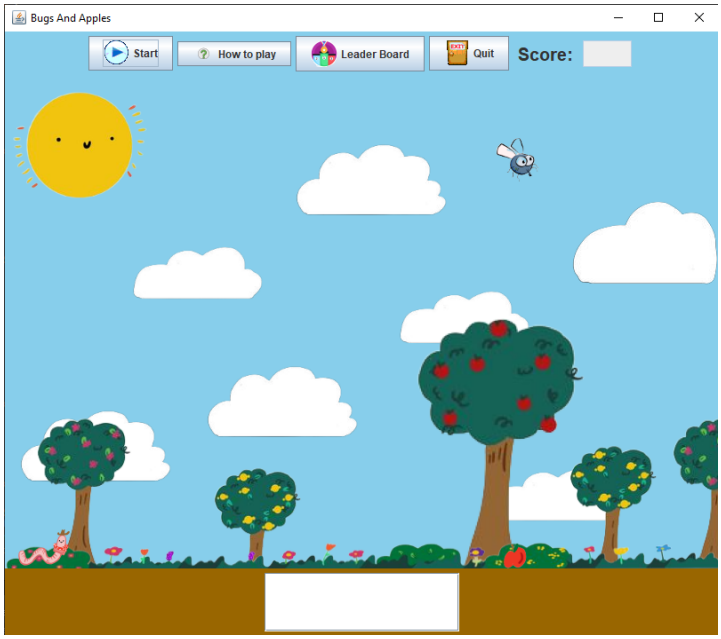
Bugs and Apples is a math learning program designed to help young children learn how to add numbers. It utilizes GUI components, colors, sounds, images, event handling, exception handling, and file I/O methods to create an interactive and engaging experience for players. In the game you control a worm that is found in the left hand side of the window. When the game starts, the user will be prompted to enter in their username. If the user does not enter in a name, they will remain anonymous. The goal of the game is to answer the questions prompted at the bottom of the window as to get to the apple faster than the other bug. Everytime the user obtains an apple, their score will increase by ten. When the user answers a question correctly, a sound will play and the worm will move a certain units forward. If the user answers incorrectly a buzzer will sound, however, the user is not penalized. If the user loses, a sound will play and the user will be notified and shown the leaderboard. The user can also open the leaderboard when they press the leader board button. After the user loses, they can also choose to restart which will start the beginning sequences over again.

The project is made up of ten different classes each with their own unique roles.

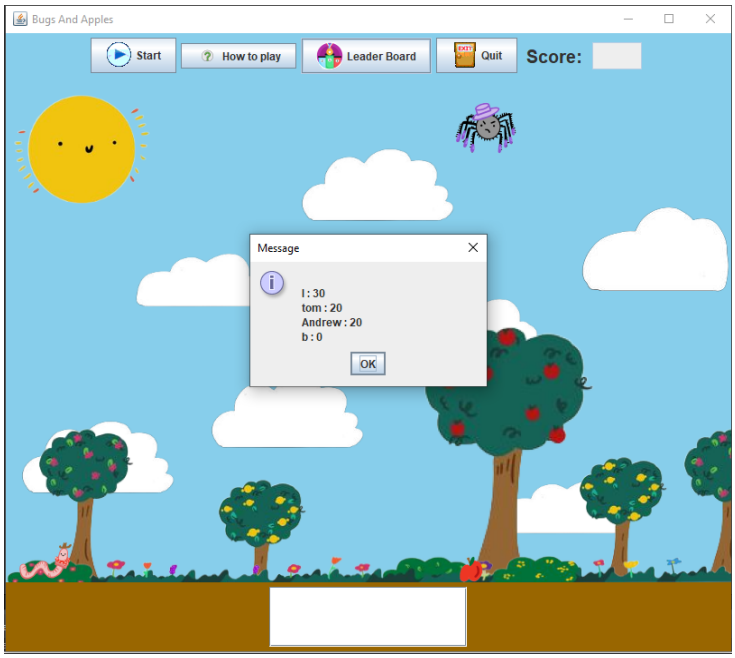
- BugsAndApplesApp
    - Simply the driver class to the other classes
    - Only use is to create BAAFrame object and show to user
- BAAFrame
    - Inherits from the JFrame class which creates an “is a” relationship
    - Centers the window
        - centerWind()
    - Creates a BAAPanel object and adds it to itself
        - BAAFrame()
- BAAPanel
    - Responsible for creating all buttons, labels, text fields and questions
        - BAAPanel()
    - Uses event handling to make the different buttons perform different actions
        - actionPerformed(ActionEvent e)
    - Creates different objects of classes to show data or draw
        - writeLeaderBoard, Font, BAACharacterPanel
    - Inherits from JPanel
- BAACharacterPanel
    - Paints all images in window
        - paint
    - All images are hand drawn
    - Holds all methods to control images
        - stop(), restart(), resume(), move(String input)
    - MultiThreading
        - begin(), paint(Graphics g), run()
            - Uses threads to move worm and bugs
            - Three threads created and started
            - Everytime image is repainted the bug moves one unit
            - Everytime the user answers a question correctly the worm is allowed to move a certain amount of units

        - If the bug gets to apple first
            - User is prompted
            - Sound is played
            - Leaderboard is shown
            - Game is paused threads are stopped
        - If worm gets to apple first
            - User is prompted
            - Sound is played
            - Window is reset and apple is redrawn in random location
- BAANumbers
    - Creates methods to produce random numbers
        - X position
            - ranXPOS()
        - Question numbers
            - ranNum()
        - Bug flags
            - ranBugNum()
- writeLeaderBoard
    - Uses I/O to create a file and store the top highest scores
        - writeDat(String name, int score)
    - Can read I/O file
        - readData()
    - Can collect data from previous games
        - collectPrevious()
- BAAPlayerBase
    - Uses an ArrayList to store Players
    - Uses methods to interact with the ArrayList
        - Add
        - Delete
        - Sorting (based on score)
        - Printing
- Players
    - Object to store information
        - Username
        - Scores
    - Has toString method
- RegexValidator
    - Used to find usernames and scores from textfiles
    - Returns results
- MusicHandler
    - Used to handle sounds

Gameplay Images




Bugs And Apples
How to play
Leader Board
Stop
Quit
Score:
18 + 17 =


Bugs And Apples
Start
How to play
Leader Board
Quit
Score:
Message
I : 30
tom : 20
Andrew : 20
b : 0
OK