Final Project

Course: SDEV140

Instructor: Nick Laplante

Student: Myron Snelson

# Project Status Report

* Name of the GUI application: PEMDAS+
* Purpose: The purpose of this application will be to challenge new computer science students to resolve expressions using correct operator precedence
* Reason: I would like to make the learning of operator precedence fun yet challenging for new computer science students
* Goals:
  + Create an educational application
    - Partially complete – the application is not finished
  + Make the application fun
    - Partially complete – I hope to make it more fun
  + Make the application challenging
    - Partially complete – I hope to make it more challenging
  + Make the application game-like in its approach with lives and scores
    - Complete
  + Allow the user to exit the application after each challenge
    - Complete – the user can always use the Windows close button to exit the application
  + Provide the user encouraging feedback
    - At the end of selected rounds
      * Complete
    - At the conclusion of the program execution
      * Complete
* Outline
  + Display lives and score
    - Complete
  + Set expression length
    - Complete
  + Generate an expression
    - Randomly generate integers for the expression
      * Complete
    - Randomly generate operators for the expression
      * Complete
    - Solve generated expression
      * Complete
  + Display a locked lock image
    - Complete
  + Ask the user to solve and then enter the value of the generated expression
    - Complete
  + If they enter a correct answer on their first attempt
    - Display an unlocked lock image
      * Complete
    - Award them points: 50/rounded resolution minutes (minimum 1)
      * Partially complete – need to add timer logic if possible
    - Update score
      * Complete
    - Increase the length of the expression
      * Complete
  + If they If they enter a correct answer on their second attempt
    - Display an unlocked lock image
      * Complete
    - Award them points: 25/rounded resolution minutes (minimum 1)
      * Partially complete – need to add timer logic if possible
    - Update score
      * Complete
    - Increase the length of the expression
      * Complete
  + If they enter a correct answer on their third attempt
    - Display an unlocked lock image
      * Complete
    - Award them points: 10/rounded resolution minutes
      * Partially complete – need to add timer logic if possible
    - Update score
      * Complete
    - Increase the length of the expression
      * Complete
  + If they fail to enter a correct answer on their third attempt
    - Reduce their number of lives by 1
      * Complete
    - Show them the correct answer
      * Complete
    - Update lives
      * Complete
    - Reduce the length of the expression
      * Complete
  + If remaining lives is zero
    - Rate and display their skill level based on score
      * Partially complete – need to add rating
    - End program
      * Complete
  + Ask the user if they wish to keep playing
    - If “yes”, start another round
      * Complete – game resets itself, they can use exit button if they want
    - If any other reply, rate and display their skill level based on remaining lives and score
      * Not started – not currently using points and lives to rate
    - Each remaining life will be worth 100 points
      * Not started – not currently using points and lives to rate
    - End program
      * Not started – not currently using points and lives to rate
  + Additional items to meet final project requirements
    - A working GUI tkinter application with at least two windows
      * Complete – created main window and pop-ups count
    - Include at least three buttons
      * Complete – game included two buttons
    - Include at least three call back function with each button, including exit button
      * Complete
    - Implement secure coding best practices, including input validation to check if the user entered the correct data type, make sure the entry box is not empty, etc.
      * Complete
    - Validation testing
      * Not started – will do this once application is deemed complete
    - User manual creation
      * Not started – will do this once application is deemed complete
    - The link of the GitHub repository for your final project
      * Complete
      * <https://github.com/midkids/IvyTech.git>
  + Additional items for better design
    - Add color to windows
      * Complete
    - Make hard-coded literals constants
      * Not started – will do once application is completely functional
    - Review Chapter 6 to determine if any more GUI features can be added
      * Not started – will do once application is completely functional
  + Problems encountered
    - Layout took some time
    - Adding logic for parentheses is proving to be complex
    - Will try to keep parentheses logic simple
  + Next steps
    - Add parentheses logic
    - Review all incomplete items listed here and complete if I have the knowledge