PF Project Report

Mohammad Bilal Aziz, k200397, 1-D2

Murtaza Salman, k200398, 1-D2

Abrar Saleem, k200129, 1-D1

January 2021

Quick Maths

Contents

0.1 Introduction

0.1.1 Acknowledgements

First of all, I would like to thank our teachers, **Sir Musawar, Miss Rahemeen, and Sir Kariz** for their support and influence on this project. They helped us in various ways, answering any query we have and helping us in improving our projects. Secondly, we would also like to thank **Shah M Hasan**, for inspiring me to develop this idea and develop a program

0.1.2 About

Our project is a math game in which the user will have to select the difficulty level he/she has to play in. Each level has questions according to its difficulty and the questions are generated randomly. File handling is used to save the high score which will be used to display the rank. Threading is used to run the counter and display the questions simultaneously.

0.1.3 Rules

A counter of 60 seconds will start once the game begins. Random questions will be generated at until the countdown timer ends. A penalty of -5 seconds will be given for incorrect answer in each mode. If answered correctly, then +5 seconds in easy mode, +6 seconds in medium mode, and +7 seconds in hard mode will be granted. Score will be +50 if a question is answered correctly. If the user answer questions correctly in a row, he/she will have a win streak. If win streak is greater than or equal to 3, then +70 score will be granted. If win streak is greater than or equal to 5, then +90 score will be granted. Any wrong answer will reset the win streak to 0.

0.1.4 Info

The total score will be displayed at the end and if the user's score is greater than the high score, then high score will be updated. Else, it would display the score only. If the high score of the user is less than 100, the user's skill rank will be "Noob". If the high score is between 100 to 500, skill rank will be "Semi Pro". If it is between 500-1000, skill rank is "Pro". If it is more than 1000, the user's skill rank will be "Expert". If you want to play Medium level, your skill rank must be Pro. If you want to play Hard Level, your skill rank must be Expert.

0.2 Background

0.2.1 Research

For this project, we researched many things because some of the functions and methods used were not taught to us, so we took help from external sites like [**Tutorials Point**] and a little help from [**Saurabh Shukla**] and [**Stack Overflow**]. The concept of threading was new to us so we researched particularly about this method, and implemented it in our program. Also, we looked up on websites like [**Geeksforgeeks**] to help us develop the idea of random numbers and chose a random character from array.
Along with these, we did a number of researches to fully develop the final program we now have.

0.2.2 Project Selection

This idea and notion of this project was inspired by Shah M Hasan(4th Year NED). He developed a similar game like this. Once we got the idea, we started developing this game and along with the help of our teachers, we added some more features to give the program more complexity and let the user have more fun while playing this game.

0.3 Problem Analysis

In this program, random problems needed to be generated. Also, the high score should be saved and displayed on another file. Furthermore, we had to implement a counter to ensure that the user completes the question in the given time.

For random functions, we used stdlib.h and time.h header files to import the functions needed to generate the random problems. Operators generated should be random too. So, we introduce a random number to generate number from 0 to (n-1), where n = number of operators, and stored the operator in a character array. This generates the operator at random too. Hence, the questions generated are completely at random.

Then we used the method of filing to save the high score. Once the high score is saved in a text file, the program will display the high score in the end. If the current score is greater than high score, the program will overwrite the previous high score and update the new one. The high score also ensures the rank of the player. If the player's rank, according to the high score, is PRO or above, then he/she can play medium level. In order to play the hard level, the skill rank should be EXPERT.

Implementing a counter was a tough part. For the program to run correctly, we need to run the counter in background while the screen is displaying the question. But C is a sequential language. Hence, in order to run the counter while displaying the problem, the concept of threading is used, which allows user to multi-task. So, in this way, counter was implemented by using a while loop and then running it simultaneously with the question.

These three were the main problems which we had to counter in our program and the methods explained our how we achieved our solution.

0.4 Problem Analysis

Several Methods of C language are used in this project. They are listed below:

- Conditional Statements
- Functions
- Loops
- Pointers
- Filing
- Threading

0.5 Implementation

There are many tools and techniques used in this program:

- Dev C++ and Codeblocks GNU compiler is used to compile and run the code
- Libraries used are:
  - stdin.h [for standard input/output functions]
  - conio.h [for getch () function]
  - uninstd.h [for sleep () function]
  - string.h [for importing string functions]
  - stdlib.h [for importing rand () and srand () function]
  - pthread.h [for threading functions]
  - math.h [Math library used for importing math functions]
  - time.h [To generate random numbers by using time () function]

0.6 Results

0.6.1 Findings

The program was executed properly and after running the code several times and doing some debugging, we managed to remove all the bugs from the code. The final result was just as expected.

In the process of developing this program, we came across many features and were able to look in the code more professionally, as it is the first major project of Programming we have been given. Apart from that, some of the features like, GUI and continuously running timer was not implemented in this program, we plan to implement it in future to widen our knowledge.

0.6.2 Summary

The program asks to enter the difficulty mode, and if the user's skill rank is sufficient to enter that mode, the user can play. The user then solves a series of random problems in which he/she gets points and bonus time for each correct answer, and penalty for wrong answer. The user will have advantage if he/she has a win streak. When the counter reaches 0, the time will end and the total score will be displayed. If it is more than the high score, then it will update the high score in the file, other wise the program will display the the current score and the high score.

The link of repository on github of the code is provided below:

https://github.com/Bilal-01/PF-Project.git

0.7 Conclusion

So, after using a variety of different tools and techniques, and combining different methods, we developed this math game. Due to limited time and skills, we were unable to add some more features, which we plan to develop in future. Some of those features are:

1. User Graphics Interface
2. More advanced difficulty levels
3. More efficient and reliable code

Apart from these features, we also plan to install other features which we will learn in time.