CS118 Programming Fundamentals

Fall 2023 FAST-NU, Lahore

Project – Candy Crush

Almost all of us are familiar with the game Candy Crush and many of us might have tried it at least once in our lives. We'll try to develop a similar game in C++ using SFML graphics.

Your version of Candy Crush must have the following features:

1. There are only two possible moves: any two tiles can be swapped vertically or horizontally. You can implement controls using mouse or keyboard.
2. Whenever a user selects a tile to swap, it must be highlighted.
3. The board configuration must be generic: we just define the number of rows and columns (globally), and the board and the tiles are generated accordingly. You can define these parameters globally using macros or *const int*.
4. The following combinations of tiles must be possible (at the very least):

3 horizontal tiles	3 vertical tiles	3 diagonal tiles (in any orientation)	3 tiles forming an elbow (in any orientation)
4 horizontal tiles	4 vertical tiles	4 diagonal tiles (in any orientation)	
5 horizontal tiles	5 vertical tiles	5 diagonal tiles (in any orientation)	

5. If no moves are possible, shuffle the board (as happens in the original game).
6. The user must be able to save and load the game. Whenever a user saves a game, the current score, remaining moves and the current board configuration must be saved so that the user continues the game exactly from where he left off.
7. User can ask for a hint and the game must highlight some possible move.
8. Implement special tiles for some of the nicer combinations. Implement this feature for combinations of 4 and 5 tiles:
    - 4 Tiles: Whenever any combination of this special tile is made, it explodes 3 X 3 region around it.
    - 5 Tiles: Whenever any combination of this special tile is made, it explodes the row and column in which this special tile was present.
9. Decide reasonable scores for all the moves/combinations.
10. **Bonus:** Implement timer for each move. For example, give user some time (say 10s) to make a move. If he doesn't make a move within the allotted time, cancel that move (i.e. decrement moves) and penalize him (by deducting some score). Don't forget to show the timer clearly on the screen.

Note

- Variable and function names should be meaningful and without abbreviations.
- Marks will be deducted if you use any other library (e.g. string) or data structure (e.g. dynamic arrays, vectors, maps, etc.) of C++ that hasn't been taught during the course.
- Marks will be deducted for a messy and confusing code.
- Bonus marks will be awarded only if the core functionalities are complete.
- Plagiarism will not be tolerated. Plagiarism in any mean might result in the deduction of absolute marks or F in the course.

Submission

Submit a zip file named with your roll number. For example, a student with roll number 20L-1234 will submit his zip file named 20-1234.zip. Your zip file must contain:

1. A source code folder containing the source code and all resource files (e.g. images, audios, etc.)
2. A folder containing game executable (you might need to place all resources too in this folder)
3. A readme file which tell the user how to play your game

Happy Coding 💪
Have Fun ☺