Computer Organization & Assembly Language Fall 2021
Final Project

Number Crushing Game

Deadline: 11:59 PM, 31 December 2021

Instructions:.

- You are required to work in MASM in Visual Studio with Irvine Library. Project incompatible of MASM in Visual Studio with Irvine Library will not be considered.
- A group of maximum size **three** is allowed.
- Cross Section groups are **not** allowed.
- User Interface is important in this project. Try to develop an attractive user interface.
- Use of extra features in the project is encouraged.
- Use good programming practices (well commented and indented code; meaningful variable names, procedures, readable code etc.).
- Only one group of the student submit the project in Zip File.
- Evaluation Criteria will be shared with you soon.
- Copy/cheating case will be awarded an "F" grade in the course.

Game Description

Number Crush is a **"match-three" game**, where the core game play is based on swapping two adjacent random values among several on the game board to make a row or column of at least 3 matching-random values. On this match, the matched random values are removed from the board, and **random values above them fall into the empty spaces**, with new random values appearing from the top of the board. This may create a new matched set of random values, which is automatically cleared in the same manner. The game is split among many levels, which must be completed in sequence.

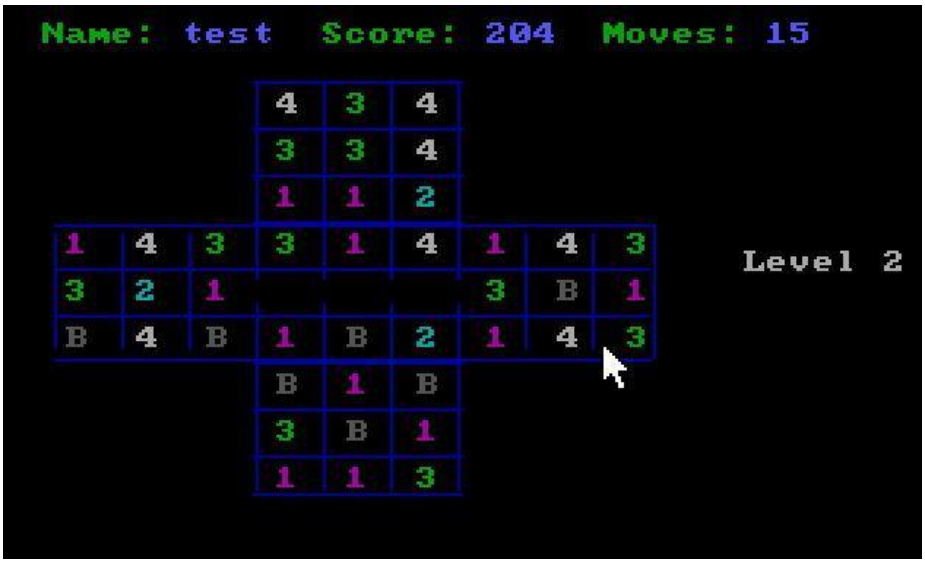
LEVELS

Level 1

Level 1 has a 10x10 board. When a number is swapped with another number, if a combo exists, the combo is crushed, dropped, and score updated accordingly. Otherwise, the numbers are swapped back! The board is filled with random numbers from 1 to 5. It has bomb 'B' too. When a number is swapped with bomb, all of its occurrences are destroyed.

Level 2

It has a plus shape board with same functions of level 1. The board is filled with random numbers from 1 to 4.



Level 3

It has same board shape as level 1 but we have blockers ‘X’ too for restricting player movement.

How to Update SCORE AND MOVES

1) During crushing, the score added depends on the size of combo. A combo of 3 adds 3 to the score. A combo of 4 will add 4 and so on.
2) During explosion, it is different though. The added score depends on how many occurrences are destroyed and from which location they are destroyed. If a number is at bottom, more numbers will have to be dropped from top and hence more score.
3) The user is given a total of 15 moves in each level. When all moves are finished, user is promoted to new level and moves are again reset to 15.

File Handling

- All Individual levels score will be saved in the file.
- Stores the highest score and player name in a same file.
- Record in a file should look like in the format given below

Ali Raza
Level 1: 20
Level 2: 30
Level 3: 46
Highest Score: 46

BONUSES

- When bomb is used, all occurrences of the exploding row/col are first highlighted for a second, then explosion proceeds.
- If after swapping, no combo exists, the numbers are swapped back.
- Changed background and look of level 3.
- The string ‘crushing’ is displayed when combos are being crushed and score is being updated.
- ‘Explosion’ is displayed when a bomb destroys a row or col in board.

Divide and Conquer

You need to write different procedures to help yourself out in this project. You can use some these procedures (not mandatory) for performing the game. And you also need to write extra procedures for complete functionality. The main method should consist of only calling other procedures.

populateBoard (.)

It populates the board using random numbers.

drawBoard (.)

This includes drawing border.

updateBoard (.)

This function checks for combinations, crushes them, auto-fills and drops until all the combos are removed from the array. It makes use of another function **checkCombo** which is called in a loop unless and until the value of 'crush' variable is set to 0. 0 means there are no more combinations in our board array. Vertical and horizontal combos are checked.

takeInput (.)

It takes input for two cells. Then if cells are adjacent, they are swapped. If no combo is formed after swapping, the no's are swapped back.

initiateBomb (.)

If any no. is swapped with bomb, it destroys all values of that row or column through which number is swapped by bomb.

drawString (.)

This function draws the player's name, score and moves left on the top. Along with this, it also shows the current level, and strings of 'explosion' and 'crushing'.

Good luck ☺