MINESWEEPER GAME

END TERM REPORT

by

AAKASH, SUMEET SIHAG

Section: K19PT

Roll Numbers: 14, 05

Department of Intelligent Systems,

School of Computer Science Engineering,

Lovely Professional University, Jalandhar

November, 2020

Student Declaration

This is to declare that this report has been written by us. No part of the report is copied from other sources. All information included from other sources have been duly acknowledged. We aver that if any part of the report is found to be copied, we are shall take full responsibility for it.

Name: Aakash, Sumeet Sihag

Roll Number: 14, 05

Date: 31st October, 2020

TABLE OF CONTENTS

BONAFIDE CERTIFICATE

Certified that this project report “MINESWEEPER GAME ” is the bonafide work of “AAKASH” and “SUMEET SIHAG” who carried out the project work under my supervision.

Name- Dr. Dhanpratap Singh
UID- 25706
Department- School of Computer Science and Engineering

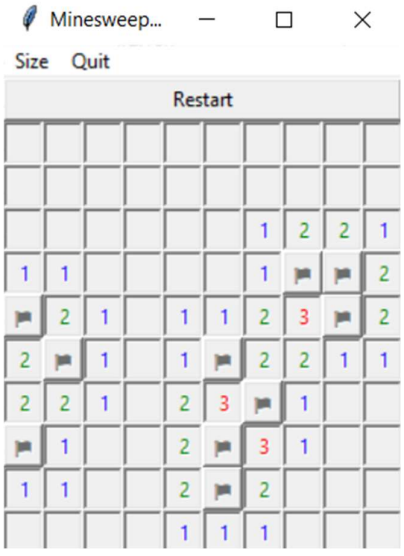
Background and Objectives of the project

- **Background:**

Minesweeper is a single-player puzzle game that consists of a grid of cells, where some of the cells contain hidden “mines”. Clicking on a cell that contains a mine detonates the mine, and causes the user to lose the game. Clicking on a “safe” cell (a cell that does not contain a mine) reveals a number that indicates how many neighboring cells – where a neighbor is a cell that is one square to the left, right, up, down, or diagonal from the given cell – contain a mine.

- **Objective:**

The objective of the game is to clear the rectangular board containing hidden "mines" without detonating any of them, with help from clues about the number of neighboring mines in each field. Playing Minesweeper involves a fair amount of logic. A clever player will use the numbered cells to deduce the location of mines. For assistance, most implementations of Minesweeper allow the player to mark or flag possible mine locations. However, this is simply for bookkeeping as the game does not validate any flagged squares. Higher difficulties of Minesweeper involve a greater degree of deductive reasoning as the mine density (number of mines over number of cells) increases. Oftentimes, mines cannot be deterministically located, and so the player must resort to guessing.



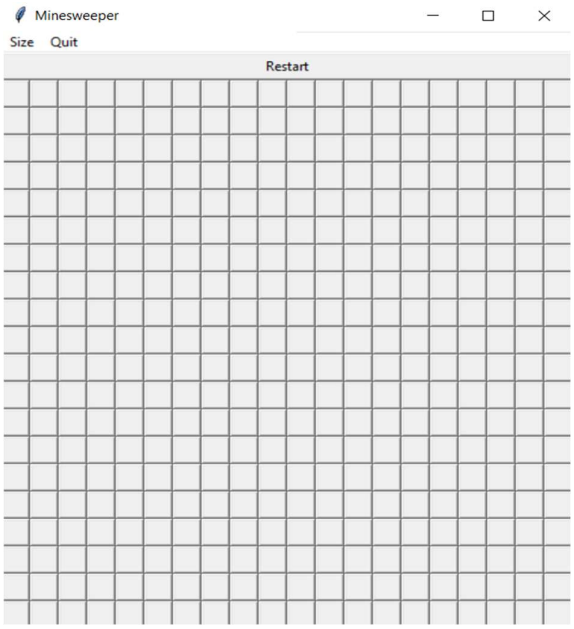
(Figure showing flagged cells in a Minesweeper game.)

Description of the project

- **Description:**

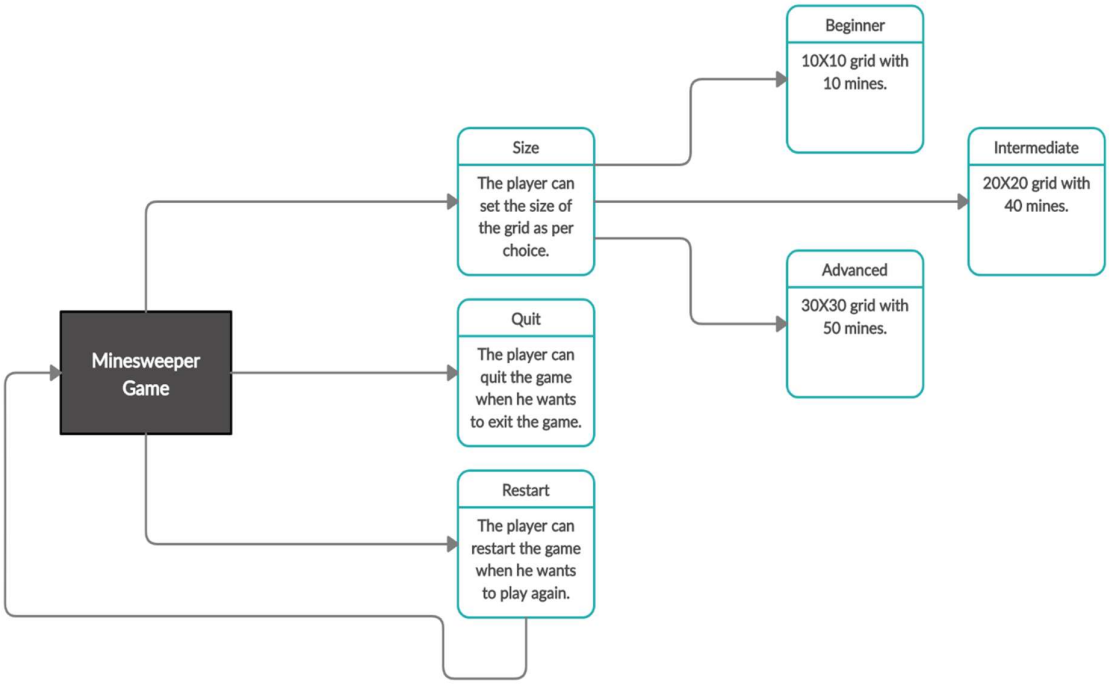
    - In Minesweeper, mines are scattered throughout a board which is divided into cells. Cells have three states: uncovered, covered and flagged. A covered cell is blank and clickable, while an uncovered cell is exposed. Flagged cells are those marked by the player to indicate a potential mine location.
    - A player left-clicks a cell to uncover it. If a player uncovers a mined cell, the game ends. Otherwise, the uncovered cells displays either a number, indicating the quantity of mines adjacent to it, or a blank tile (or "0"), and all adjacent non-mined cells will automatically be uncovered.
    - Right-clicking on a cell will flag it, causing a flag to appear on it. Flagged cells are still covered, and a player can click on them to uncover them, although typically they must first be unflagged with an additional right-click.
    - To win the game, players must uncover all non-mine cells, at which point the timer is stopped. Flagging all the mined cells is not required.
    - A Minesweeper configuration is a grid, typically rectangular, of possibly covered squares that may be partially labeled with numbers and/or mines.
    - A configuration can be thought of as the state of a Minesweeper game, including all the numbers, marked mines, and covered squares. A solution to a configuration is an assignment of mines to the covered cells which gives rise to a consistent Minesweeper grid.
    - A Minesweeper board is said to be consistent if there is some assignment of mines to the blank/covered squares that gives rise to the numbers shown.
    - If a player believes a square contains a mine, he/she is allowed to place an indicator on that cell. These are called marked or flagged squares.
    - An unmarked square is one that has no flag or mark on it to denote it has a mine.
    - A covered, unprobed, or unknown cell is a position whose contents are unknown to the player. On the other hand, the contents of an uncovered or probed cell are available to the player.
    - A mine-free or simply free cell is one that does not contain a mine, it typically refers to a covered square that the player can safely click or probe.

- **How to play:**
  - Given to the player is a square grid consisting of "n x n" hidden cells. Any hidden cell contains one of the following:
    i. A Mine
    ii. Blank Cell
    iii. A number (say X) indicating that the cell has X mines in its neighborhood. Note that blank cell could be represented by number "0".
  - The mines are randomly placed on the grid.
  - The rules of the game are as follows:
    i. If the player chooses a cell that has a hidden mine, the game is over. Explode!
    ii. If the player chooses a cell that has a hidden number, then the spot is uncovered and the number is now visible to the player.
    iii. If the player chooses a cell that doesn't have a mine or number, then ripple effect takes place (see section "Ripple effect" below) uncovering all the blanks in the neighborhood until either the grid boundary is reached or a number is reached along its path. The objective of the game is to identify, logically, all the mines on the grid.



(Figure showing 10X10 grid of Minesweeper game.)

- **<u>Pictorial flow of the Project</u>:**



(Figure showing Pictorial flow of the Project.)

Description of Work Division in the Project

- Group member- 1

  (Name- Aakash, Reg. No.- 11912274, Roll No.- 14)

  - Project Contributions :--
    i. Preparation of game window.
    ii. Preparation of the game board/grid.
  - Report Contributions :--
    i. Pictorial flow of the Project.
    ii. Technology and Framework used.
    iii. Screenshots of Implementation of the Project.

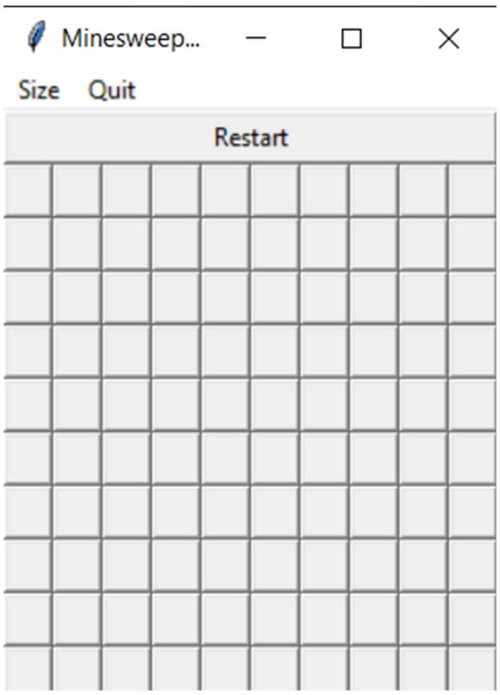
- Group member- 2

  (Name- Sumeet Sihag, Reg. No.- 11912288, Roll No.- 05)

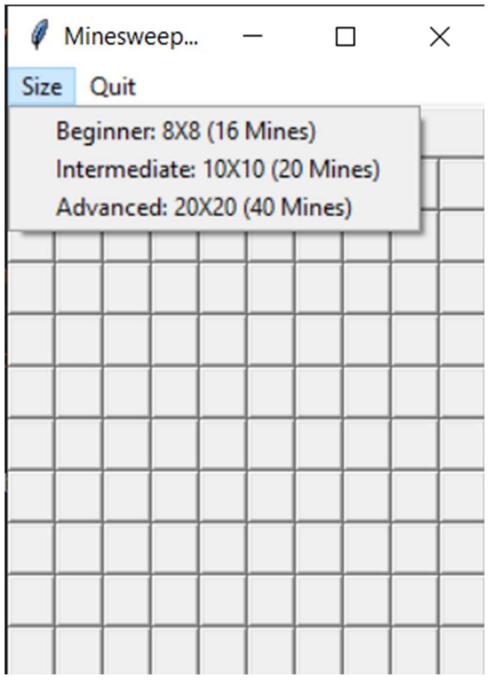
  - Project Contributions :--
    i. Preparation of menu bar.
    ii. Preparation of the functions with mouse clicks.
  - Report Contributions :--
    i. Background and Objectives of the Project.
    ii. Description of the Project.

Screenshots of Implementation of the Project

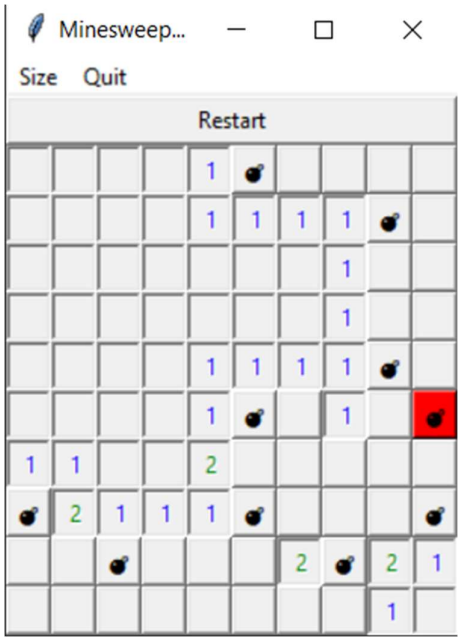
1) First View of the Window :--

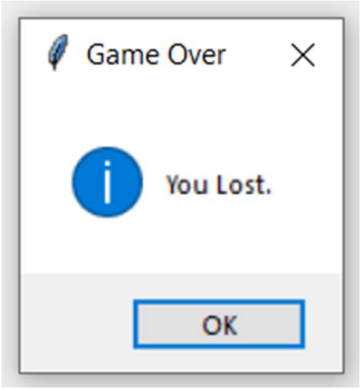


2) Sizes available :--

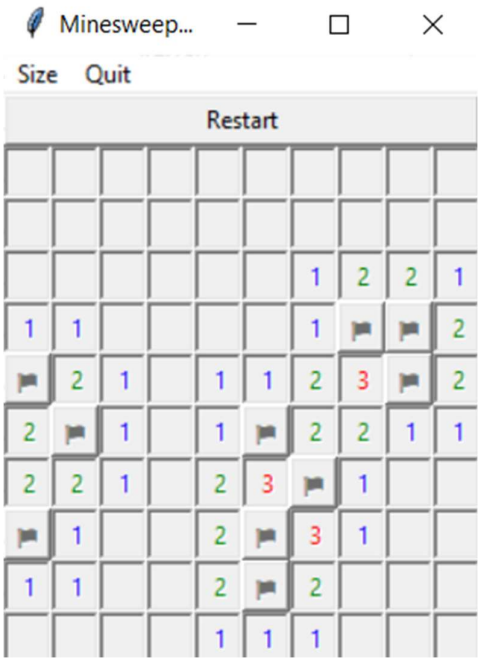


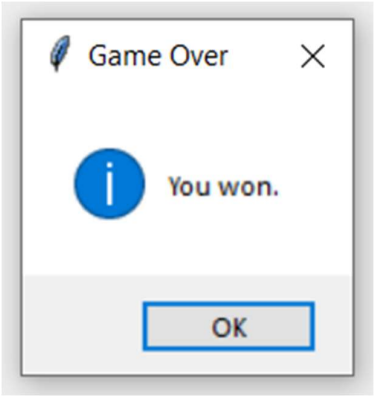
3) After losing the game :--





4) After winning the game :--





Technologies and Framework used in the Project

1) Python GUI :

   In this project, the method of GUI used is Tkinter. Python with tkinter is the fastest and easiest way to create the GUI applications. Creating a GUI using tkinter is an easy task.

   There are two main methods used in Tkinter to create a python program :

   a) Tk()
   b) mainloop()

   Tkinter also offers the access for the configuration of the window :

   a) pack()
   b) grid()
   c) place()

2) Widget :

   Widgets are the elements of Graphical User Interface (GUI) in python that displays the information or gives a way for the user to interact with the OS.

   The widgets used in this project are as follows :

   a) Label
   b) Button
   c) Message
   d) Text
   e) Menu

3) Python functions :

   A function is a block of organized, reusable code that is used to perform a single, related action. Functions provide better modularity for your application and a high degree of code reusing.

   Syntax of writing a function-

   def function_name():

   Function description…

   ---

4) Python Lists :

   A list is an ordered set of values, where each value is identified by an index. Lists are similar to strings, which are ordered sets of characters, except that the elements of a list can have any type. In this project, lists are used in designing the grid of the game and for adding colors to the grid.

5) Random :

   The random module is used to generate random numbers in any program. In the project, random module is used to generate random numbers and mines randomly in the grid for the Minesweeper game. We just have to write “import random” at the top of the program to import the random module if required in the program.

- **References :**
  - https://www.tutorialspoint.com/python/python_gui_programming.htm#:~:text=Tkinter%20is%20the%20standard%20GUI,Tkinter%20is%20an%20easy%20task.
  - https://www.tutorialspoint.com/python/python_gui_programming.htm
  - https://docs.python.org/3/library/tk.html

- **Project GITHUB link:**
  https://github.com/Aakash265/K19PT-INT-213-PROJECT