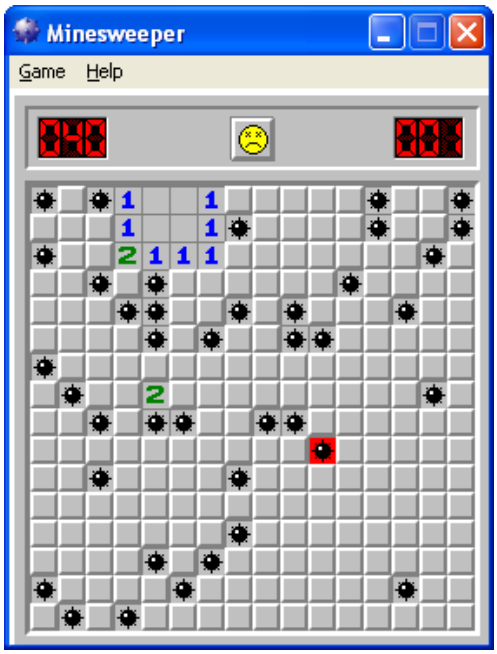
COMP 1451 – Assignment #1 (50 points)
Due: 11:59 p.m. the night before Session 3

Many computer games are based on the notion of a grid, i.e. a two-dimensional array. Some examples: Minesweeper, Battleship.



Minesweeper

Your assignment is to implement a GuessingGame project. You are given an outline for the project consisting of four classes: InputReader, CoordinatePair, Grid, Game. Classes InputReader and CoordinatePair are complete and require no changes. Classes Grid and Game have the instance variables declared, and all the required methods as "stubs", i.e. each method has a comment explaining what it will do, and the method signature. Your task is to complete these methods and implement the game. Feel free to add other classes and/or methods as you see fit.

Here is how the GuessingGame works:

- A two-dimensional array of chars (the working grid) is constructed. Use constants for the dimensions (number of rows and columns). By default Java will set the contents of the grid to "null" chars. This is char `'\0'`.

- A "target" char (you decide which char) is placed in an arbitrary number of randomly-chosen grid locations, e.g. `'T'` is placed in five different grid locations, randomly generated. Be sure to fill five <u>different</u> locations if five is your chosen number of targets. (Check for duplicates.) Use constants for the target char and the number of locations to fill.

- A second grid is constructed of the same dimensions as the original. This is the display grid, used to give feedback to the user.

- The user is prompted a number of times to type grid coordinates. (Use a constant to specify how many tries the user gets.) For each pair of coordinates the user types, the display grid is updated (and re-displayed) with one symbol for a “hit” and another for a “miss”. A “hit” occurs when the user’s choice of coordinates matches the placement of the target char in the working grid. The “hit” and “miss” chars replace the default null chars in the appropriate locations in the display grid. If the user types invalid coordinates an error message is displayed and the user is prompted to try again.

- Give your user clear instructions – this is important any time you write software for people to use and will be expected throughout the course even if it is not explicitly mentioned in the requirements.

- When the user’s turn has ended a message is displayed telling how many targets were hit, and the working grid is displayed.

Think before you code! What are the attributes and responsibilities of each class? How do the classes relate to each other?

Marks will be given for:
- Style – see the style guide Appendix J in your textbook.
- Correctness and completeness – code meets the requirements listed above.

Create a .zip file containing your entire BlueJ project (zip the folder, not the individual files). Name the .zip file with your name and the assignment number, e.g. “Susan_Wong_Assign_1.zip”. Upload the file to WebCT before the cutoff time.

Below is a sample screen shot at the end of a game. The sample shows the final prompt to the user, and the resulting display. Note that we do not expect fancy graphics.