THE UNIVERSITY OF SYDNEY

INFO1113 / COMP9003 Assignment Warmup

Due: 8 September 2024, 11:59PM AEST

This assignment is worth 4% of your final grade.

Task Description

Implement the Minesweeper game using the Processing library for graphics and gradle as a dependency manager. You can access the documentation from here. As with any assignment, make sure that your work is your own, and do not share your code or solutions with other students.

Each tile in the 18x27 grid is initially blue (hidden), and may contain a mine. Your program must accept a parameter from command line arguments, which is the number of mines that should be spawned randomly on the board. If no parameter is provided, or an invalid one is, then the default is 100 mines.

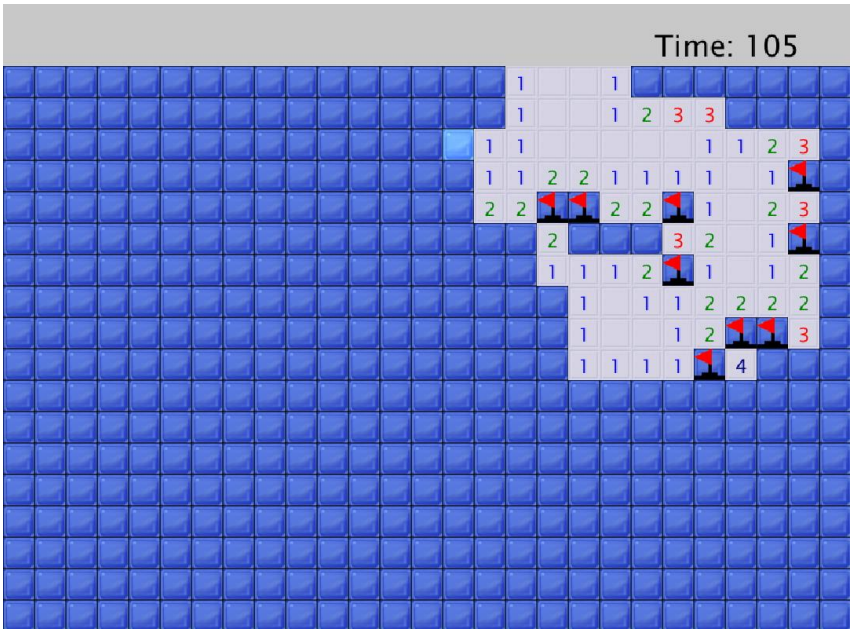

Figure 1. Minesweeper example game in progress.

The player can left click on a tile to reveal it. If there is a mine, all mines will begin exploding and the game ends. Display the message “You Lost!” in the top bar. The mines explode in a cascade/progressive manner, where each explosion will begin 3 frames after the previous one began.

The mine animation image sequence is provided in the scaffold, comprising of the 10 files “mine0.png” to “mine9.png”.

If there isn’t a mine on that tile, the tile will show a number representing how many adjacent mines there are. The colours for these numbers are given to you in an int[] array for RGB values in the scaffold, `mineCountColour`. If there are no adjacent mines, it remains blank, and all adjacent tiles will also be revealed. If all non-mine tiles are revealed, the game ends. Display the message “You win!”. The player can right click on a tile to flag () or unflag it. If flagged, it cannot be left-clicked to reveal.

A timer in the top-right corner records how many seconds elapsed since the game began. The player can press ‘r’ to restart the game.

Marking Criteria (4%)

Your final submission is due on Sunday 8 September 2024 at 11:59PM. To submit, you must upload your build.gradle file and src folder to Ed. Do NOT submit the build folder. Ensure src is in the root directory with the other files, and not part of a zip, then press MARK.

Shown during tutorial in week 5 (-1 deduction if your submission breaks these features)	• Window launches and shows initial layout correctly with blue cells. • Hovering over tiles with cursor changes the blue tile image to highlight it • Left clicking on a cell reveals it, to show the grey tile underneath • On each frame, cells adjacent to a revealed blank cell are revealed • If a player reveals all tiles without clicking on any mines, they win the game
0.5 mark	• Mines are randomly placed on the board when the game starts, depending on the first given command line argument input (default 100 if not provided) • Timer counts up every second in the top-right corner of the screen. It stops when the game ends.
0.5 mark	• Right click on a blue cell causes it to be flagged as containing a mine. The flag indicator is shown, and the tile can no longer be left clicked to reveal it, unless the flag is removed by right-clicking again.
1 mark	• Revealed tiles show the correct number and associated colour depending on the count of adjacent mines (1-8 or blank).
1 mark	• If a player clicks on a mine, it explodes, along with all other mines on the board. This causes the player to lose the game, and they can press the key ‘r’ to restart, with a new set of random mines. Display message “You lost!”.
1 mark	• Mine explosion is animated with each image lasting 3 frames. • The mines explode in sequence such that each explosion only begins 3 frames after the previous one begins its explosion.

Academic Declaration

By submitting this assignment you declare the following:

I declare that I have read and understood the University of Sydney Student Plagiarism: Coursework Policy and Procedure, and except where specifically acknowledged, the work contained in this assignment/project is my own work, and has not been copied from other sources or been previously submitted for award or assessment.

I understand that failure to comply with the Student Plagiarism: Coursework Policy and Procedure can lead to severe penalties as outlined under Chapter 8 of the University of Sydney By-Law 1999 (as amended). These penalties may be imposed in cases where any significant portion of my submitted work has been copied without proper acknowledgment from other sources, including published works, the Internet, existing programs, the work of other students, or work previously submitted for other awards or assessments.

I realise that I may be asked to identify those portions of the work contributed by me and required to demonstrate my knowledge of the relevant material by answering oral questions or by undertaking supplementary work, either written or in the laboratory, in order to arrive at the final assessment mark.

I acknowledge that the School of Computer Science, in assessing this assignment, may reproduce it entirely, may provide a copy to another member of faculty, and/or communicate a copy of this assignment to a plagiarism checking service or in-house computer program, and that a copy of the assignment may be maintained by the service or the School of Computer Science for the purpose of future plagiarism checking.