

**Instructions:**

Design an original piece of Software using Processing that incorporates all skills learned in ICS4U

**Program requirements:**

1. All work must be clearly documented (Mindomo and Processing, Gantt chart, GitHub Classroom)
2. All code must be well documented (comments) and neatly formatted as per conventions (i.e. variable naming conventions).
3. The program should be organized using Object Oriented Programming (OOP) (eg. classes) and fully detailed in organizational charts (eg. hierarchy diagram in Mindomo)
4. You must maintain a Mindomo chart plan showing how you organized the program (eg. flow chart diagram to detail the **menu driven** game).
5. **The program should make use of all programming concepts covered in the course. More to come**
6. The program should be as crash-proof as possible.
7. You must submit a test plan and test data which demonstrates how you checked for stability.

**Grading:**

- 15% of your final mark will consist of the initial game design incl. external documentation (i.e. Gantt Chart, Flow chart diagram in Mindomo, pseudocode, GitHub ongoing code maintenance, ongoing planning.) This will be included in your term mark.
- The internal documentation, code and quality of this summative will count for approximately 85% of your final summative mark and will be assessed through multiple means (i.e. interviews, digital submissions on GitHub, Mindomo documentation)

Your individual mark will reflect the way in which you have integrated the programming concepts that you have learned in this course. (eg. MD arrays, subroutines (methods), OOP, repetition, selection, searches & sorts, File IO)

- Your final mark will also be adjusted to reflect the level of the task you have taken on. (i.e. Higher level for properly implementing a difficult concept or difficult new rule or developed random/skilled computer players – See Mr. Linseman if this is not feasible)
- If you work as part of a team, each task/detail of the project must be credited to an individual who was in charge. Also, every programming concept needs to be covered by each member.

The final overall grade is based on your classroom production which will be monitored.

Interviews: Interviews **may** be used to properly assess your team's understanding of the parts of your team's program. During the interview(s), which will take place throughout or towards the end of the semester in class, be prepared to fully explain all parts of your team's program's code. If you are unable to explain how each portion of your software, this will negatively affect the overall grade. Therefore, do not simply plagiarize programs already out there or a mark of **ZERO** will result!

Teacher Comments: See GitHub Classroom Readme File.

**Overall:**

Code for Summative (Level for portion out of 85%): \_\_\_\_\_

Summative Planning (Level for portion out of 15% - Part of Term): \_\_\_\_\_