ICS4U Final Project Details – 15% of final mark

What you need to submit: (The due dates are listed under timeline.)

A proposal of your final project:

In a group of 1 or 2, clearly state what the final program your group intend to create will do. Explain the reason(s) why your group has chosen this topic. Two pages maximum.
Hint: Write programs that are useful to someone. See if your other teachers have tasks that you can help using a computer program.

A program planning document:

A rough diagram of the GUI. A flowchart structured diagram or pseudocode demonstrating interaction between functions, procedures, event listeners, variables, files IO, etc. A description of your data modeling (classes, data structures) process. A list of tasks that each group member will be responsible for.

A progress report:

A brief description of what has been completed and what needs to be done. In addition, mention any challenges that has been overcome and challenges that needs to be resolved. A copy of your Gantt Chart 1 page per group.

A testing report:

A complete list of the testing processes that you have used to test your code.

At the final evaluation meeting:

You have to hand in the following:

A list of contributions from each group member.
An instruction manual in how to use the program.
Upload program and any data files (zipped) to Moodle.

The meeting itself consists of:

Prepare a demo of program with your own computer to show me.
Answer a few questions regarding your code and how your program works.

The program itself:

- Must contain at least one OOP class.
- Must contain a GUI of some kind.
- Must involve file IO.
- Must not crash.
- Must contain some new learning.

Timeline:

Jan 9th – An **approved** proposal.

Jan 15th – A progress report (meeting)

June 19th – Program should be near completion. Begin testing and manuals.

June 24th – Program demonstration meeting. (Last day of classes)

Marking scheme

Group

Proposal:	1	2	3	4
Program Planning document	1	2	3	4
Gantt Chart	1	2	3	4
Progress Report (C):	1	2	3	4
Testing Report (K):	1	2	3	4
Instruction Manual (A):	1	2	3	4
Program Demo (C):	1	2	3	4
GUI Design (K&A):	1	2	3	4

Individual

Final Code: (based on code you did if in a group)				
Use of proper variables (K&A):	1	2	3	4
Use of proper header & comments (C):	1	2	3	4
Readability (A):	1	2	3	4
User Friendliness (K):	1	2	3	4
OOP Implementation (T&A): *** x2**	1	2	3	4
File I/O Implementation(K&A)	1	2	3	4
Difficulty (T&A): *** x2**	1	2	3	4
Additions (T&A):	1	2	3	4
Proper use of class time (K&A):	1	2	3	4
Responses to questions during final evaluation (C):	1	2	3	4

TOTAL / 80

ICS4U Final Project Rubric

Category	Level 1	Level 2	Level 3	Level 4
Knowledge/ Understanding	• Demonstrates limited knowledge and understanding of programming terminology • Demonstrates limited knowledge and understanding of proper coding techniques.	• Demonstrates some knowledge and understanding of programing terminology • Demonstrates some knowledge and understanding of proper coding techniques.	• Demonstrates considerable knowledge and understanding of programming terminology • Demonstrates considerable knowledge and understanding of proper coding techniques.	• Demonstrates thorough knowledge and understanding of programming terminology • Demonstrates thorough knowledge and understanding of proper coding techniques.
Thinking	• Instruction are produced with errors. • Program is written with incorrect techniques and methods • uses critical thinking processes with limited effectiveness	• Instruction are produced functionally. • Program is written with poor techniques and methods • uses critical thinking processes with some effectiveness	• Instruction are produced with accuracy. • Program is written with correc techniques and methods • uses critical thinking processes with considerable effectiveness	• Instruction are produced with creativity and accuracy. • Program is written with advance techniques and methods • Uses critical thinking processes with a high degree of effectiveness
Communication	• Uses technical terminology incorrectly. • Instructions are incomplete. • Instructions are aimed at the wrong level for the audience.	• Uses technical terminology correctly sometime. • Instructions are complete, but difficult to follow. • Instructions are confusing to the target audience	• Uses technical terminology correctly at most of the time. • Instructions are complete. • Instructions are undertandable to the target audience.	• Uses technical terminology correctly at all time. • Instructions are complete and easy to follow. • Instructions are aimed at the right level for the audience.
Application	• Demonstrates poor understanding between the program and its visual representation. • Instructions cover little part of the program. • Code is written with few proper code maintanence technique.	• Demonstrates some understanding between the program and its visual representation. • Instructions cover some aspect of the program. • Code is written with some proper code maintanence technique.	• Demonstrates good understanding between the program and its visual representation. • Instructions cover most aspect of the program. • Code is written mostly with proper code maintanence technique.	• Demonstrates excellent understanding between the program and its visual representation. • Instructions cover all aspect of the program. • Code is written with proper code maintanence technique.