**ICS 4UI Major Programming Project**

**Object Programming:**

Much of this major programming project will focus around your ability to embrace object-oriented programming concepts that will be introduced throughout the term. That means that you will be starting this project, and then you will have to adjust your code as new programming concepts are introduced. This will upset you sometimes, but I can live with that. The clearer your code is, the more easily it will port to new structures that are introduced throughout the term. **Journalize** how upset you are. It makes for entertaining reading.

The multiple submission process will give us a chance to make sure your experience is as constructive as possible. My support notes are functional, but not great, that is why I consider this project to be as much a research project as a programming project. That leads me to the next section:

**Research and referencing your sources:**

Most programming is learning from other’s examples. I would expect most of you to be looking online for examples of how to approach your solutions. I also believe that copy/paste is sometimes more effective than re-typing what you see on the web. Just like in a history essay, you provide your teacher with sources of your quotations and numeric support, I will ask for the same. **When you embed any pieces of code from someone else, please comment the web site that helped you.** You will also have a list of sources on one page at the end of your project.

|  |  |  |
| --- | --- | --- |
| Weight | Description | Due Date |
| 10 | Status Report 1 | Friday, Feb 23 |
| 10 | Status Report 2 | Friday, March 9 |
| 10 | Status Report 3 | Friday, March 30 |
| 10 | Status Report 4 | Monday, April 16 |
| 10 | Status Report 5 – term 2 mark | Friday, May 4 |
| 10 | Status Report 6 | Monday, May 21 |
| 10 | Status Report 7 | Friday, June 8 |
| 50 | Final Package | Tues, June 20 |
| 120 | TOTAL |  |

You will be required to keep all your work in one folder. When it is handed in, I will be making sure your have followed through from your proposal all the way to your presentation.

**Project Journal :FreedCamp.com**

You will also be keeping a programming journal of the project using cloud-based project management software: FreedCamp. In this journal you will be entering your development of the program specifications, as well as frustrations with the process. You will be sharing any new programming techniques that you acquired while solving problems within your project. Some students try to “retrofit” their journal, spending the last day filling in a bunch of nonsense for me. Don’t. Anytime we have a “discussion” about the program specs, or when your program crashes for x days in a row – journalize it. When you reach a new level of your program that you are exceptionally proud of – journalize it.

FreedCamp: You will be creating tasks within the calendar. You will share the mandate for your task, as well as the GitHub repository link. Within each task, your team can add comments on the progress and frustrations. Most importantly, I will be adding comments to the task – giving direction or sharing confusion. I will assume that your team is still working on the task until it is marked as complete.

**Expectations:**

With your Status reports, I will make “suggestions” as to where you should go from there. I expect you to follow those “suggestions”, or give me strong rationale as to why you disagree, or show me code as to why you could not make my suggestions work. In a world of lots of copied code, modifying expectations is a great check on plagiarism. I expect you to copy some code from others, and incorporate it into your work – with proper sourcing. I also expect you to make modifications in accordance to my “suggestions” as a further way of showing you understand and can modify other’s code.

**Scratch**: There will be a folder for each scratch concept that you tested before you integrated it into your final project.

# **Status reports**

This will be your progress report of how your program has developed. Here, you will summarize your major breakthroughs, major challenges, and modified plans for the future. Within one folder, you will be **submitting your entire program folder** with your summary report, so that I can run your program and check your code. **You will also be submitting your journal as well as your manual.**

**Status Report**

List of programs. Clearly describe the problem that you are solving. Please put the date that you worked on it:

Major Challenges/setbacks( reference specific code please):

Source any web site/book that helped you with that concept:

Describe the code and the lesson that you learned from it:

With each status report, you will be submitting EVERYTHING. Organization is key. When I go to the groupwork folder**, I should see your project submitted in the following format:**

YourLastName: Under this folder will be the following folders:

**Documents**: It will hold all of your documents: journal, status report, list of sources, and all the other documents that will be submitted in your final project.

**Programs**: There will be a folder for each project.

**Status Report Marking Scheme**

|  |  |
| --- | --- |
| Mark | Description |
| 10 | Great progress.  Clear challenge.  Solid documentation and list of sources |
| 9 | Great work, but description, code, or documentation is lacking. |
| 8 | 2 of the 3 aspects are lacking |
| 7 | All 3 are lacking, or 1 is missing, or did not follow suggestions from previous report. |
| 6 | Very little progress shown, or did not follow suggestions from previous report. |
| 5 | Warning: this effort will NOT pass in final report |
| 4 | Little sign of work |
| 3 | Less sign of work |
| 2 | Project submitted with NO progress |
| 1 | Where is it??? I cannot give you a zero, even though you probably earned a zero. |
|  |  |

Major Programming Project Marking Sheet

|  |  |
| --- | --- |
| Topic: | Programmer: |

|  |  |  |  |
| --- | --- | --- | --- |
| **Criteria** | **Description** | **Marks** | **Out of** |
| Package  Clarity | The package was complete and in order.  All sections were clear, and easily understood. |  | 10 |
| Bugs, Specs and journal | All three were clear and insightful. |  | 10 |
| Program:  Challenging | The problem that you chose to solve proved to be challenging. |  | 15 |
| Program:  Clear front end | Your program interface proved to be clear to the end user: both for data input and program output/display. The **manual** was an excellent support. |  | 15 |
| Program:  Clear code | Your code was clear, concise, well-documented, and structured.  Clear **comments** throughout: especially to give an overview of a method, class, or frequently used variables. |  | 15 |
| Program:  techniques | You used the best programming techniques to solve the problem. |  | 15 |
| Lesson | You shared a new programming technique that was not in my notes. You clearly showed how it could be used, and why all high school programmers should be using it. |  | 10 |
| List of sources | Clear links as to where you learned and borrowed programming techniques. This will be an entire list of the sites that you used in order to make your program better. Clearly reference the parts of your code that benefited from these sites. Your code will have comments of your sources as well. |  | 5 |
| Notes to future programmers | These notes give great direction to future programmers as to where there is room for improvement, and possibly how to fix the problems (if only you had more time!). |  | 5 |

|  |  |
| --- | --- |
| **Criteria** | **Comments** |
| Package  Clarity |  |
| Bugs, Specs and journal |  |
| Program:  Challenging |  |
| Program:  Clear front end |  |
| Program:  Clear code |  |
| Program:  Techniques |  |
| Lesson |  |
| List of sources |  |
| Notes to future programmers |  |