

# CSC 1060 – Computer Science I (C++)

## *Final Project Requirements*

### OVERVIEW

You are required to complete a final project for your final exam. The project will consist of the design, development, test, debug, and presentation of a C++ program of your choice. This is a good opportunity to solve a problem of interest to you, to be creative, and to show your programming skills. The project can be anything you like, but it must have the required C++ programming concepts, listed below, which have been covered in this course.

After completing this final project, you will have a good introduction to the way larger programs are created. You will go through a lot of planning items before you code. You will realize that going through this planning process will allow your programming tasks to go smoother by minimizing bugs or errors in programming logic.

### INSTRUCTIONS

#### Getting started

Think of an interesting problem to solve using C++. For example, a game, a data management system, or an inventory management system. Students in the past have created things like a grocery shopping list, a car dealership program for their family business, or a program to handle inventory for a jewelry business they were starting. It can be anything you like. If you can't think of anything personal to you, please feel free to consult your instructor for ideas. You may also use a programming challenge from your textbook. You will of course have to enhance the application with additional features to satisfy the final project requirements. Once you have selected an idea, notify the instructor by submitting the final project you will be writing for approval.

#### Your Required Tasks in Order of Completion

1. Obtain project approval
2. Complete user stories
3. Complete actual program
4. Complete PowerPoint Presentation
5. Requirements Location Document

#### 1 - Project Approval

Prior to starting your project, you **MUST** obtain project approval. The purpose of the project approval is two-fold. First, project approval allows your instructor to review your project to assure that your program will allow you to demonstrate the required programming concepts for the final. Your instructor may offer suggestions to expand, contract, or change your program as needed. Often students either don't choose a program that has enough functionality, or one that has too much and will be difficult to complete in the allotted time period.

Secondly, your instructor may be able to provide you with some warning about completing your project. **Understand that plagiarism is a *SERIOUS* offense.** This final project has been a standard for this course for several semesters. Your instructor has access to all previously submitted final projects. **Your program must be written by you.** Although you may obtain ideas from the internet or other students, your code must be your own. There have been at least one or two students each semester that attempted to test the system. What you must concern yourself with is that you must make sure that no other student has turned in the exact same code you did. The program can be the same but the code must not. As each semester passes, trying to find a project to copy the code and submit it as your own becomes harder and harder.

**IMPORTANT:** Please understand that should plagiarism be discovered in your final program, all previous assignments for the final in which you may have previously received credit will also receive a **zero**. Should you feel the need to contest receiving a zero on your final, you must arrange an in-person meeting with the instructor within 1 week after final grades are posted.

To submit your project for approval, create a Microsoft Word document that contains your name, class, and the name of your program. Then submit a **one to two paragraph summary** about what your program does. This summary should be detailed enough so that your instructor can identify most of the programming concepts needed to complete the application. Think of this document as your 90 second elevator pitch that you would give someone you wanted to sell your program to.

**IMPORTANT:** When submitting this document to the Dropbox, please remember to at least include your name and a title for your program. Also, DO NOT DOUBLE-SPACE OR INDENT YOUR PARAGRAPHS!!

## 2 – User Stories:

With this final you will be required to create user stories to aid in the identification of the requirements for your application. Below is an example of how you would create these user stories. Note, that you should create a least 5 user stories for your application.

Suppose that a customer has requested a sales tracking system. The first step in an agile process (*a process you will learn more about in a later course*), like any other development method, would be to define the system requirements. The customer begins by meeting with programmers and providing user stories. A **user story** is a short, simple requirements definition. Programmers review user stories to determine the project's requirements, priorities, and scope. Here are three user story examples:

- *As the sales manager, I want to identify fast or slow-moving items so I can manage our inventory more effectively.*
- *As a store manager, I need enough lead time to replenish my stock so I don't run out of hot items.*
- *As a sales representative, I want to offer the best selection of fast selling items and clear out the old stock that is not moving.*

User stories do not deal with technical details and are so short that they are often written on index cards. You will list these user stories in a single Microsoft Word document. Each user story is given a priority by the customer, so the requirements can be ranked. In addition, programmers assign a score to each user story that indicates the estimated difficulty of implementation. What you will do for your final is number each of your user stories in the order of priority. This means that you will code each functionality in that order.

Note that it is not necessary that your final project contains all the functionality you have described in your user stories. However, your program **MUST** contain all the programming concepts required for the final. When writing user stories it's important to answer two questions:

- Who will be using the program (restaurant owner, customer, inventory employee, etc).
- How will the users be using the program? What will they be able to do?

Once you have completed your user stories, **please submit that document in the User Stories Dropbox prior to the due date listed in the Dropbox**. Your instructor will then review your user stories and make updates if needed to assure that your project has all the programming elements covered in this course.

### 3 - Program Requirements:

Use the following categories to write the program.

- **Think of the User** - specify the problem, including somewhat detailed requirements to make the program user friendly.
- **Design the program** - Sketch the User Interface. Design the program - determine the classes, fields, methods, objects, etc.
- **Programmer** – Write pseudocode for all methods and then write the code.
- **Tester** - Develop a test plan including test procedures & test data.

The program should be interactive with the user. The code must include the following:

1. Definition of at least **one class**.
2. Instantiation and use of at least **one object** of your class.
3. Pointers **MUST** be used throughout the application;
4. Only include **main logic in your main function** and use functions for reusable code
5. Use of a **loop** of your choice
6. Use of a **decision structure**.
7. Your application must be driven by data in a **data file**. This includes reading from a file as well as writing to a file.
8. Use of at least one **dynamically allocated array**
9. Definition of an **enumerator**
10. Declaring, initializing and use of an **enumerator variable**
11. Definition of a **structure**
12. Declaring, initializing and use of a **structure variable**
13. Use of at least one **function** in your main program. This function **MUST** have a parameter list and return a value. (You may choose to accomplish this in multiple functions.)
14. **Declaration** of at least one **pointer variable**
15. Use of the **Address of Operator** of a pointer variable
16. Use of the **Dereferencing Operator** of a pointer variable
17. Use of a **dynamic variable**
18. Include **comments** in the code

**IMPORTANT:** Because your final project will contain multiple files, it is important to complete the **Requirements Location document**. This document is attached to the Final Project and Presentation Dropbox and **MUST** be submitted with your final.

#### 4 - PowerPoint Presentation

As the final step, a PowerPoint presentation MUST be submitted with your program. The presentation must accomplish the following:

- Provide an introduction to the programmer and the program. Be sure to provide a name for your application and give a summary of what it does.
- Introduce the problem that the program solves and how it solves it.
- Explain the major parts of your code and how it works. For example, detailing the classes and how objects are handle in your program. It is best to provide screenshots of your code here.
- Provide a demonstration of your program. Screenshots with explanations will accomplish this task.

#### SUBMITTING YOUR FINAL

When submitting all of your final project documents, you must use the following naming convention for all of your files. You start with the course number, then put your first initial and last name followed by the document name. For example, **CSC160\_KHoward\_Flowchart** or **CSC160\_KHoward\_Final Program**.

Failure to use this naming convention could result in points being deducted from your final. Also, when submitting your final NetBeans Project, please submit the **ENTIRE** NetBeans Folder as a zip file. To do so, following these instructions:

1. When your program is complete, save and close your solution.
2. Locate the solution folder on your computer. Do not open that folder.
3. Right click on the solution folder and click **SEND TO** then **COMPRESSED (ZIPPED) FOLDER**.
4. Submit the compressed zipped folder.

#### GETTING HELP

It is important for you to understand that you are not in this alone. Myself and the class will be there to help you complete this project. Please feel free to email me or post questions in the general discussion area. If there is something that you would like me to review to make sure you are on the right track, feel free to send it to me ahead of time. You may send stuff to me for review as many times as you want. However, it is important that you provide me with SPECIFIC questions you want answered. I will not use these opportunities to grade your assignment before the due date. Understand that it is easier and quicker for me to grade these projects when most, if not all, of the submission materials are correct.

#### FINAL GRADING

As mentioned previously, you have a number of documents to submit for your final. Below is a table that detail the grading scale.

Task / Grade Item	Due Date	Points
Project Approval	4/3/2023	10
User Stories	4/10/2023	10
Power Point Presentation	5/1/2023	20
Final Program	5/1/2023	60

