ECE3220 – Software Design in C & C++
Spring 2023
Project Description

Step 1:

Provide the information below using the template that is shared with you.

- Form your Team (2-3 people), including their information.
- Write an abstract for your project, indicating what your project will do (max. 2 pages).
- Provide your GitHub repository link.
- List the features/functionalities (with bullets) of your project. You can include the features of your project after the abstract.

If your project is private, please give us access to your GitHub repository, to the GitHub usernames below:

- ekincanufuktepe
- gbenga007

Send your first draft on Canvas, by 4/7/2023. There is no hard deadline for the project description and features, but the sooner you submit your draft, the sooner we can give you feedback to adjust your project features. Even if your project is not description and features are not determined yet, you still have to at least submit your project team information by **4/7/2023, 11:59PM (this is a hard deadline)**. If we do not receive any team information, we will assume that you will do your project alone.

Step 2:

- Start designing project.
- Distribute tasks in your team.
- Communication is key to finish your project if you are stuck and cannot move forward ask help from your team members. You can submit issues in your project if you would like.
- Make sure that you have test cases for your project as well. Writing test cases is not all about submitting invalid inputs (that is called negative testing). You want to first make sure that your project is working as it is expected to be with valid inputs (positive testing). Valid inputs should be selected carefully (we will discuss the details of testing in the class).

Grading Criteria

- Following good programming practices
- Well commented code
- Low coupling, and high cohesion
- Free of memory leaks, dangling pointers, and double frees
- Free of unused or unreachable codes
- Test cases, code coverage above 85%
- No long functions size
- Following SOLID principles, usage of design patterns.

- A working project
- Report (Writing README.md on your Github repo., see “Github Repository Page” section)
- Project demo and presentation
- Well written and organized commits. Explain the change/modification clearly in the commit. For example, why was the change required, what did you change?
- Individual contribution (commits will be reviewed)
- Try to include at least 2-3 design patterns, but do not include it just to say you used a design pattern. Design patterns must be used at the right place!
- **BONUS (+20 Pts):**
  You can only do one of them. Also, NO, you will not get +40 pts if you do both.
  1. If your project is not a GUI-based project, setting up a Docker container for your project. However, you should include your docker container repository in your GitHub repository page.
  2. Implementing a GUI-based project.

Rules & Recommendations

- While evaluating and grading your projects, we will be using a web-based plagiarism tool to detect if you have got the code from somewhere else. Any code we detect that has been obtained from another repository, the overall Project will be graded as 0.
- The project demo will be on **May 3rd, 2023,** during class hours. Some might do their presentation on **May 1st, 2023** as well.
- The deadline for the project is, **May 3rd, 2023, 11:59pm**. You are also required to upload your projects as a zip file on Canvas for official grading. You can simply use the “Code → Download Zip” feature from GitHub and upload it on Canvas. While submitting your projects, please share your GitHub repository link as well. One of the team members would be enough to do this submission.
- Even though we will be reviewing your git commits for checking who contributed to the project, everyone should submit a text file that grades their group members (except themselves). Please write a short description on how your group member contributed to the project that supports your grade. The grade should be between 0-100.

GitHub Repository Page

- Use your README.md file.
  - Explain your project.
  - Include your team members.
  - Include the dependencies that is required for your project. What are the steps for the setup, what environment is needed? Don’t forget to include the versions of the dependencies.
  - How to use and run your project.

Some Example Projects in the Past

Please note that these projects do not represent the lower or upper expectations from your projects.

- Personal Finance Project: https://github.com/dcjn7z/3220PersonalFinanceProject
- Dungeon & Dragons: https://github.com/ScientiaDominus/ECE-3220-Project

- Amp V2: https://github.com/Masonrf/Amp-V2
- NPC game: https://github.com/Treeja/ECE3220Project