

**Recep Tayyip Erdogan University**

**Faculty of Engineering and Architecture**

**Computer Engineering**

CE205- Data Structure

**Homework-3 (Week-6)**

**Fall Semester, 2021-2022**

|  |  |
| --- | --- |
| Instructor | Asst. Prof. Dr. Uğur CORUH |
| Contact Information | [ugur.coruh@erdogan.edu.tr](mailto:ugur.coruh@erdogan.edu.tr) |
| Google Classroom Code | **y3qryym** |
| Publish Date | **14.11.2021** |
| Due Date | **25.11.2021 12:15-13:15** |

**Complete the following homework requirements, prepare them in the format given in the description below until the deadline and time, and upload them to the classroom's related assignment.**

**Grades:**

|  |  |
| --- | --- |
| Problem-1 | 100 points |
| **Total** | **100** points |

***Development Environment***

In this problem you will develop a C# application that provides visual solutions for graph problems.

I have prepared a sample implementation for MST algorithm in this template.

<https://github.com/ucoruh/CE205-HW3-Template>

You can remove git remotes from this downloaded repo with

git remote **remove** origin

and then add your private GitHub repo

git remote **add <url>**

after this operation push your project to GitHub repo and add me contributor for this project.

You should provide working code without any bugs.

Be careful about project types. The project type will be the library. If you use the template, you will have already generated projects for you

Do not forget to provide README.md files with clear descriptions

Clearly explain your codes with comments and generate Doxygen documentation.

***Code Sharing***

You will create your project with git version management support, and you will add a gitignore file for your project to decline unnecessary files. You will push your lite code to your GitHub account, and you will share this link in the classroom. Also, you will submit your compressed code to the classroom with the report. The report is necessary for formal validations.

Be careful to submit codes without unnecessary files such as intermediate build files, executable outputs, etc. these files are ignored via virus protection tools.

***Grading Criteria***

1. *Github Code Sharing*
2. *Classroom Code and Report Sharing*
3. *Code Comments and Doxygen Code Generation*
4. *Project Types*
5. *Code or Project Bugs (Not Running or Complaining)*
6. *Unit Tests and Assertions*
7. *Algorithm Solution Methods and Explanations*

**Problem-1: (100 points)**

In this problem you will select two algorithms from the list

* BFS (50 points)
* DFS (50 points)
* SCC (50 points)
* Topological Order (50 points)

And you will implement visual algorithms in given template. You can use MST sample usage.

There is a sample random graph generation also provided via sample. There should be three feature

* Generate Random Graph
* Clear or Show Other Edges
* Reset Graph
* Solve

Step by step solution is optional such as previous or next step

Good Luck!