

# Assignment02

## Object Oriented Programming

Submission deadline: 8<sup>th</sup> March, 2023 – 11.59 PM

### Submission Guidelines:

- You need to only submit your source file on **google classroom within deadline**.
- Late submissions upto **24 hours** are allowed with a penalty of **20%** on google classroom. After that **no** submission will be entertained
- Name the source file as “assign01-roll#.cpp”. For example **assign01-i200561.cpp**
- Submit only .cpp file
- **Strong plagiarism check will be executed, while checking your assignments. The assignment is individual so don't share your code with anyone.**
- **Group assignment** of two students is also allowed, but the group members will only be able to get max 80% marks each.
- If plagiarism is found, source will also get zero, and a warning.

## Student Information System v2

In this assignment, you will build upon the already implemented basic version of the “**Student Information System**”. The same text file is provided with this assignment as well, which contains the students' data.

- Now you will need to make a Student structure to store a student's data
- You need to make an appropriate size dynamically allocated **array** of type struct student, where you will store all the information from the file into this array.

Apart from the already implemented menu, you will also implement the additional menu given below **in red font**.

1. Show all students
2. Search a student by Rollno  
Take input a roll number, and try searching it. If found, print all the information associated with that student
3. Find a student with maximum CGPA  
There can be more than one students with same max CGPA. You may print only one of them, or print all of those students
4. **Sort the Student's data**
  - a. **Sort w.r.t. Rollno**
  - b. **Sort w.r.t. Name**
  - c. **Sort w.r.t. cgpa**

5. Add new student (this is now NOT a bonus rather it's a necessary task )

To add a new student, take input all the information of a student, and then store that at the end of the Student.txt file.

### Time Stamping

Whenever you add new data to the file, date and time of that moment needs to be recorded in another file "Activity.txt", which should just mention the data and time in a line.

### Bonus Task

In the bonus task, you need to restrict the use of this program to only the registered users. The registered users' data will be in a file "Registered.txt" which contains the username and password for each registered user. (create this file yourself)

You need to authenticate the user before he is able to use this software.

If the user provided username is not in the "Registered.txt" file, then prompt that no such username exists.

If the user enters correct username, but wrong password, then ask him to enter correct password

After **successful** login, the program will show the normal menu as discussed above. But while doing the timestamping the name of the user also needs to be recorded in the Activity.txt alongwith the date and time.