**Object Oriented Programming   
& Data Structures**

**Programming Assignment Instruction**

**To Teaching Assistants: Please check that the information at the top bar is correct.**

**You must use the assignment template to implement your programs**.

**You do not receive any score points if you do not use the template.**

In this assignment, you are going to implement some classes which are integrated in a large system. You can see the files in 00\_StudentWork. These classes are:

* GRAPH\_SYSTEM
* MY\_2048
* IMAGE\_SYSTEM and IMAGE\_NODE

Please read the instructions for implementing these classes. **Write your programs in Visual Studio 2019 on the .NET platform. The compiler version must be v142.**

Open the project: SOGLFramework.sln. **We will rebuild your program in the Release mode and check your program.**

**How to run the program? You can find the executable file SelfMotivatedSystem.exe**

**in ./bin/Release**

**To run the program**: 1) go the folder ./bin/release/. 2) Then execute SelfMotivatedSystem.exe.

**Do not press F5 to run the program. We do not set the file directories in the project. Go to the folder and click** SelfMotivatedSystem.exe to run it.

**The demo program may have bugs.** These bugs are useful for you to understand that if we do not do a good job to check our programs thoroughly, all these bugs cannot be fixed. However, as you can see, you can still run the programs without a severe runtime error. **Thus, please follow the instructions. If the instructions are not clear to you, send us an email to clarify the issues. Could you find any bugs in the demo program? Do you know how to fix the bugs?**

**Submission**:

1. Change the folder name to ID\_Name, where ID is your student ID and Name is your name. Zip and then upload the entire folder of the project (must contain source code) to E3 platform before the deadline.
2. You must demonstrate your work to our TAs in the lab session.
3. If you cannot demonstrate your program to our TAs, your score is zero.

**Penalties**

1. Late submission: 40% penalty each day.
2. Cheating: you will be received a score of zero. For example, borrowing your source code to others or/and copying others’ source code.

**You can add new functions but you must not add new files. Deduct 50pt if you add new files.**

**Requirement Specification**

1. **Basic tasks.**
2. Write your name, student ID and email address in the header file mySystemApp.h
3. Press ‘s’ or ‘S’ to show your student information: **date**, student ID, name and email address. showStudentInfo\_2022( ) in mySystemApp.cpp
4. Set your name and ID for cn\_StudentInfo in mySystemApp\_HandleEvents.cpp
5. Press F5 to show student information at the top bar of the window.

**Items I, II, III and IV must be done. If not, your score is zero.**

Key usages:

F1: perform Graph System

F2: perform My\_2048 with dimension 4x4.

Press the spacebar to toggle the autoplay mode. Obtain at least one game with 512 in FOUR attempts. It must take less than 30 seconds for each attempt on a computer with a 2.4GHz CPU.

Press ‘a’ or ‘A’ to perform an action automatically for one step.

F3: perform My\_2048 with dimension 8x8

F4: perform Image System

F5: show your student information. Set **cn\_StudentInfo** properly**.**

i, I: ask for input (to see other key usages in the current system)

s, S: show the student information

1. **System tasks.**

[40%] Implement GRAPH\_SYSTEM. You can see the details in the header and source files (mySystem\_GraphSystem).

[30%] Implement MY\_2048. You can see the details in the header and source files (mySystem\_2048).

[30%] Implement the image system. You can see the details in the header and source files (mySystem\_ImageEditor.h, mySystem\_ImageEditor.cpp, mySystem\_ImageEditor\_draw.cpp).