**Object Oriented Programming   
& Data Structures**

**Programming Assignment Instruction**

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

**Table of Content**

1. Instruction
2. Requirement Specification
3. Submission information
4. Penalties
5. Assignment template conversion from Visual Studio 2010 to Visual Studio 2019

**The assignment template is written in VS 2010. Let’s convert it to VS 2019!!!**

1. Instruction for the subsystem: mySystem\_MonteCarlo
2. Instruction for the subsystem: mySystem\_SineCosineFunction
3. Instruction for the subsystem: mySystem\_CubicFunction
4. Instruction for the subsystem: StudentManager

**1. Instruction**

**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 the folder **00\_StudentWork**. These classes are:

* mySystem\_MonteCarlo
* mySystem\_SineCosineFunction
* mySystem\_CubicFunction
* mySystem\_StudentManager

**Write your programs in Visual Studio 2019 on the .NET platform.**

**The compiler version must be v142.**

**We will rebuild your program in the Release mode and check your program.**

**How to run the program? You can find the executable file enjoy\_programming.exe**

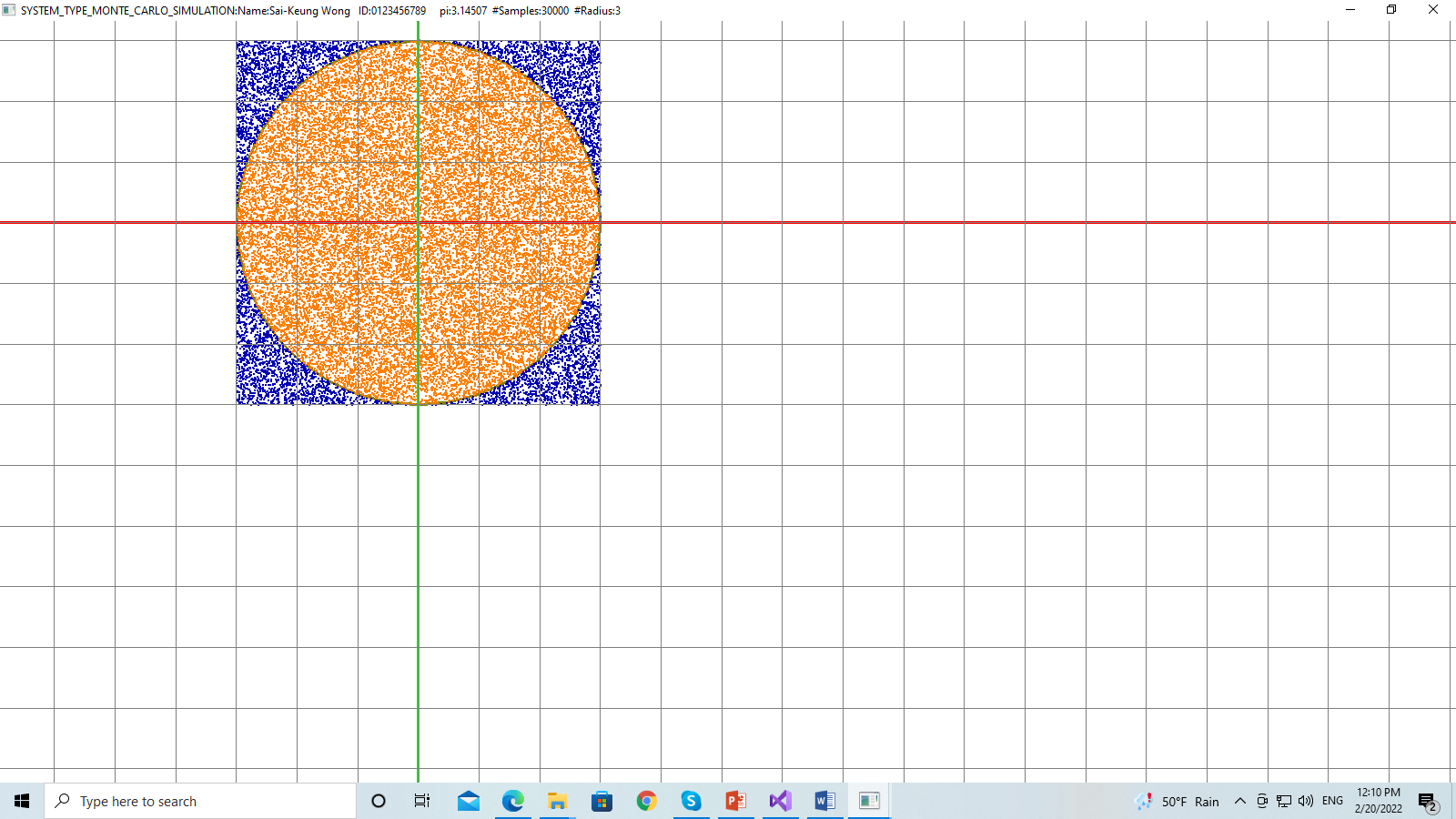
**in ./bin/Release**

**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?**

**2. Requirement Specification**

Use double to define a variable which is a floating point number. All the calculations should be done in double precision. Don’t use float. Show the value of a floating point number up to 8 decimal digits.

1. **Basic tasks.**
2. **Write your name in the header file mySystemApp.h**
3. **Set the macro STUDENT\_INFO in mySystem.h correctly. So that the top bar of the window shows the subsystem name, your name and student ID, as shown in the following figure.**



**Figure 1.1: Snapshot of the subsystem MonteCarloSimulation. The top bar shows the subsystem name, student name, ID, estimated pi, #samples, and radius.**

1. Press ‘s’ or ‘S’ to show your student information: **date**, student ID, name, and email address. showMyStudentInfo( ) in mySystemApp.cpp

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

**Key usages**

F1: perform Monte Carlo Simulation

F2: perform Sine-Cosine Function Calculation

F3: perform Cubic Function Calculation

F4: perform the student record management

i, I: ask for input

s, S: show the student information

**3. Submission**:

1. Change the folder name to ID\_Name, where ID is your student ID and Name is your name. Upload the entire folder of the source code to E3 platform before the deadline.
2. You must demo your work to our TAs in the lab session.
3. **If you cannot demo your programs, your score is zero.**

**4. Penalties**

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

**5. Assignment Template conversion from Visual Studio 2010 to Visual Studio 2019**

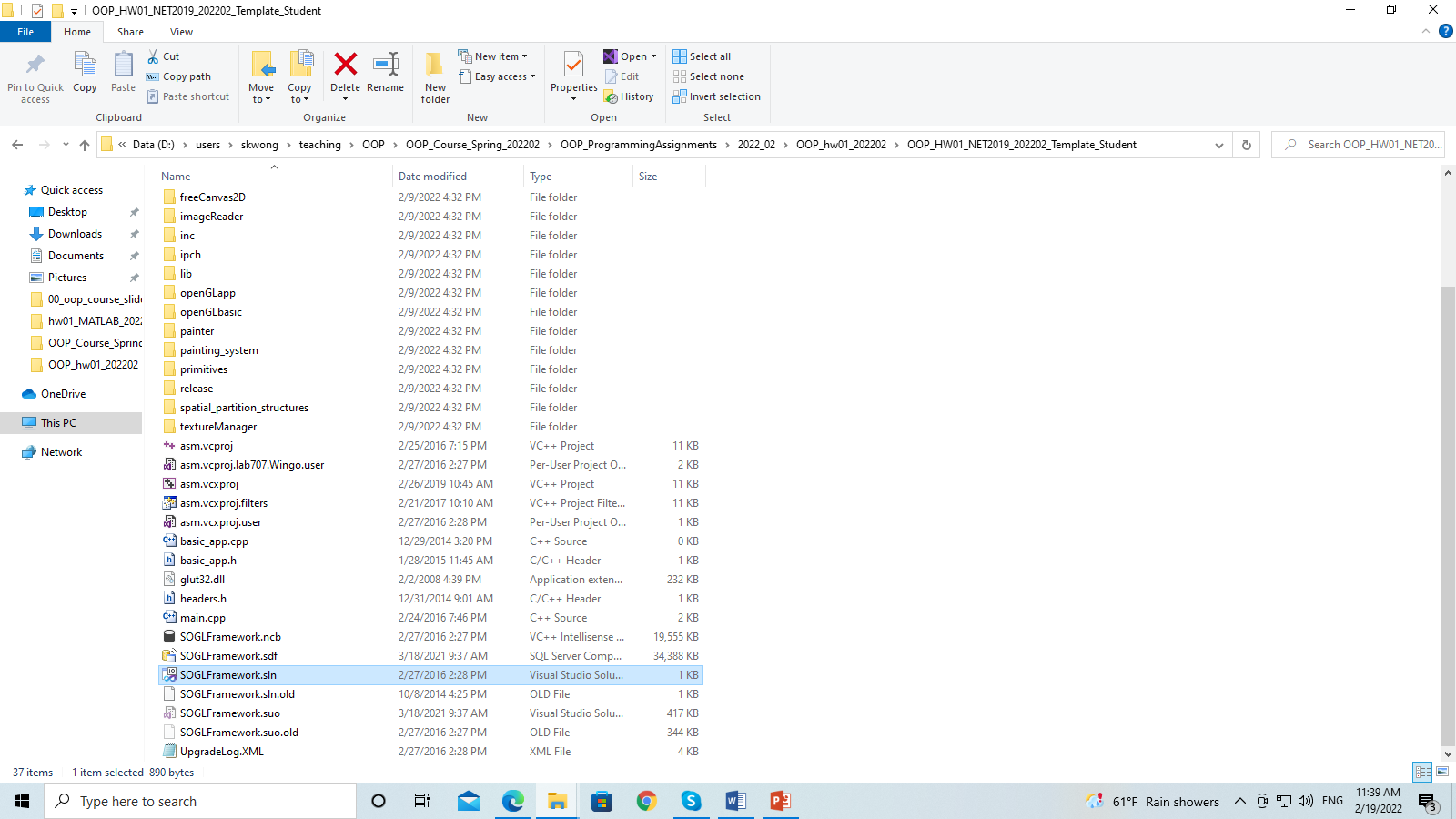
We may encounter that some programs are implemented on some old platforms. How do we use the new platforms to compile and build such programs. This exercise is important for you to understand that if the programs are well written, we can easily convert them so that we can use a new platform to handle the programs.

Follow the steps to convert the assignment template written on Visual Studio 2010 to Visual Studio 2019 and use **Platform Tool** **v142**.

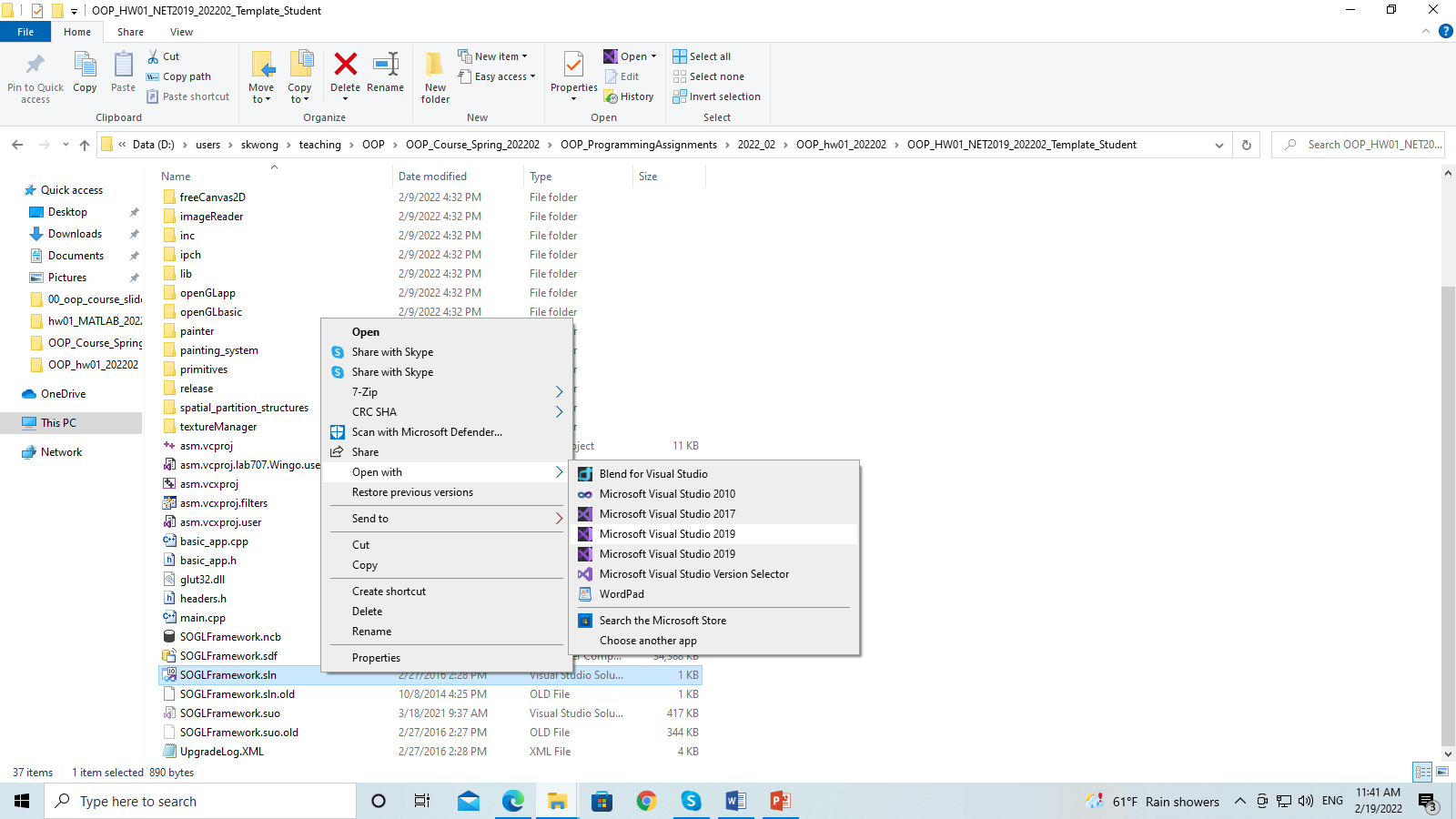
You may encounter different situations when you convert the assignment template. Here, we show two situations that you may encounter: (1) set the version to be upgraded; (2) set the compiler version manually in Visual Studio.

**5.1 Situation One: In this case, Visual Studio will ask you to upgrade the platform to the newest version and compiler version to v142.**

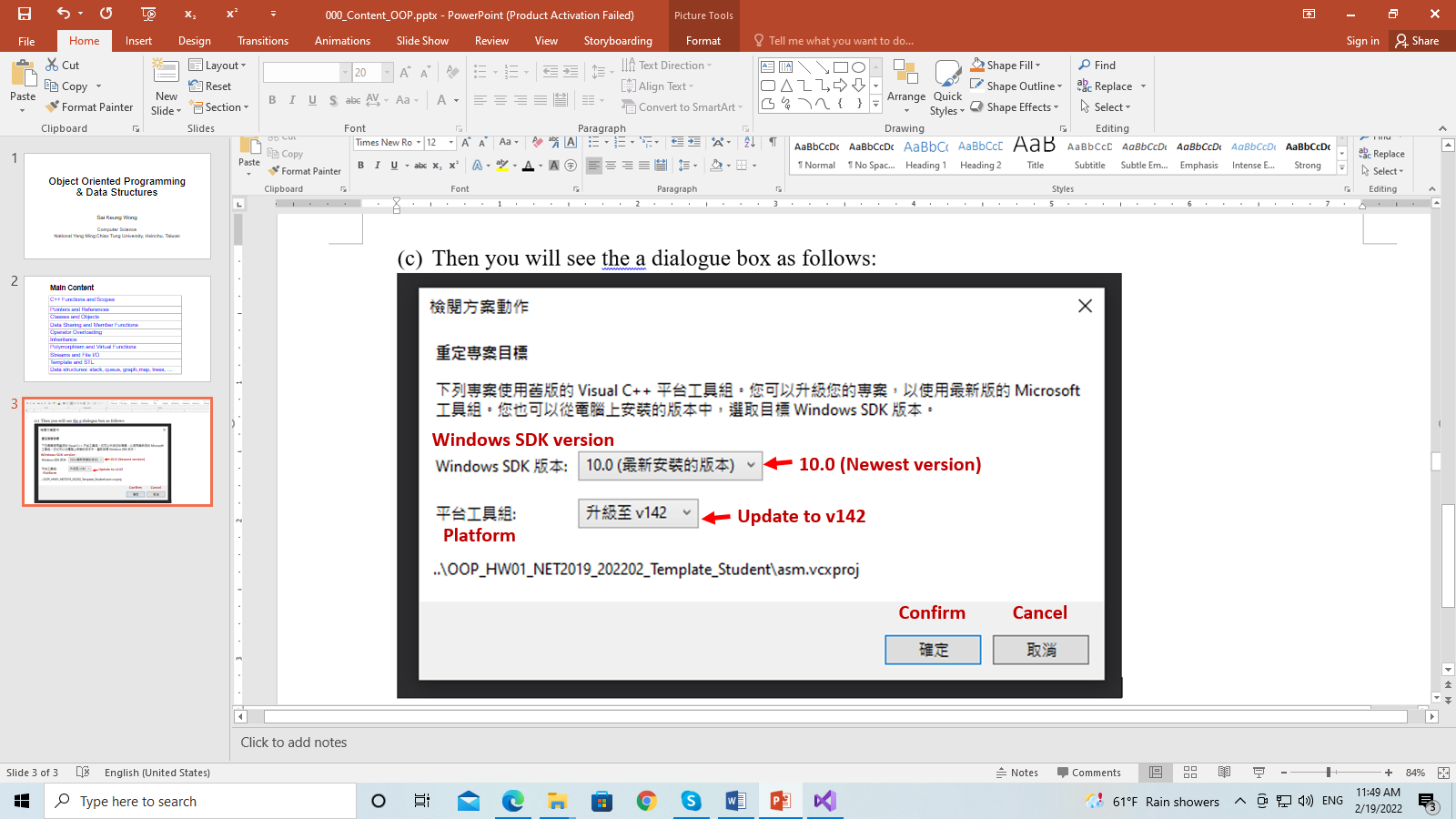
1. In our case, we need to find out the main project file which is SOGLFramework.sln



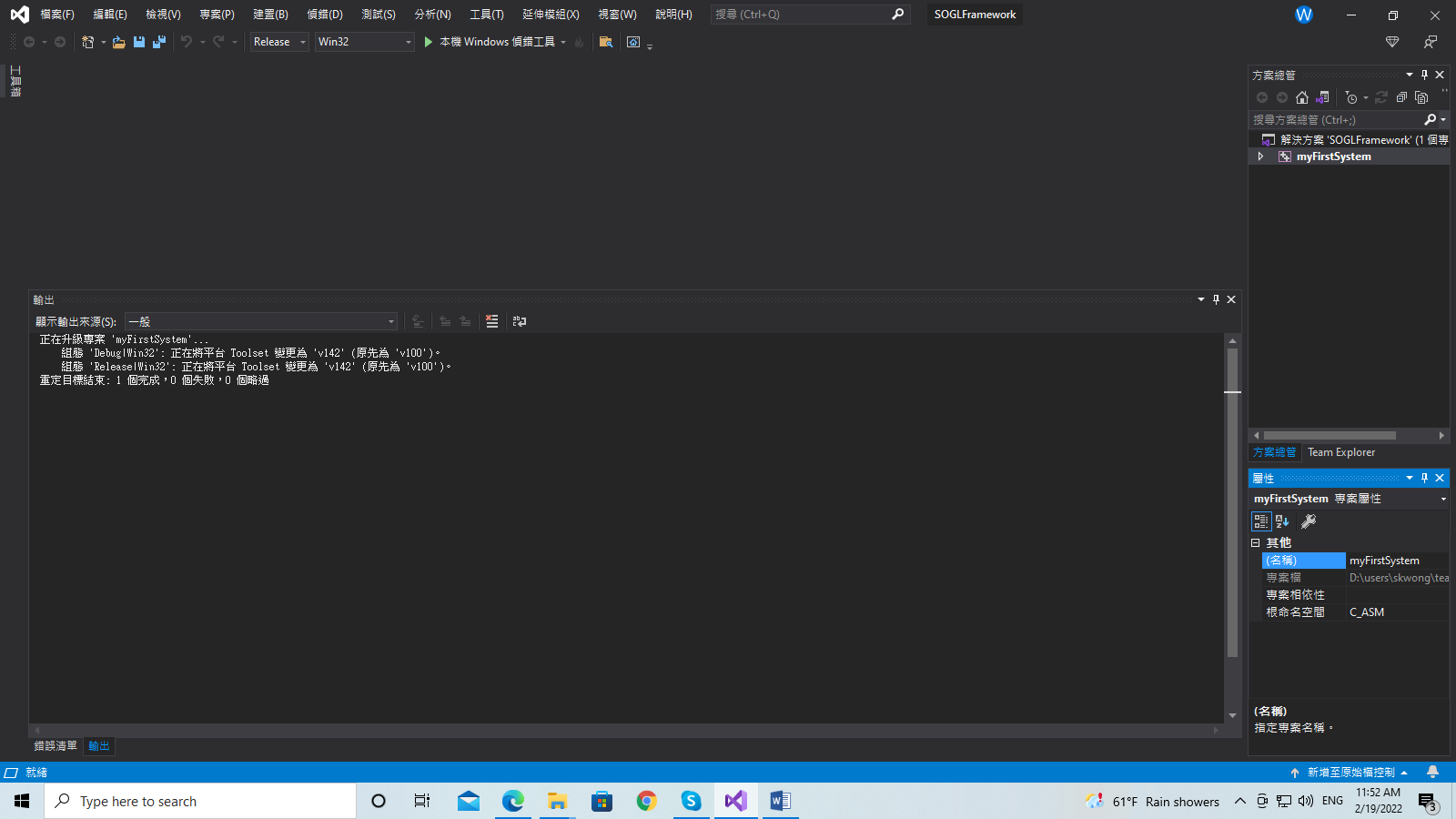
1. Use the mouse and right click on the project file SOGLFramework.sln and then select “Open with VS 2019”



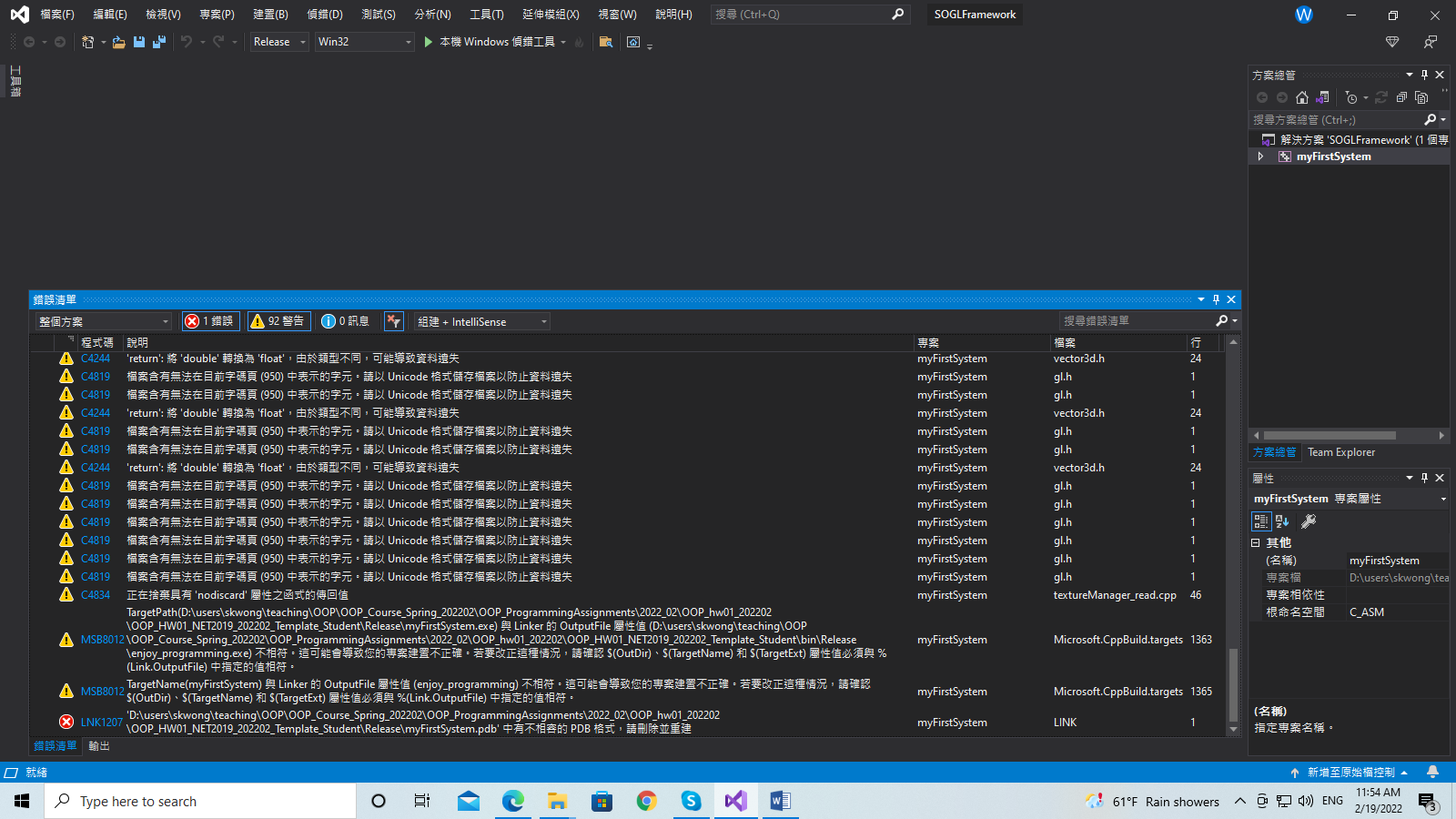
1. Then you will see a dialogue box as follows:

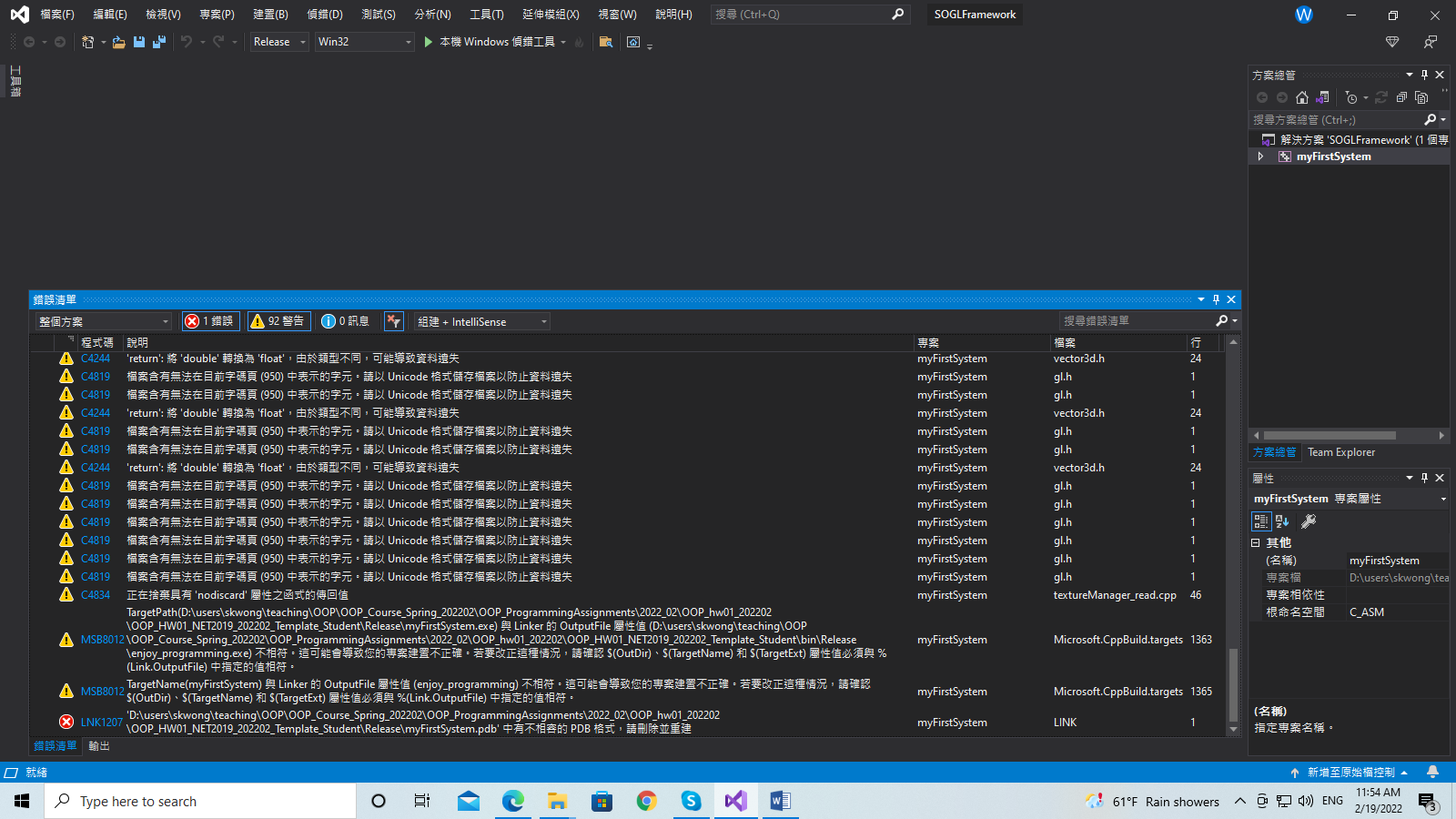


1. Make sure that you set the compiler version to v142. Then press Confirm. Then you will the following window



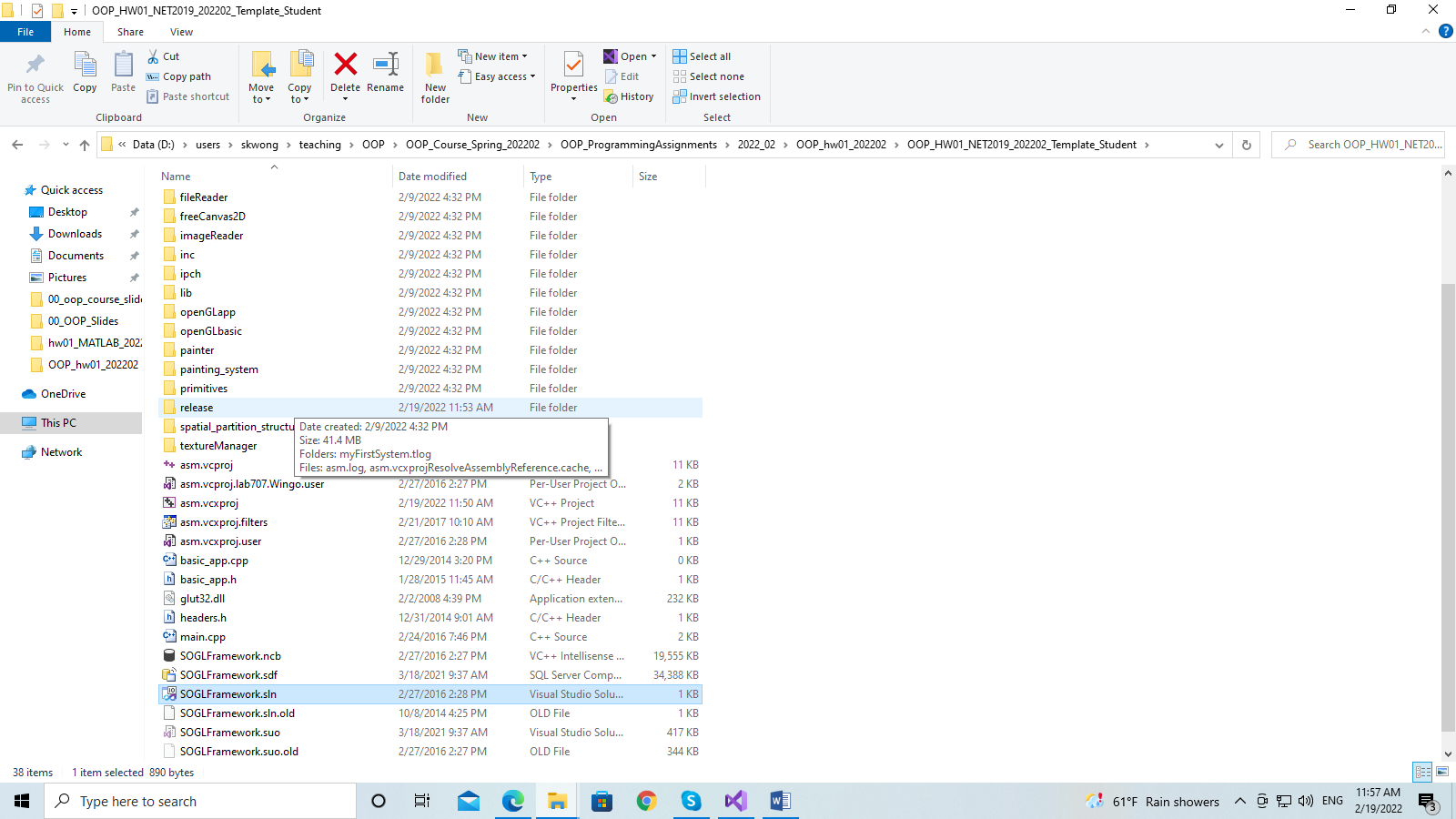
1. Now press ALT + B + B to build the program. As the template program is quite large, it will take a while to build the program. Then you will see the following messages. There is a severe error.

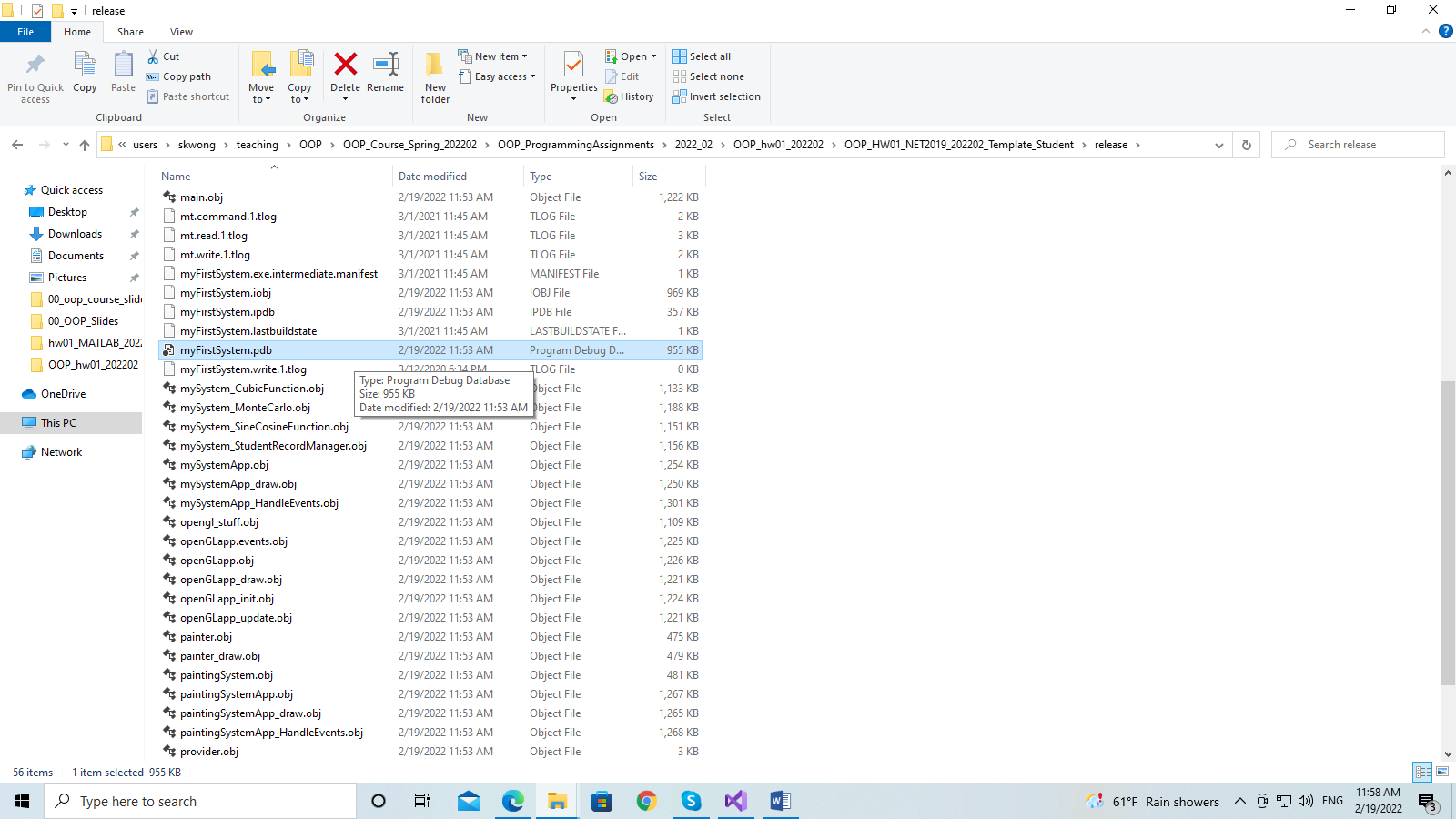




The last message tells you that we need to delete the file: **myFirstSystem.pdb** in the folder .\Release

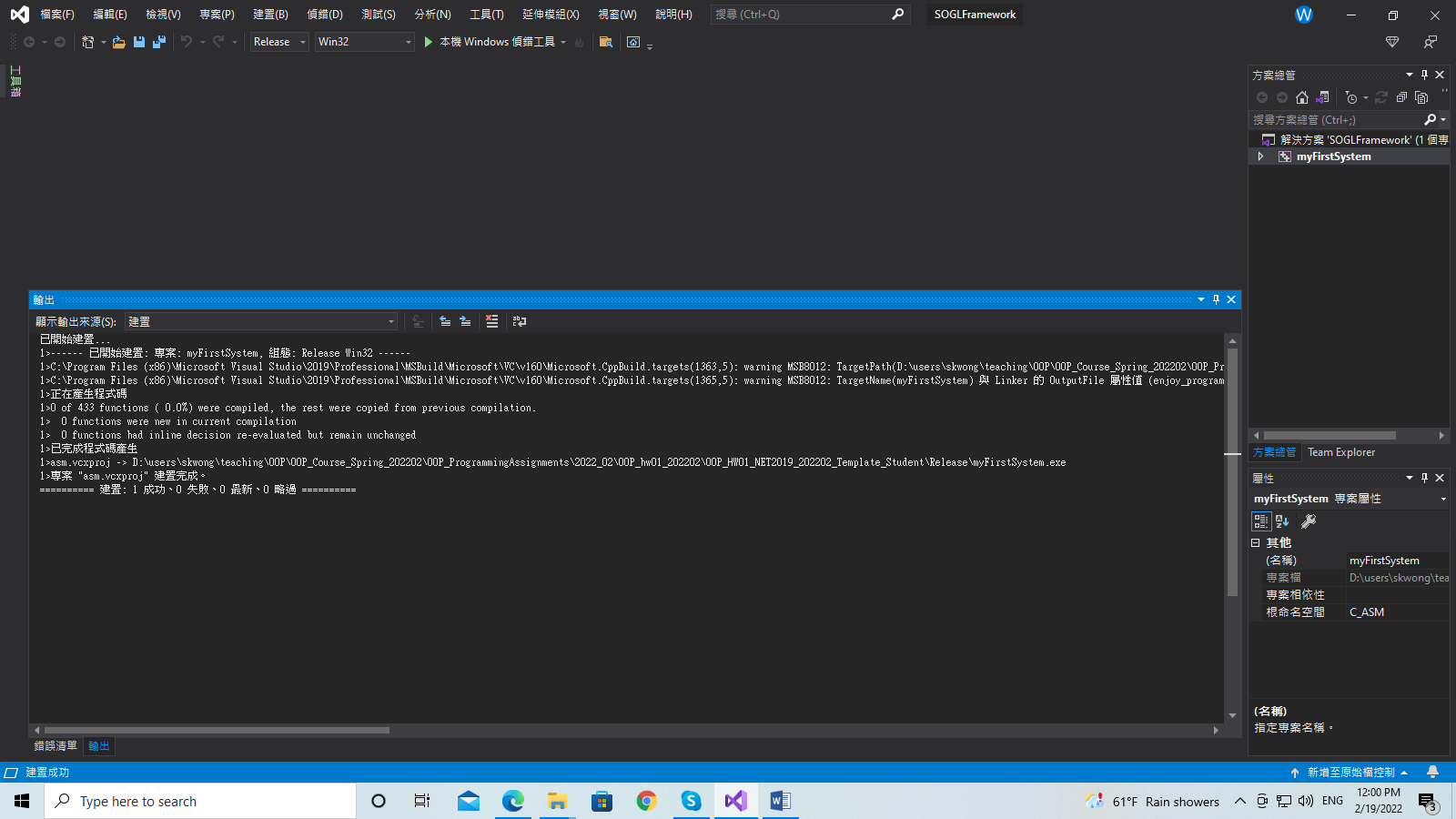
Go to the folder and delete the file.





Delete the file: myFirstSystem.pdb.

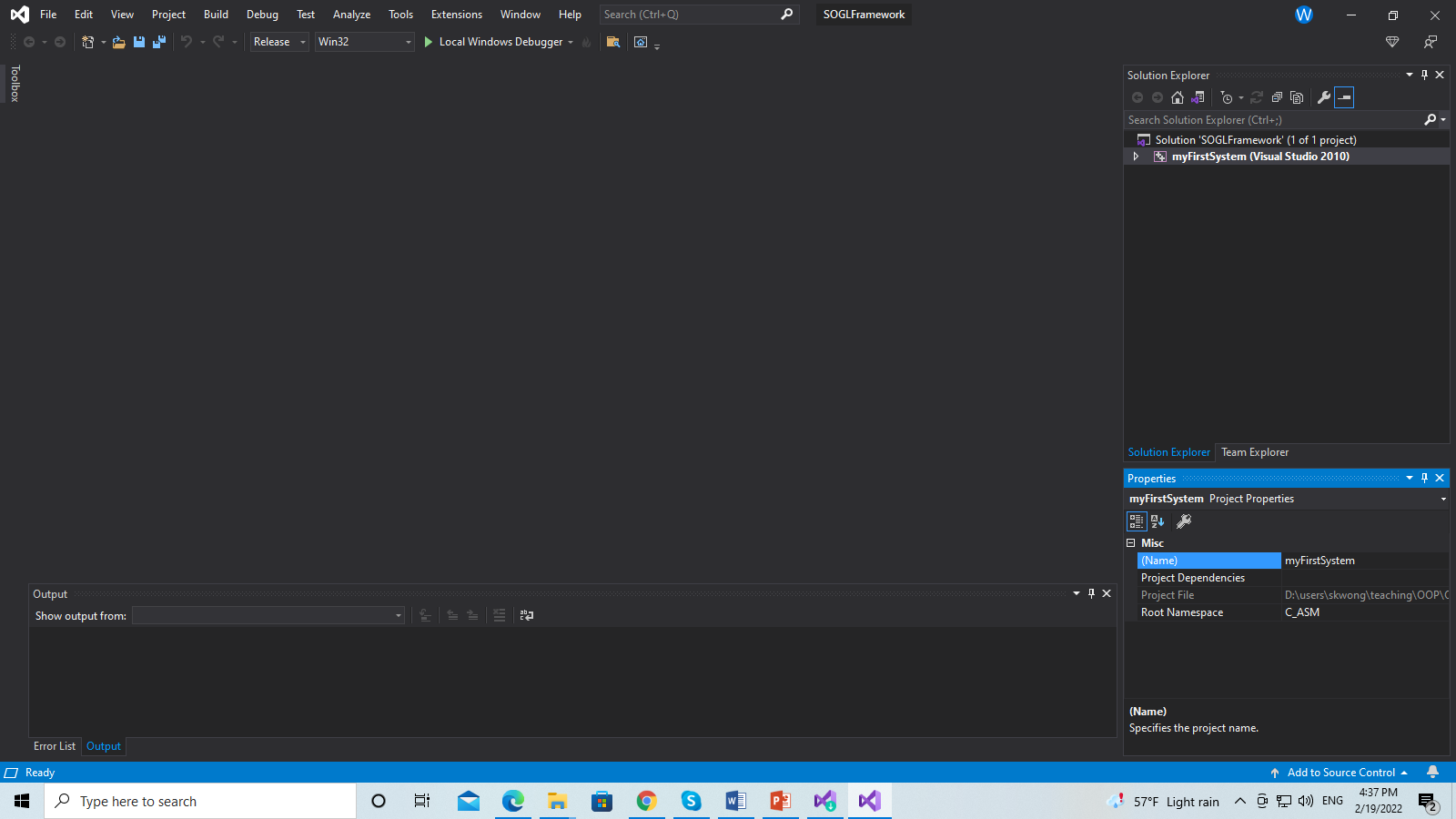
After that, go back to Visual Studio and build the program again, by pressing ALT+B+B. We are done.



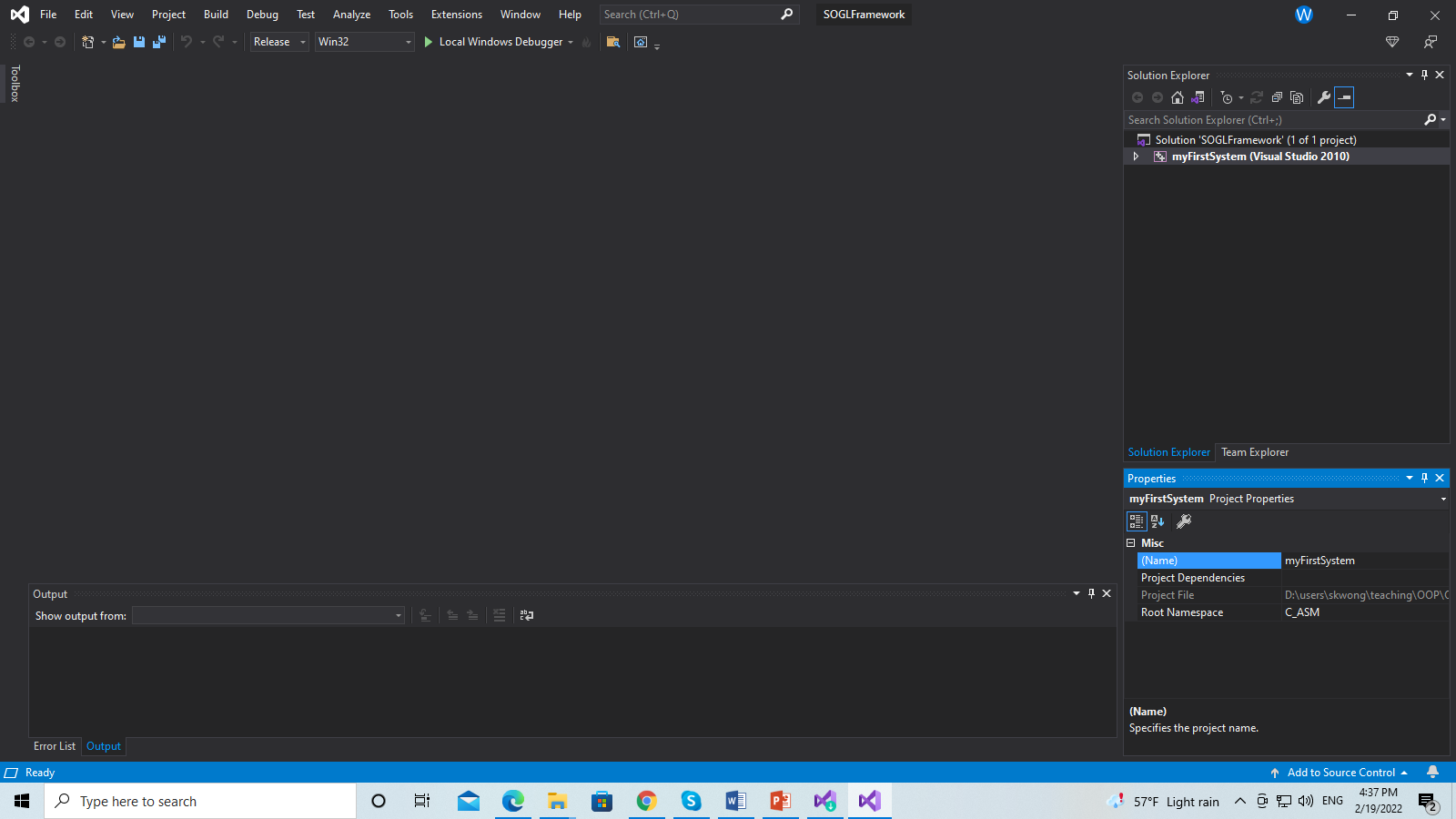
The message says that the program is built successfully. Well done!

So you see how easy it is to convert a program written on an old platform to a new platform. However, in real life if the programs are not “well-written”, we will encounter a lot of problems. We may need to rewrite a large portion of such a program. Luckily, we can convert our assignment template so easily in our case.

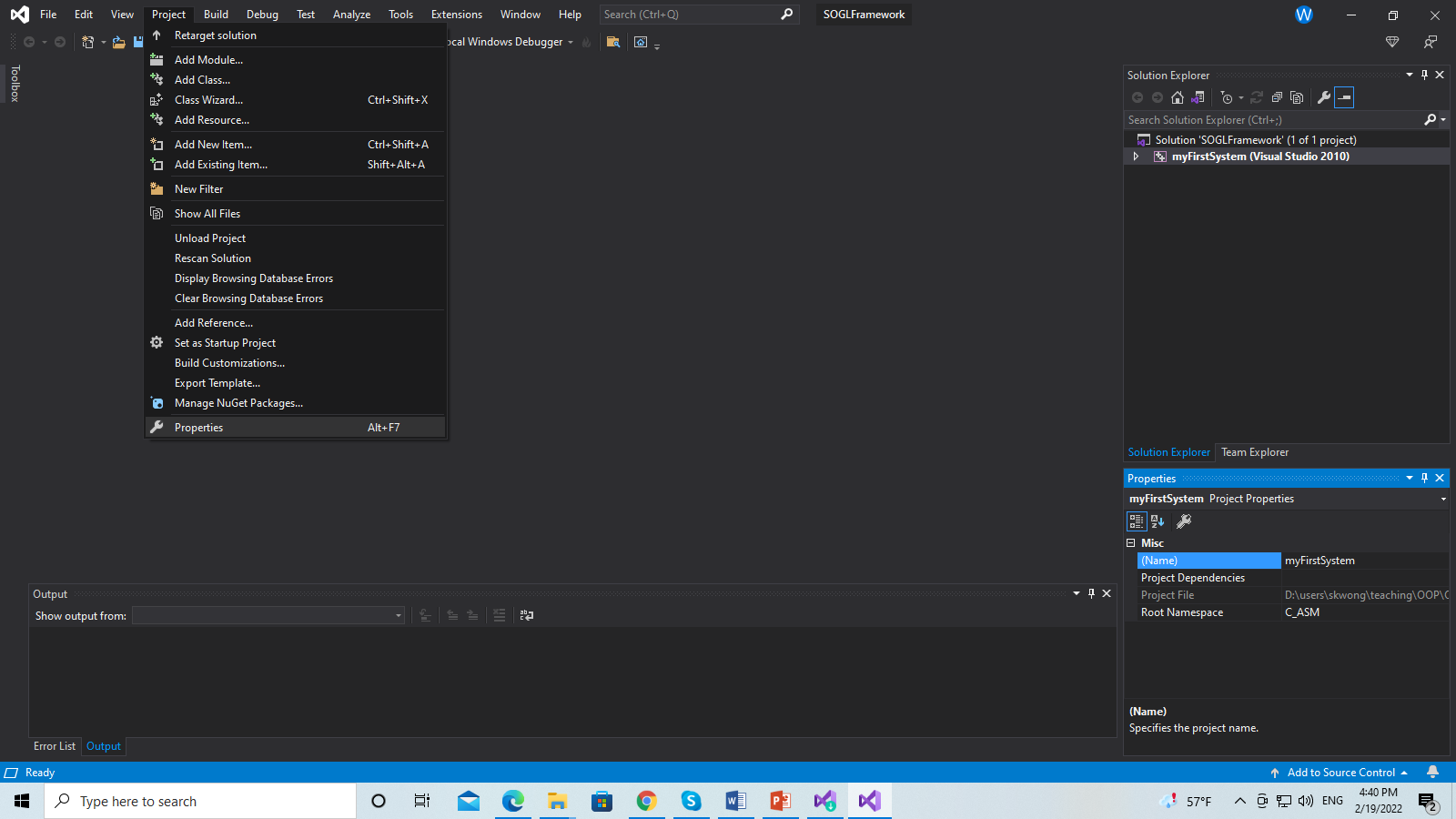
**5.2 Situation Two: In this case, you open the project and enter Visual Studio. You need to set the compiler version to v142 manually.**



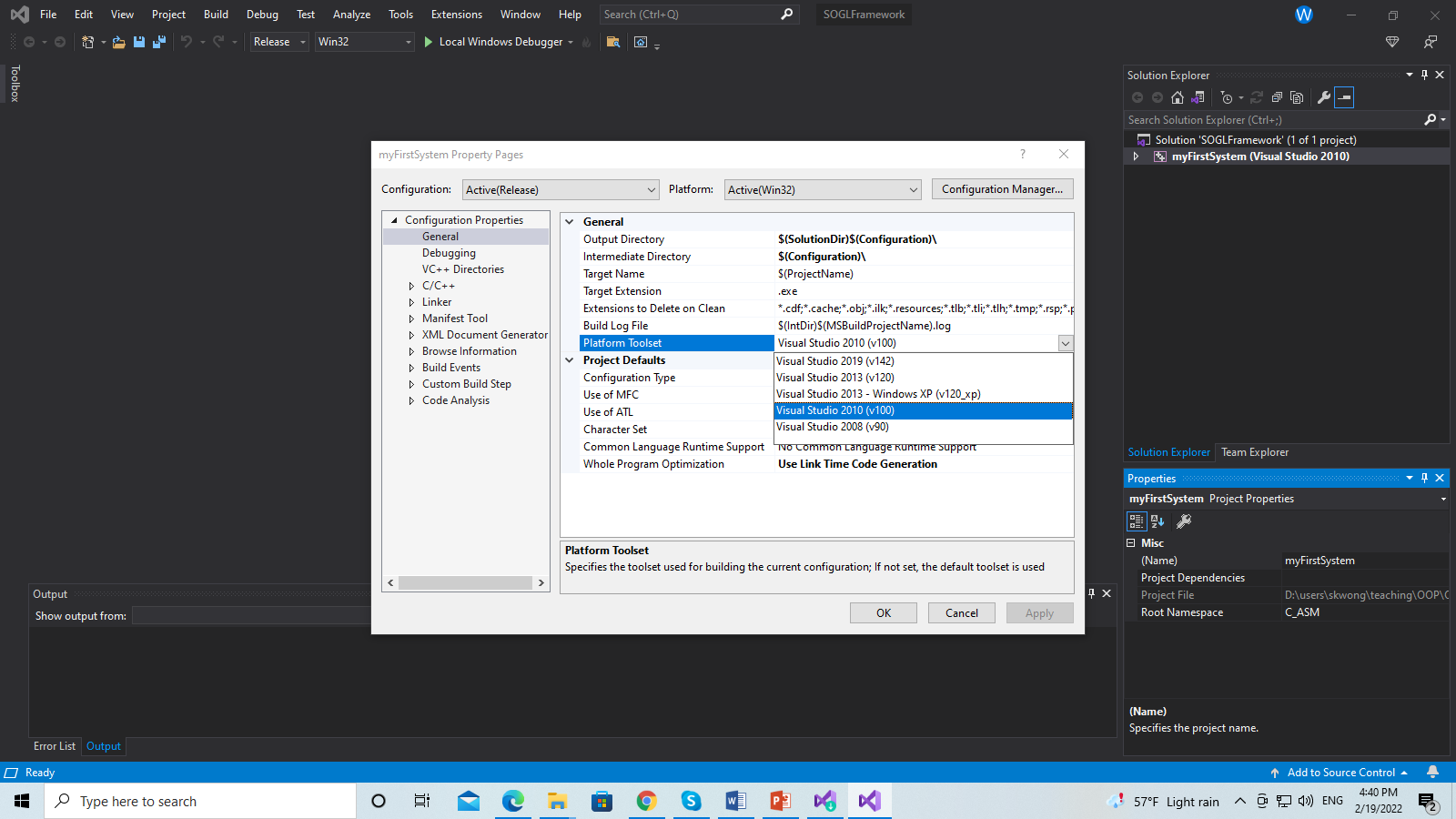
**Figure 5.2.1: We are in Visual Studio 2019. On the right side, you can see the version.**



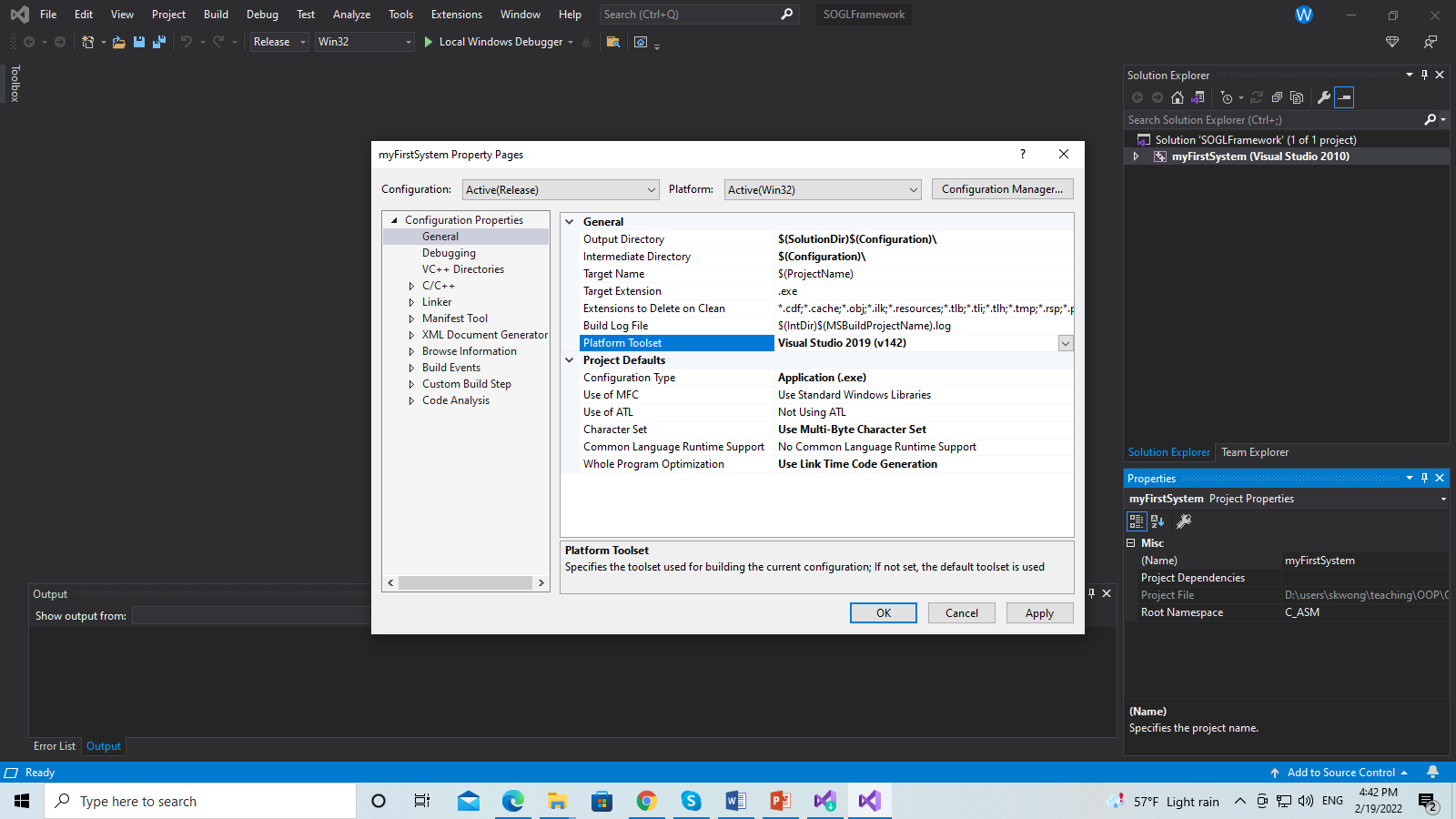
**Figure 5.2.2: The VS version 2010.**



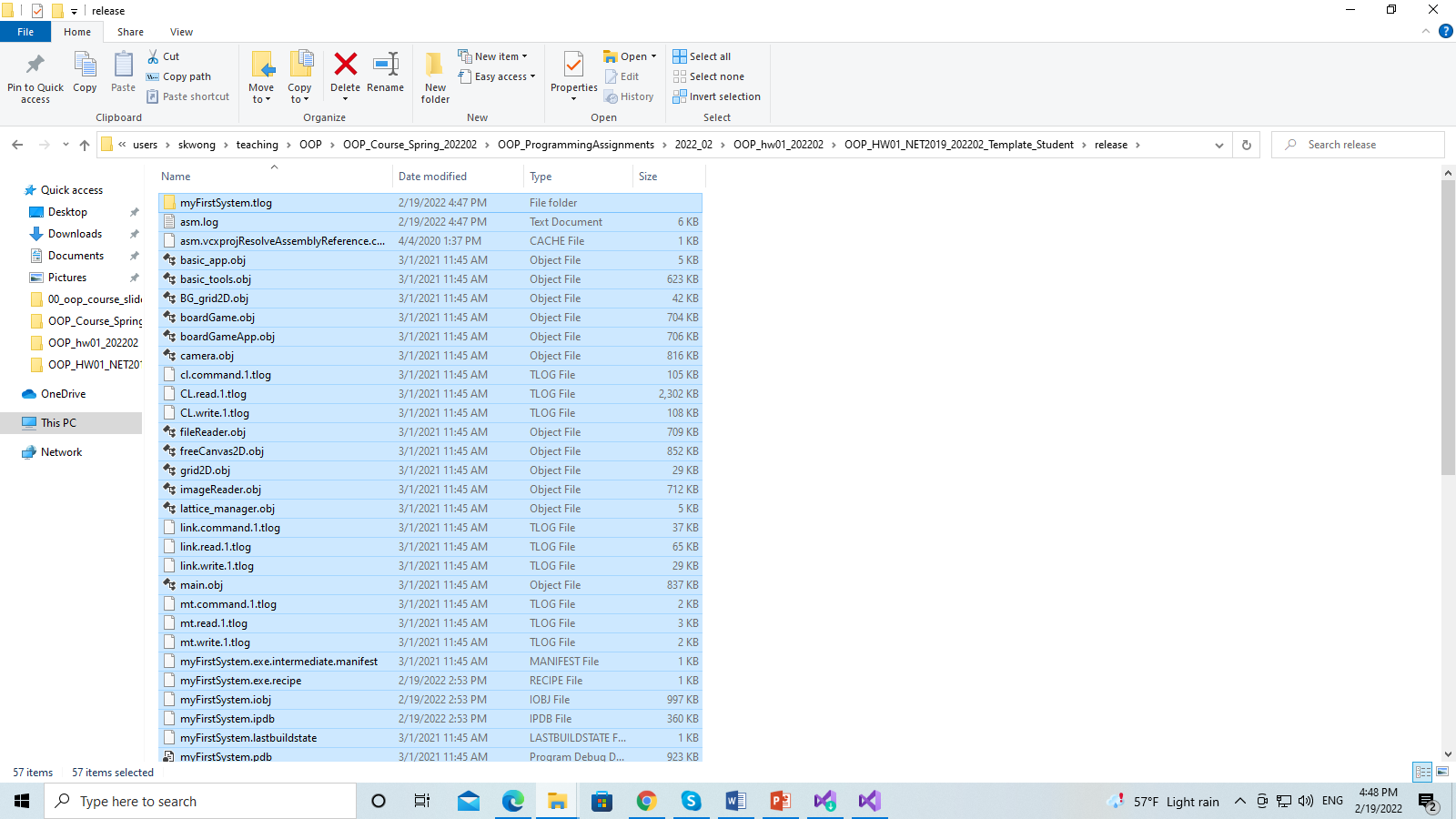
**Figure 5.2.3: Now, go to Project->Project properties**



**Figure 5.2.4: Now, in the project properties, on the left panel, go to: Configuration properties->General. In the dialogue of General, select Platform Toolset and set it to v142.**



**Figure 5.2.5: After setting the Platform Toolset to v142, click “Apply” and then “OK”.**



**Figure 5.2.6: Now, delete all the .obj files and the pdb file in the folder ./Release.**

**In Visual Studio, press ALT+B+B to build the program. Make sure that a message shows “build successfully”. Yea, you can rebuild (CTRL+ALT+F7) everything too :D**

**6. Subsystem mySystem\_MonteCarlo**

We use Monte Carlo simulation to estimate the pi (i.e., π) value. We sample points inside a square.

**Implement the following items**

1. Show a message about the Monte Carlo simulation.
2. The default radius of the circle is 5.0. The default number of sample points is 10000.
3. Use the Monte Carlo simulation to estimate the pi value and show the pi value. Use generateUniformSample ( ) to generate samples. Use computePI( ) to compute the value of pi and display pi on the console window.
4. Implement the member function getRadius( ): Get the radius
5. Implement the member function getNumSamples( ): Get the number of samples.
6. Implement getPI( ): get the computed pi value.
7. Implement reset( ) the data members to their default values. Then regenerate the sample points. After that, estimate and output pi.
8. Implement the member function: bool getSample(int sampleIndex, float &x, float &y) const

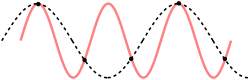
Get the coordinates (x,y) of a sample point based on the sample index (starting from 0). Return true if the sample point lies inside the circle and otherwise return false.

1. Press ‘9’ to decrease the number of sampler points by 2000 each time. Compute the new samples. Implement method decrease( )**.**
2. Press ‘0’ to increase the number of sampler points by 2000 each time. Compute the new samples. Implement method increase( ).
3. Press ‘n’ to decrease the radius of the circle by 1. The smallest radius is 1. Compute the new samples and output pi**.** Implement method decreaseRadius( )**.**
4. Press ‘m’ to increase the radius of the circle by 1. The largest radius is 10. Compute the new samples and output pi**.** Implement method increaseRadius( )**.**
5. Press ‘r’ to reset. In this case, reset( ) will be called by the client program. Make sure that your function reset( ) is implemented correctly.

**7. Subsystem mySystem\_SineCosineFunction**

We draw the curve of a function *y = f(x)* given the interval of the independent variable *x*.

**Implement the following items.**

1. The default number of sample points for drawing the curve is 50.
2. The interval of x is [-10, 10].
3. The default value setting: a = 1, b = 2.
4. void reset( ): reset all the data members to their default values.
5. void getXInterval( double &x0, double &x1 ) const: get the interval of x.
6. int getNumOfSamples( ) const: get the number of sample points.
7. double getFunctionValue( double x ) const: get the value of the function for a given x value.
8. Press ‘5’ to decrease coefficient a by 0.05.
9. Press ‘6’ to increase coefficient a by 0. 05.
10. Press ‘7’ to decrease coefficient b by 1.
11. Press ‘8’ to increase coefficient b by 1.
12. Press ‘9’ to decrease the number of samples by 2. The minimum number of samples is 2.
13. Press ‘0’ to increase the number of samples by 2. The maximum number of samples is 100.
14. Press ‘r’ to reset. In this case, reset( ) will be called by the client program. Make sure that your function reset( ) is implemented correctly.

The function is: *f*(*x*) *= a cos* (*b x*). Do some experiments. Have fun! Do you know the **Nyquist–Shannon sampling theorem?** The right image is from wiki**:** [Nyquist–Shannon sampling theorem - Wikipedia](https://en.wikipedia.org/wiki/Nyquist%E2%80%93Shannon_sampling_theorem)

**8. Subsystem mySystem\_CubicFunction**

Purpose: Draw the cubic curve.

**Implement the following items.**

1. Set the interval of x as [-100, 100].
2. Set the default number of sample points as 250.
3. void reset( ): reset all the data members to their default values.
4. void getXInterval( double &x0, double &x1 ) const: get the interval of x.
5. int getNumOfSamples( ) const: get the number of sample points.
6. double getFunctionValue( double x ) const: get the value of the function for a given x value.
7. Press ‘n’ to decrease coefficient b by 0.01
8. Press ‘m’ to increase coefficient b by 0.01.
9. Press ‘<’ to decrease coefficient c by 0.1.
10. Press ‘>’ to increase coefficient c by 0.1.
11. Press ‘r’ to reset. In this case, reset( ) will be called by the client program. Make sure that your function reset( ) is implemented correctly.

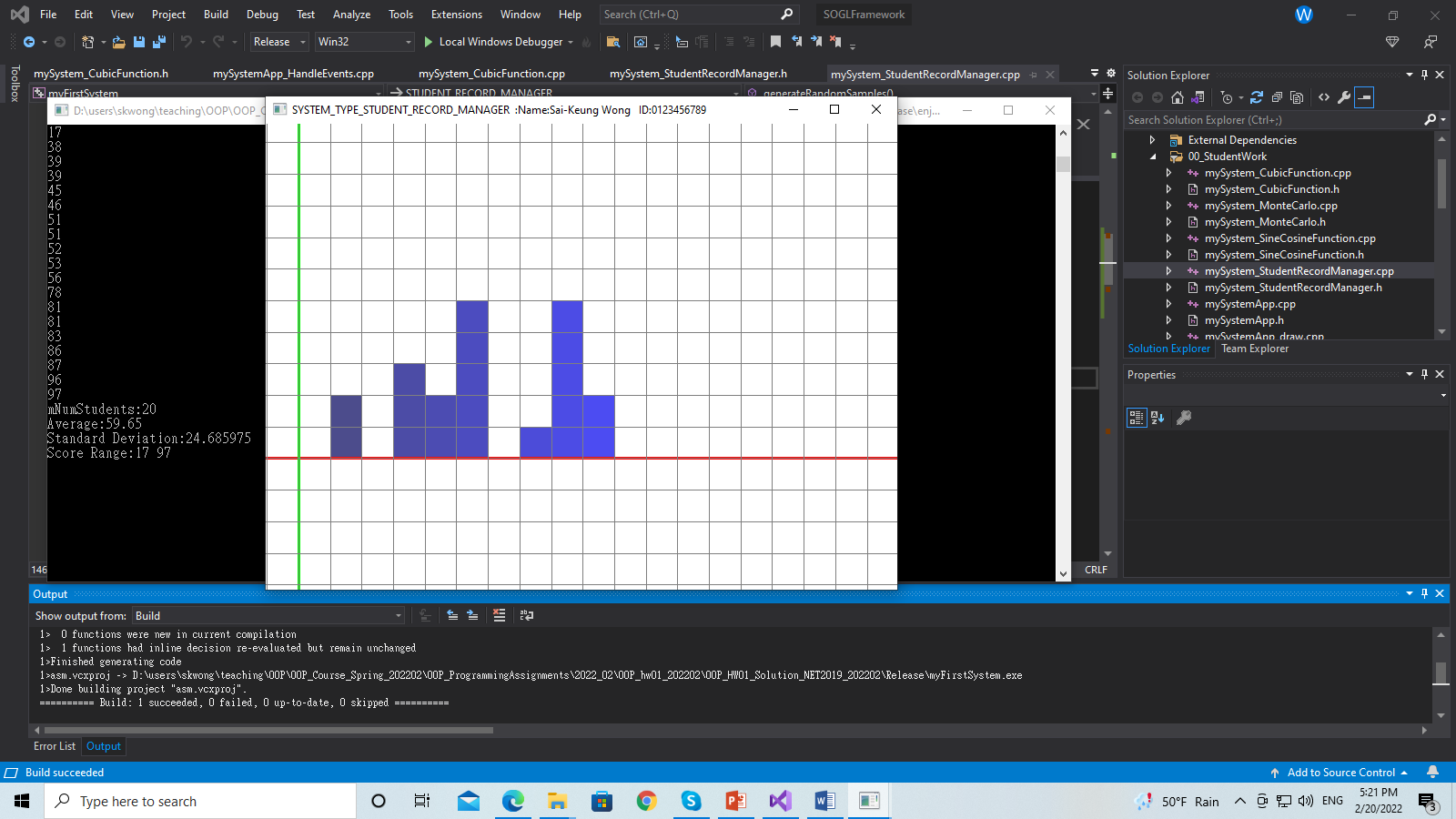
The cubic function is: f(x) = ax3 + bx2 + cx + d. **The default values: a = 0.02, b is = -0.3. c = d = 0.5;**

Play with it. Press ‘n’ and ‘m’. Observe how the curve shape is changing?

**9. Subsystem mySystem\_StudentManager**

Purpose: Enter some numbers and then compute their average and standard deviation. The program shows the histogram of the scores of all the students. **Implement the following items.**

1. Press ‘i’ or ‘I’ to ask for input.
   1. Ask the user to input the number of students. The number of students is in [2,100].
   2. Ask the user to input the score of each student. The score range is [0, 100].
   3. Show the scores in an ascending order. That is, sort the scores.
   4. Show the number of students.
   5. Show the average score.
   6. Show the standard deviation of the scores.
   7. Show the range of the scores.
2. Get the number of students whose score is inside an interval [s0, s1] (inclusive). Implement getNumOfStudentsInScoreInterval(…).
3. Press ‘r’ to randomly generate scores of 20 students. Sort the scores and show the sorted scores. Show the number of students. Show their average and standard deviation. Finally, show the range of the scores. Implement generateRandomSamples( ).



Standard deviation = sqrt( sum(x – x’)\*( x – x’)/(n-1) ), for scores of all students, where x is the score of a student, x’ is the average, and n is the number of students. Read the article about standard deviation in Wiki if you are not sure what it is.

Hint: Use ( rand( ))/(double) (RAND\_MAX)

to compute a random value between [0, 1].

%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%

%%%%%%%%%%%%%%%%%**Enjoy programming**%%%%%%%%%%%%%%%%%

%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%