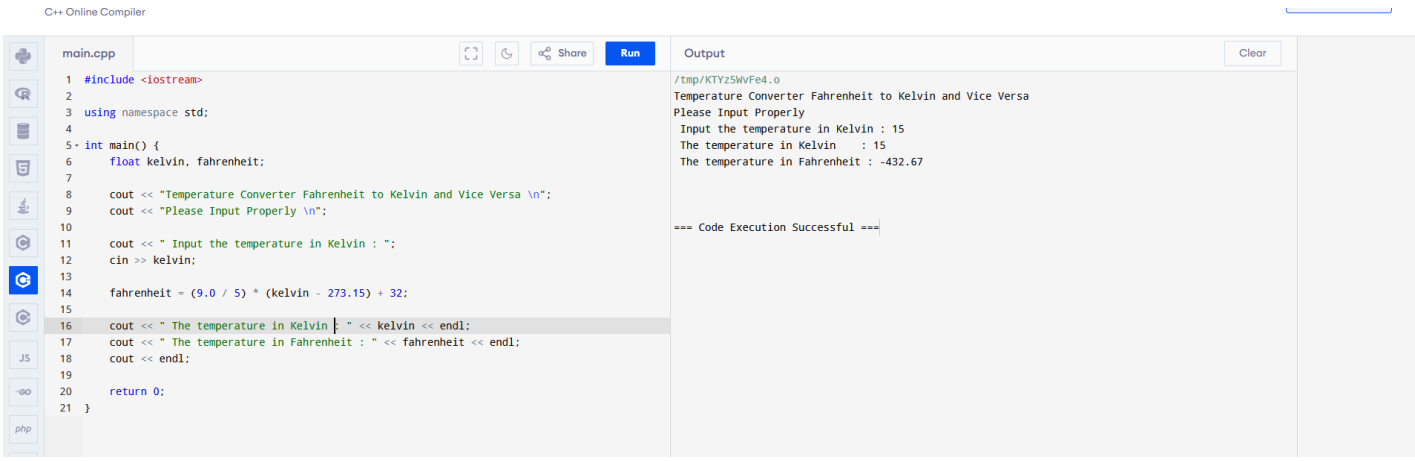
Activity No. 2	
Hands-on Activity 1.2 Basic C++ Programming	
Course Code: CPE010	**Program:** Computer Engineering
Course Title: Data Structures and Algorithms	**Date Performed:** Sep 9, 2024
Section: CPE21S4	**Date Submitted:** Sep 11, 2024
Name(s): Carl Jervie B. Carag	**Instructor:** Eng. Rizette Sayo
6. Output	

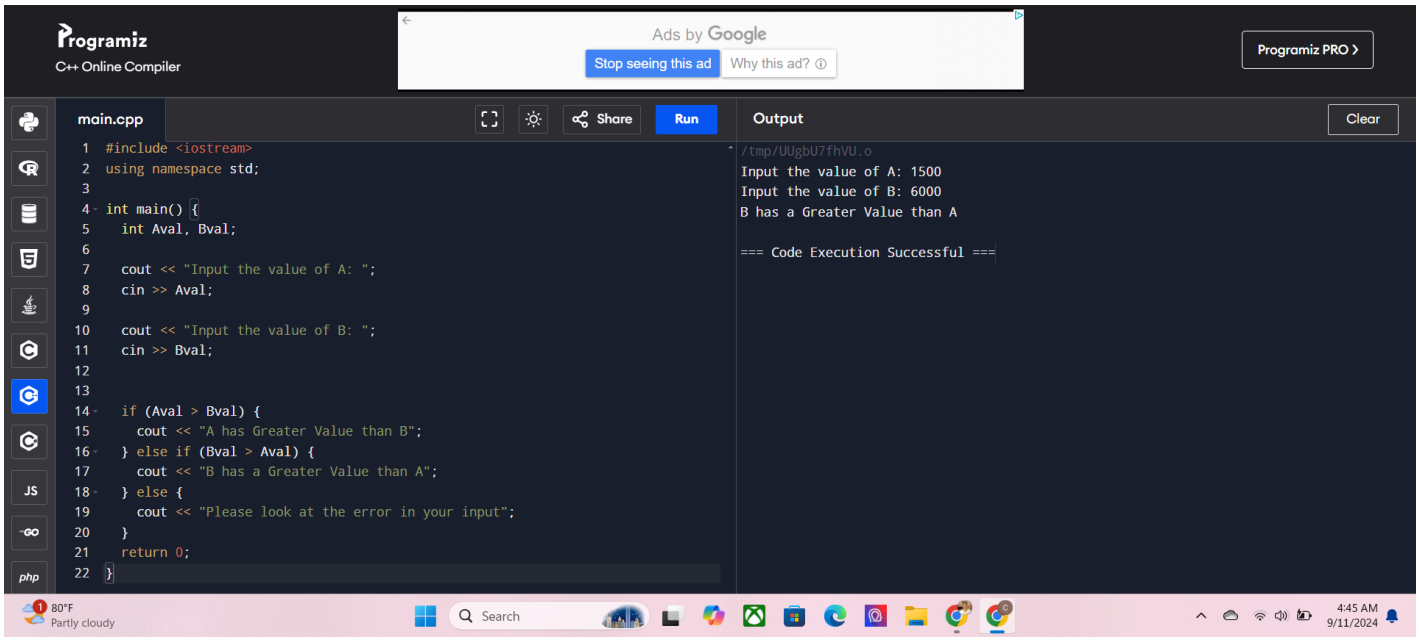
Section	Body of the Code
The First Step and foundation is Header File Declaration Section	#include <iostream> `2 #include <iostream>`
After that do the Global Declaration Section	class Triangle{ private: double totalAngle, angleA, angleB, angleC; `class Triangle{` `private:` `double totalAngle, angleA, angleB, angleC;`
Now make the Class Declaration and Method Definition Section	public: Triangle(double A, double B, double C); void setAngles(double A, double B, double C); const bool validateTriangle(); }; Triangle::Triangle(double A, double B, double C) { angleA = A; angleB = B; angleC = C; totalAngle = A+B+C; } `public:` `Triangle(double A, double B, double C);` `void setAngles(double A, double B, double C);` `const bool validateTriangle();` `};`
Construct your Main Function	void Triangle::setAngles(double A, double B, double C) { angleA = A; angleB = B; angleC = C; totalAngle = A+B+C; } const bool Triangle::validateTriangle() { return (totalAngle <= 180); }

	`` void Triangle::setAngles(double A, double B, double C) { angleA = A; angleB = B; angleC = C; totalAngle = A+B+C; } const bool Triangle::validateTriangle() { return (totalAngle <= 180); } ```
For the final output to operate do “Method Definition.”	int main(){ //driver code Triangle set1(40, 30, 110); if(set1.validateTriangle()){ std::cout<<"The Shape is Valid Triangle.\n"; } else { std::cout<<"The Shape is NOT a Triangle.\n"; } return 0; } ``` int main(){ //driver code Triangle set1(40, 30, 110); if(set1.validateTriangle()){ std::cout<<"The Shape is Valid Triangle.\n"; } else { std::cout<<"The Shape is NOT a Triangle.\n"; } return 0; } ```

7. Supplementary Activity



Supplementary Activity 1



Supplementary Activity 2

8. Conclusion

Overall the activities about the usage of proper library header function is very important in the start of our code, proper input of the appropriate header is the root of the code. The next is the arrangement of the indention and the correct syntax in the data types we declare for each of the variable that we are going to use is crucial in order for the execution of the code to work. The parameters we used in the code enables us to see how we may use the variables declared for each operation our data type is capable of. The supplementary activities reminds us that the first roots of the python codes in which are now learning is the basic fundamental lesson that needs to be memorized because majority of what we could apply in python programming could be taken also from our past lesson in c++. I will consider that the else if statement in the C++ is also one of the lessons that is easier to grasp and learn wherein it execute a body of code that verifies if the condition statement was true (if) and proceed to other function which is (else) if false.

9. Assessment Rubric