Hands-on Activity 2.1	
Data Types and Arithmetic Operations	
Course Code: CPE007	**Program:** Computer Engineering
Course Title: Programming Logic and Design	**Date Performed:** 8/7/25
Section: CPE11S1	**Date Submitted:** 8/7/25
Name(s): Angel Mae Ramirez	**Instructor:** Engr. Jimlord Quejado

6. Output

Example 1: The following program has an output of:

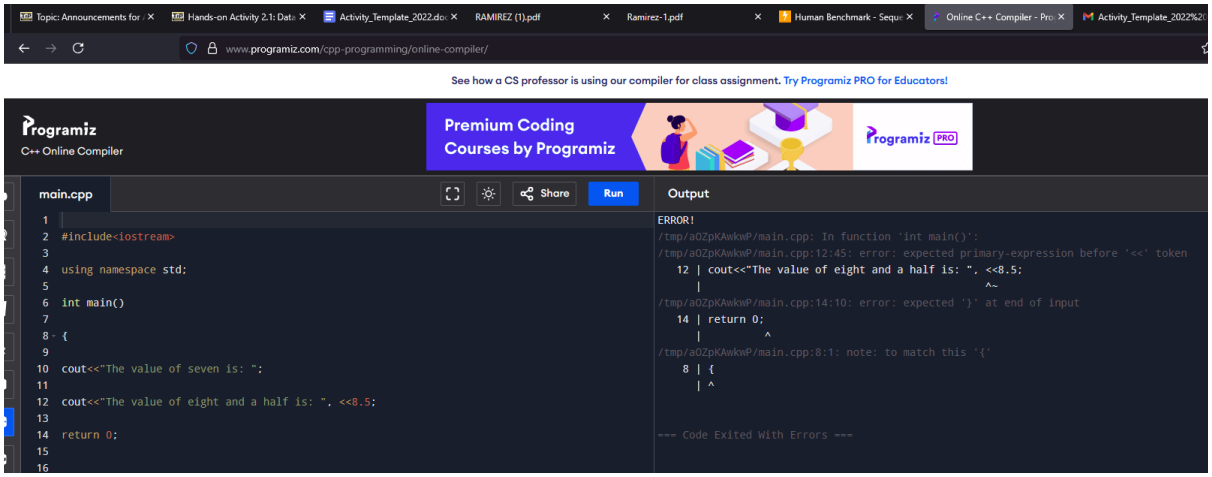
The value of seven is: 7.000000

The value of eight and a half is: 8.500000

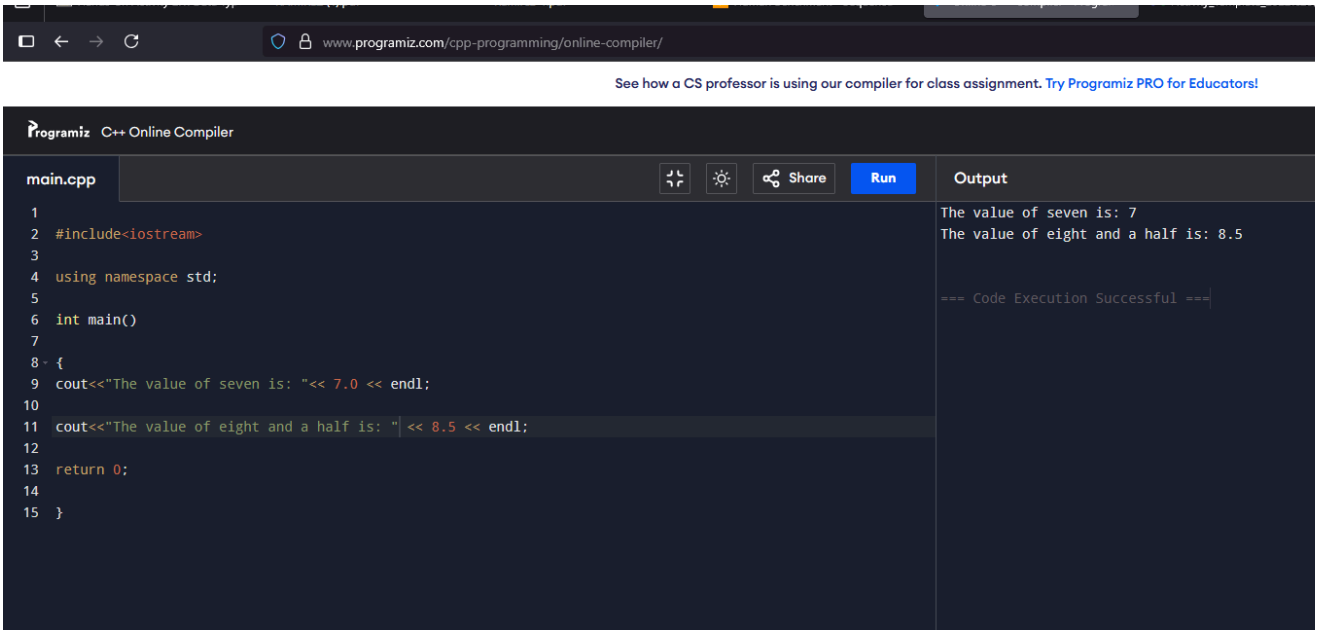
Can you find all possible compilation errors and logic errors? Can you fix them to print the same result as the expected output? Before you use your compiler, try to find the errors only by manual code analysis.

```
#include<iostream>

using namespace std;

int main()
{
cout<<"The value of seven is: ";

cout<<"The value of eight and a half is: ", <<8.5;

return 0;e
```

ERRORS



FIXED RESULTS



Example 2: The following program has an output of:

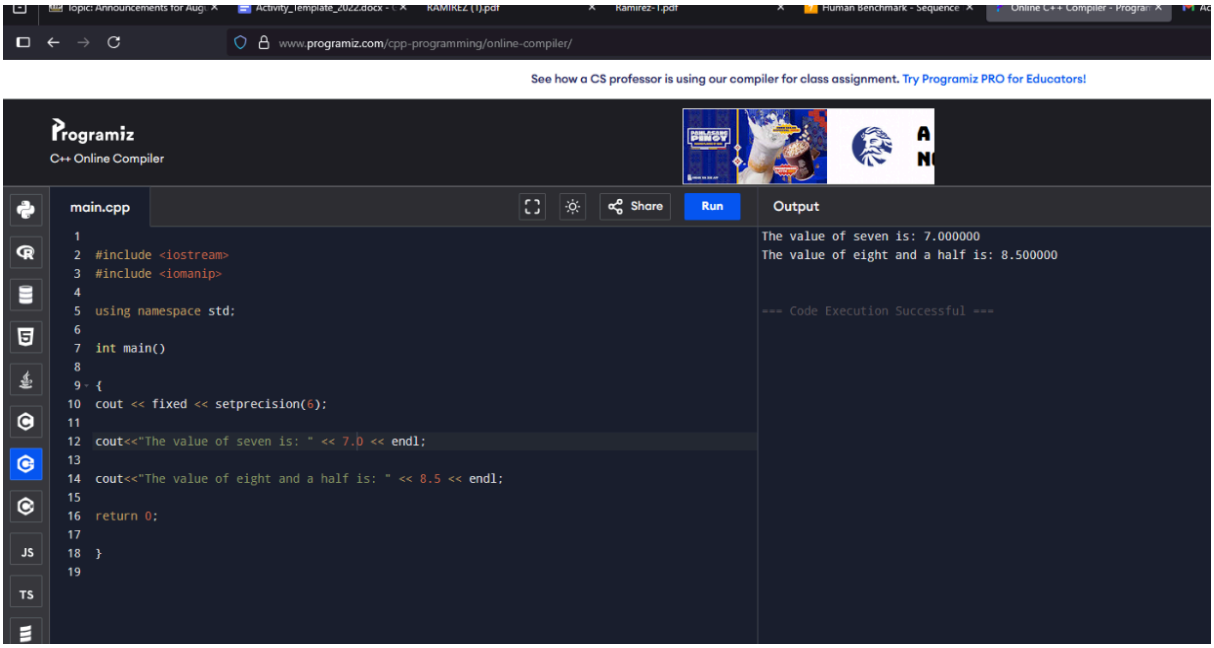
The value of seven is: 7.000000

The value of eight and a half is: 8.500000

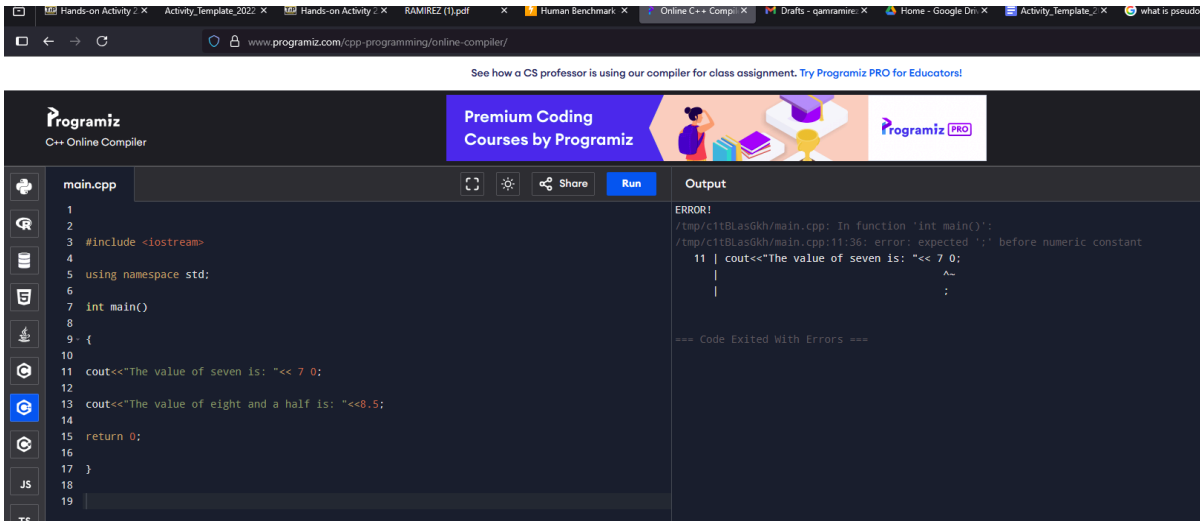
Can you find all possible compilation errors and logic errors? Can you fix them to print the same result as the expected output? Before you use your compiler, try to find the errors only by manual code analysis.

```
#include <iostream>

using namespace std;

int main()

{

cout<<"The value of seven is: "<< 7 0;

cout<<"The value of eight and a half is: "<<8.5;

return 0;

}
```

ERRORS



FIXED RESULTS



Example 3: The following program has an output of:

The value of half is: 0.500000

The value of Pi is: 3.141593

Can you find all possible compilation errors and logic errors? Can you fix them to print the same result as the expected output? Before you use your compiler, try to find the errors only by manual code analysis.

int main()

{

float halfValue = 0.6;

float piValue = 3.141 592 65;

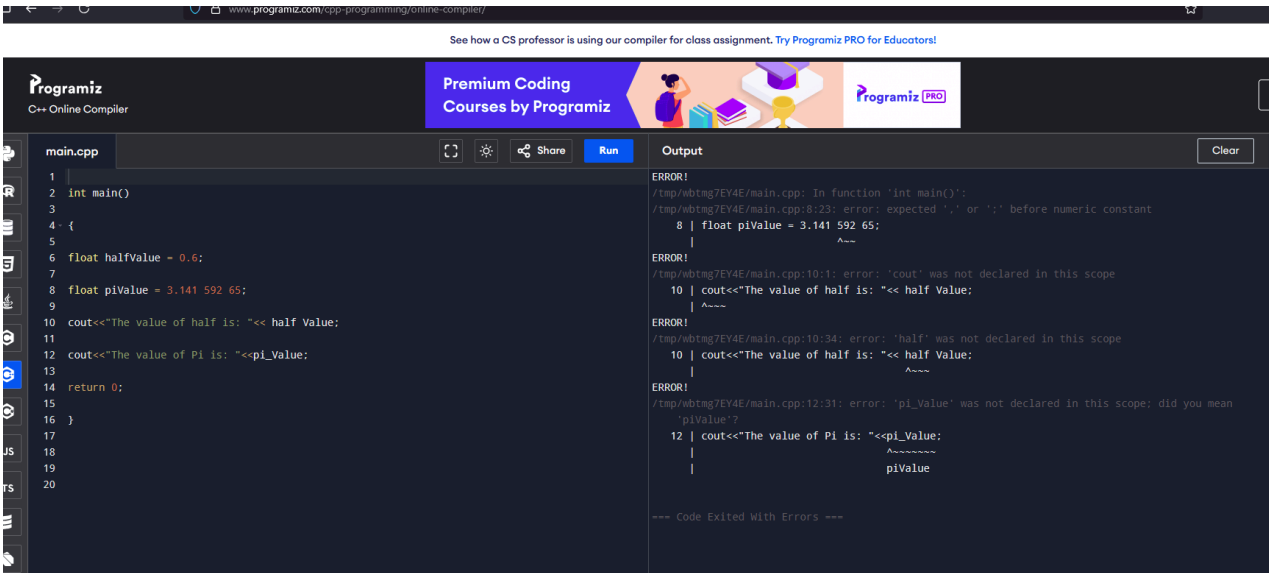
cout<<"The value of half is: "<< half Value;

cout<<"The value of Pi is: "<<pi_Value;

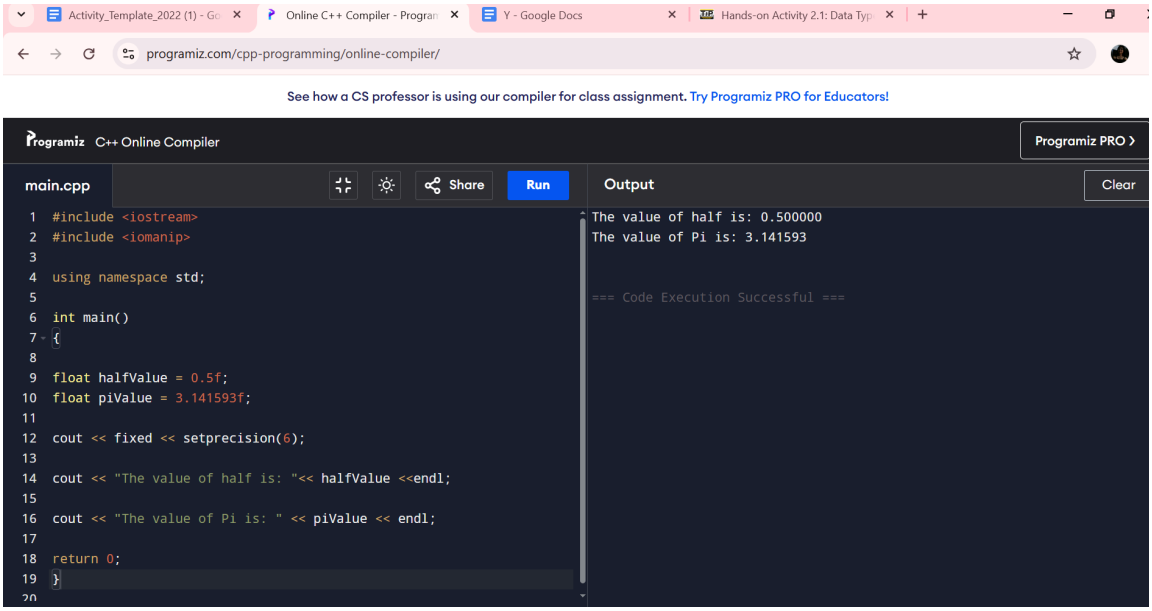
return 0;

}

ERRORS:



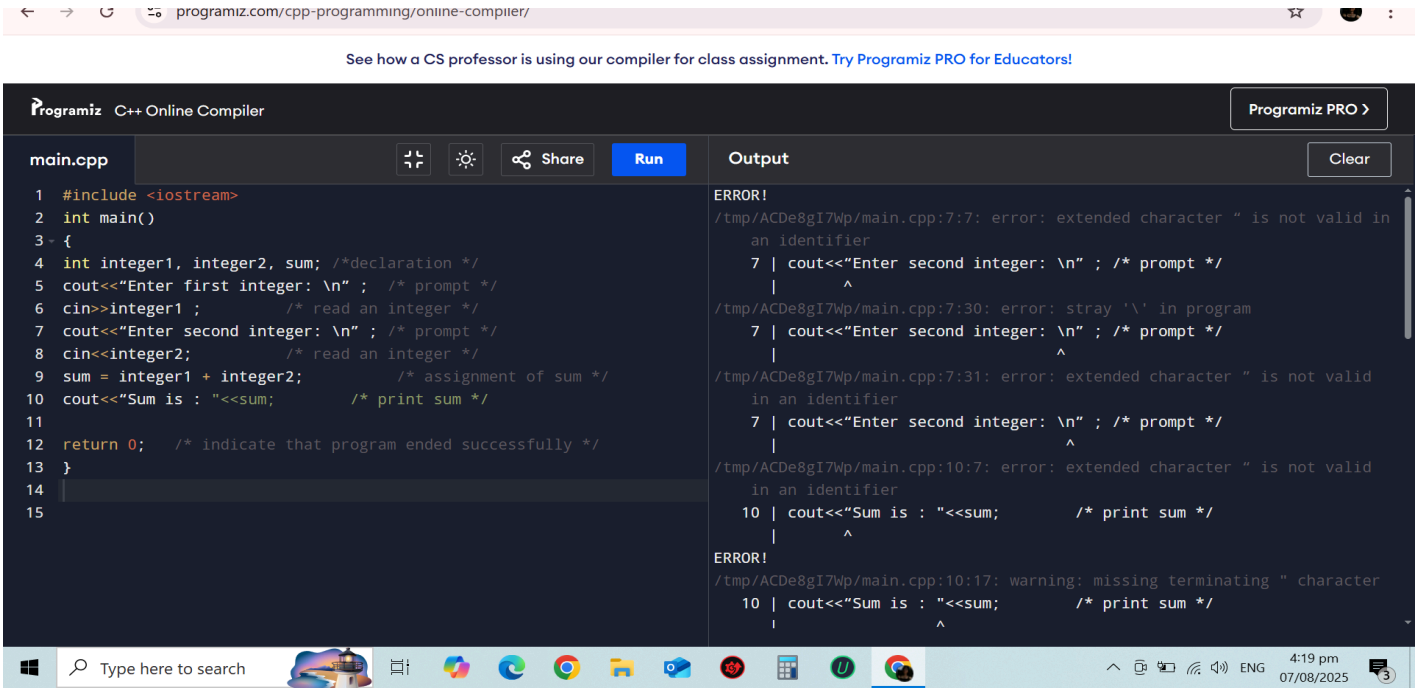
FIXED RESULT



Example 4: Sample program for Adding Two Integers

```
#include <iostream>
int main()
{
int integer1, integer2, sum; /*declaration */
cout<<“Enter first integer: \n” ;  /* prompt */
cin>>integer1 ;              /* read an integer */
cout<<“Enter second integer: \n” ; /* prompt */
cin<<integer2;               /* read an integer */
sum = integer1 + integer2;          /* assignment of sum */
cout<<“Sum is : "<<sum;        /* print sum */

return 0;   /* indicate that program ended successfully */
}
```

ERRORS



programiz.com/cpp-programming/online-compiler/

See how a CS professor is using our compiler for class assignment. Try Programiz PRO for Educators!

Programiz C++ Online Compiler — Programiz PRO ›

main.cpp — Share — Run

```
1  #include <iostream>
2  int main()
3  {
4  int integer1, integer2, sum; /*declaration */
5  cout<<“Enter first integer: \n” ;  /* prompt */
6  cin>>integer1 ;         /* read an integer */
7  cout<<“Enter second integer: \n” ; /* prompt */
8  cin<<integer2;          /* read an integer */
9  sum = integer1 + integer2;          /* assignment of sum */
10 cout<<“Sum is : "<<sum;        /* print sum */
11
12 return 0;   /* indicate that program ended successfully */
13 }
14
15
```

Output — Clear

```
/tmp/ACDe8gI7Wp/main.cpp:10:17: error: missing terminating " character
   10 | cout<<“Sum is : "<<sum;        /* print sum */
      |                 ^~~~~~~~~~~~~~~~~~~~~~~~~~~~~
ERROR!
/tmp/ACDe8gI7Wp/main.cpp: In function 'int main()':
/tmp/ACDe8gI7Wp/main.cpp:5:1: error: 'cout' was not declared in this
    scope; did you mean 'std::cout'?
    5 | cout<<"Enter first integer: \n" ;  /* prompt */
      | ^~~~
      | std::cout
In file included from /tmp/ACDe8gI7Wp/main.cpp:1:
/usr/local/include/c++/14.2.0/iostream:63:18: note: 'std::cout' declared
    here
   63 |   extern ostream cout;          ///< Linked to standard output
      |                  ^~~~
ERROR!
/tmp/ACDe8gI7Wp/main.cpp:6:1: error: 'cin' was not declared in this scope
    ; did you mean 'std::cin'?
    6 | cin>>integer1 ;         /* read an integer */
      | ^
```

programiz.com/cpp-programming/online-compiler/

See how a CS professor is using our compiler for class assignment. Try Programiz PRO for Educators!

Programiz C++ Online Compiler — Programiz PRO ›

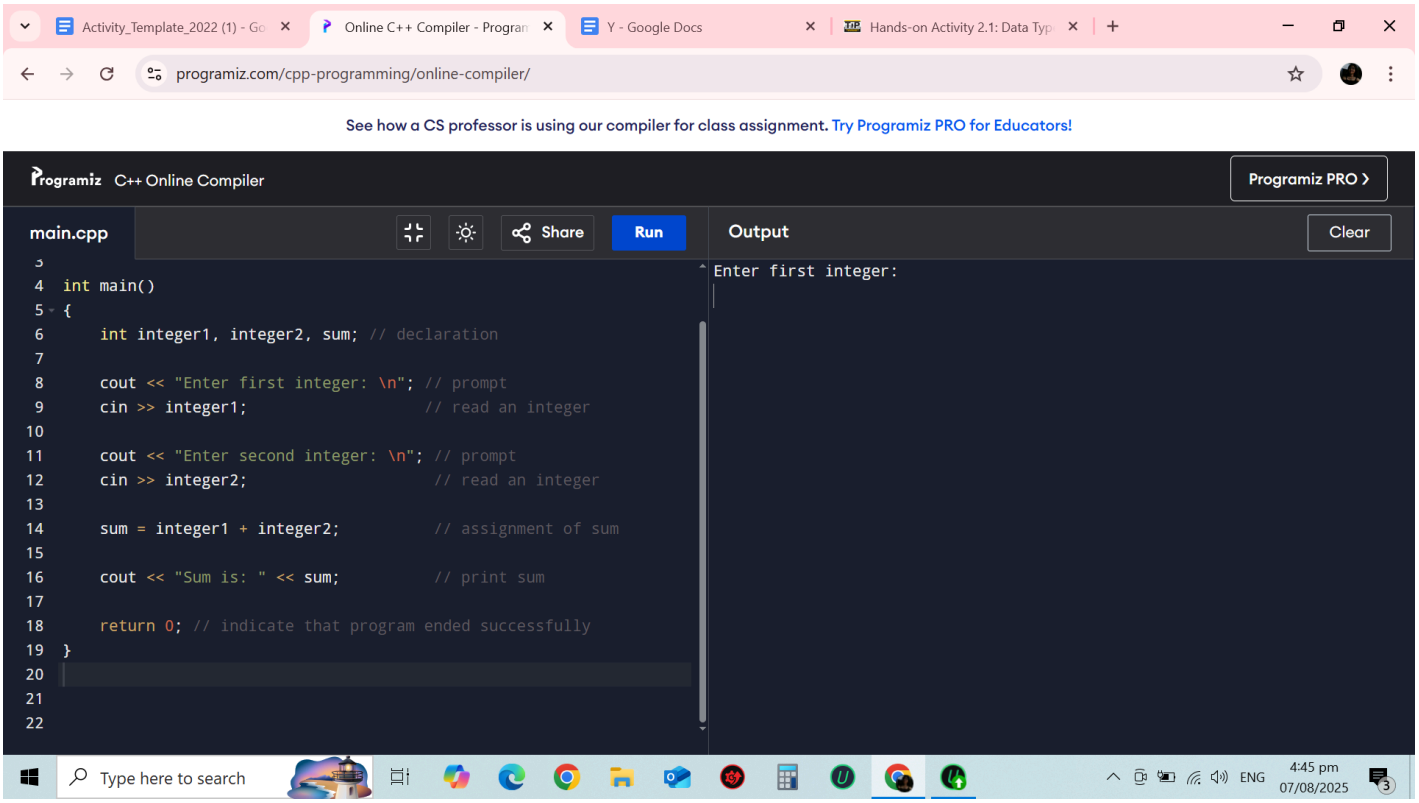
main.cpp — Share — Run

```
1  #include <iostream>
2  int main()
3  {
4  int integer1, integer2, sum; /*declaration */
5  cout<<“Enter first integer: \n” ;  /* prompt */
6  cin>>integer1 ;         /* read an integer */
7  cout<<“Enter second integer: \n” ; /* prompt */
8  cin<<integer2;          /* read an integer */
9  sum = integer1 + integer2;          /* assignment of sum */
10 cout<<“Sum is : "<<sum;        /* print sum */
11
12 return 0;   /* indicate that program ended successfully */
13 }
14
15
```

Output — Clear

```
ERROR!
/tmp/ACDe8gI7Wp/main.cpp:6:1: error: 'cin' was not declared in this scope
    ; did you mean 'std::cin'?
    6 | cin>>integer1 ;         /* read an integer */
      | ^~~
      | std::cin
/usr/local/include/c++/14.2.0/iostream:62:18: note: 'std::cin' declared
    here
   62 |   extern istream cin;           ///< Linked to standard input
      |                  ^~~
ERROR!
/tmp/ACDe8gI7Wp/main.cpp:7:7: error: '\U0000201cEnter' was not declared
    in this scope
    7 | cout<<“Enter second integer: \n” ; /* prompt */
      |       ^~~~~~
ERROR!
/tmp/ACDe8gI7Wp/main.cpp:10:7: error: '\U0000201cSum' was not declared in
    this scope
   10 | cout<<“Sum is : "<<sum;        /* print sum */
```

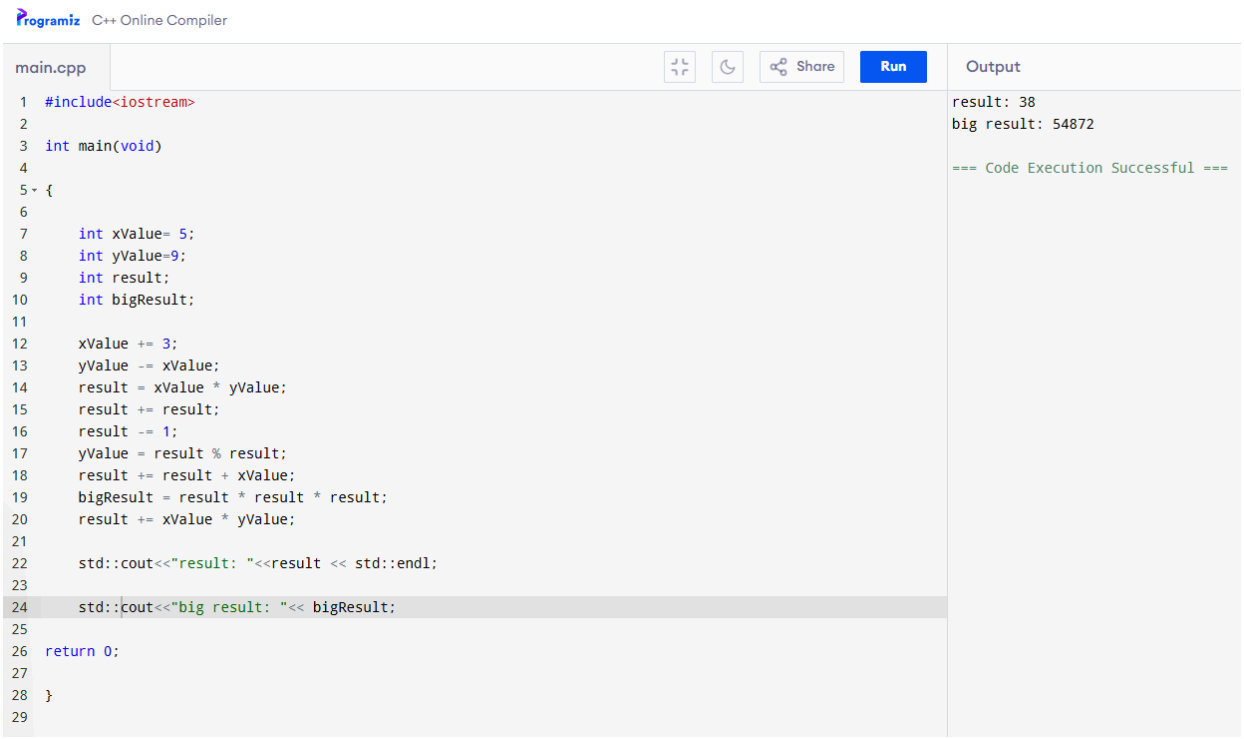
FIXED RESULTS



7. Supplementary Activity

1.Take a look at the code below: it assigns two integer values, manipulates them and finally outputs the result and bigresult variables. The problem is that the manipulations have been described using natural language, so the code is completely useless now. Act as an intelligent (naturally!) compiler and translate the formula into a real "C" code notation. Test your code using the data provided.

```
#include <iostream>
using namespace std;
int main(void)
{
int xValue=5;
int yValue=9;
int result;
int bigResult;
/*
increment xValue by 3
decrement yValue by xValue
multiply xValue times yValue giving result
increment result by result
decrement result by 1
assign result modulo result to yValue
increment result by result added to xValue
assign result times result times result to bigResult increment result by xValue times yValue
*/cout<<"result: "<<result;
cout<<"big result: "<< bigResult;
return 0
```



2. Complete the program below. Compute the accrued amount of money with a starting value of 100 and an annual interest rate of 1.5%. Compute and print the results for first three years. Your version of the program must print the same result as the expected output for every year. Compute each annual value on the basis of the previous year's value.

```
#include <iostream>
using namespace std;
int main()
{
float startValue = 100;
float interestRate = 0.015;
float firstYearValue;
float secondYearValue;
float thirdYearValue;
/* Your code *
```

Programiz C++ Online Compiler

main.cpp | Share | Run | Output

```
#include<iostream>

int main()

{

    float startValue = 100;

    float interestRate = 0.015;

    float firstYearValue;

    float secondYearValue;

    float thirdYearValue;

    firstYearValue = startValue * (1 + interestRate);
    secondYearValue = firstYearValue * (1 + interestRate);
    thirdYearValue = secondYearValue * (1 + interestRate);

    std::cout<<"After first year: "<<firstYearValue << std::endl;
    std::cout<<"After second year: "<<secondYearValue << std::endl;
    std::cout<<"After third year: "<<thirdYearValue << std::endl;

return 0;

}
```

```
After first year: 101.5
After second year: 103.022
After third year: 104.568


=== Code Execution Successful ===
```

8. Conclusion

Through correcting this C++ programming errors, I have learned the importance of understanding both syntax and logic when writing and debugging code. I realize that small mistakes like using square brackets instead of curly braces, missing semicolons, or misusing input/output operators (>> and <<), can cause compilation errors that stop the program from running entirely. I also discovered how logic errors like assigning incorrect values or using the wrong variable names can produce incorrect results even if the code compiles. And also I learned that proper formatting is crucial when displaying output, especially when working with floating-point numbers. Using functions like fixed and setprecissions from the `<iomanip>` library ensures output matches expected precision. I also understood the importance of using standard double quotes (") instead of smart quotes (“ ”) which is not recognized by C++ compilers. Overall, this exercise improved my attention to detail and deepened my understanding of how to write clean, correct, and readable C++ code.

9. Assessment Rubric

Rubric for SO 7 (6)

Criteria	Ratings						Pts
SO 7 PI 1 IILO4 Utilize lifelong learning skills in pursuit of personal development and excellence in professional practice. threshold: 4.8 pts	6 pts Excellent | Educational interests and pursuits exist and flourish outside classroom requirements,knowledge and/or experiences are pursued independently and applies knowledge learned into practice	5 pts Good | Educational interests and pursuits exist and flourish outside classroom requirements,knowledge and/or experiences are pursued independently	4 pts Satisfactory | Look beyond classroom requirements, showing interest in pursuing knowledge independently	3 pts Unsatisfactory | Begins to look beyond classroom requirements, showing interest in pursuing knowledge independently	2 pts Poor | Relies on classroom instruction only	1 pts Very Poor | No initiative or interest in acquiring new knowledge	6 p
SO 7 PI 2 IILO4 Utilize lifelong learning skills in pursuit of personal development and excellence in professional practice. threshold: 4.8 pts	6 pts Excellent | Completes an assigned task independently and practices continuous improvement	5 pts Good | Completes an assigned task without supervision or guidance	4 pts Satisfactory | Requires minimal guidance to complete an assigned task	3 pts Unsatisfactory | Requires detailed or step-by-step instructions to complete a task	2 pts Poor | Shows little interest to complete a task independently	1 pts Very Poor | No interest to complete a task independently	6 p
SO 7 PI 3 IILO4 Utilize lifelong learning skills in pursuit of personal development and excellence in professional practice. threshold: 4.8 pts	6 pts Excellent | Synthesizes and integrates information from a variety of sources; formulates a clear and precise perspective; draws appropriate conclusions	5 pts Good | Evaluate information from a variety of sources; formulates a clear and precise perspective.	4 pts Satisfactory | Analyze information from a variety of sources; formulates a clear and precise perspective.	3 pts Unsatisfactory | Apply the gathered information to formulate the problem	2 pts Poor | Gather and summarized the information from a variety of sources but failed to formulate the problem	1 pts Very Poor | Gather information from a variety of sources	6 p
SO 7 PI 4 IILO4 Utilize lifelong learning skills in pursuit of personal development and excellence in professional practice. threshold: 4.8 pts	6 pts Excellent | Ideas are combined in original and creative ways in line with the new and emerging technology trends to solve a problem or address an issue.	5 pts Good | Ideas are creative and adapt the new knowledge to solve a problem or address an issue	4 pts Satisfactory | Ideas are creative in solving a problem, or address an issue	3 pts Unsatisfactory | Shows some creative ways to solve the problem	2 pts Poor | Shows initiative and attempt to develop creative ideas to solve the problem	1 pts Very Poor | Ideas are copied or restated from the sources consulted	6 p

Total Points: 2