

Activity No. 2.1	
Hands-on Activity 2.1: Data Types and Arithmetic Operations	
Course Code: CPE007	Program: Computer Engineering
Course Title: Programming Design and Logic	Date Performed: 8/6/25
Section: CPE11S1	Date Submitted: 8/7/25
Name(s): Niel Vincent B. Condino	Instructor: Engr. Jimlord M. Quejado
6. Output	
<p>Example 1: The following program has an output of:</p> <p>The value of seven is: 7.000000</p> <p>The value of eight and a half is: 8.500000</p> <p>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.</p> <pre>#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; }</pre> <p>The error lies on the line <code>cout<<"The value of seven is: ";</code> not having the 7.0 value. This can be fixed by adding <code><<7.0;</code> and <code>endl</code> at the end. There is also the error on the line <code>cout<<"The value of eight and a half is: ", <<8.5;</code> due to having “,” before <code><<8.5</code>. This can be fixed by removing that “,”. The problem with the output not displaying other decimal values is because of it lacking the lines <code>#include <iomanip></code> and adding the lines <code>std::cout << std::fixed;</code> and <code>std::cout << std::setprecision(6);</code> inside the function.</p>	

The screenshot shows a C++ development environment with a dark theme. On the left is the code editor containing a file named 'main.cpp' with the following content:

```
1 #include<iostream>
2 #include <iomanip>
3
4 using namespace std;
5
6 int main()
7 {
8 {
9     std::cout << std::fixed;
10    std::cout << std::setprecision(6);
11
12    cout<<"The value of seven is: "<<7.0<endl;
13
14    cout<<"The value of eight and a half is: "<<8.5;
15
16    return 0;
17
18 }
```

At the top of the interface, there are buttons for 'Run' and 'Output'. The 'Output' tab is selected, showing the following text:

The value of seven is: 7.000000
The value of eight and a half is: 8.500000
*** Code Execution Successful ***

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;
}
```

The error lies on the line `cout<<"The value of seven is: "<< 7 0;` **not having a “.” between 7 and 0.** This can be fixed by replacing `7 0` with `7.0`. The problem with the output not displaying other decimal values is because of it lacking the lines `#include <iomanip>` and adding the lines `std::cout << std::fixed;` and `std::cout << std::setprecision(6);` inside the function.

The screenshot shows a C++ development environment with a dark theme. On the left, the code editor displays a file named 'main.cpp' containing the following code:

```
1 #include<iostream>
2 #include <iomanip>
3
4 using namespace std;
5
6 int main()
7 {
8     std::cout << std::fixed;
9     std::cout << std::setprecision(6);
10    :
11
12    cout<<"The value of seven is: " <<7.0<<endl;
13
14    cout<<"The value of eight and a half is: " <<8.5;
15
16    return 0;
17 }
18 }
```

On the right, the 'Output' window shows the execution results:

```
The value of seven is: 7.000000
The value of eight and a half is: 8.500000
*** Code Execution Successful ***
```

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;
}
```

The error lies on different lines, first 3.141 592 65 must have no spaces. It can be fixed by replacing it with 3.14159265. The value of halfValue is also wrong. This can be fixed by replacing its value with 0.5. Next errors are on the misnamed half Value and pi_Value. These errors can be fixed by renaming them the same as the original variable halfValue and piValue. You also need to add the lines #include <iostream> and using namespace std; at the top. The problem with the output not displaying other decimal values is because of it lacking the lines #include <iomanip> and adding the lines std::cout << std::fixed; and std::cout << std::setprecision(6); inside the function.

```
main.cpp
1 #include<iostream>
2 #include <iomanip>
3
4 using namespace std;
5
6 int main()
7 {
8     float halfValue = 0.5;
9     float piValue = 3.14159265;
10
11    std::cout << std::fixed;
12    std::cout << std::setprecision(6);
13
14    cout<<"The value of half is: "<<halfValue<<std::endl;
15    cout<<"The value of Pi is: "<<piValue;
16
17    return 0;
18 }
```

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 */
}
```

Sample Output:

Enter first integer

Enter second integer

72

Sum is 117

The first problem is that it does not include using namespace std;. This can be fixed by adding that line or by adding std:: to every function that needs it. The next error is on the line cin<< using << instead of >>. This can be fixed by replacing cin<< with cin>>.

```
main.cpp
1 #include <iostream>
2
3 using namespace std;
4
5 int main()
6 {
7     int integer1, integer2, sum; /*declaration */
8
9     cout<<"Enter first integer: \n"; /* prompt */
10
11    cin>>integer1 ; /* read an integer */
12
13    cout<<"Enter second integer: \n"; /* prompt */
14
15    cin>>integer2; /* read an integer */
16
17    sum = integer1 + integer2; /* assignment of sum */
18
19    cout<<"Sum is : "<<sum; /* print sum */
20
21    return 0; /* indicate that program ended successfully */
22 }
```

Output

```
Enter first integer:  
45  
Enter second integer:  
72  
Sum is : 117  
== Code Execution Successful ==
```

7. Supplementary Activity

Activity 1:

```
main.cpp
1 #include <iostream>
2
3
4 int main(void)
5 {
6
7     int xValue=5;
8
9     int yValue=9;
10
11    int result;
12
13    int bigResult;
14
15
16    xValue +=3;//increment xValue by 3
17
18    yValue -= xValue;//decrement yValue by xValue
19
20    result = xValue * yValue;//multiply xValue times yValue giving result
21
22    result = result + result; //increment result by result
23
24    result = result + result; //increment result by result
25
26    result = result + result; //increment result by result
27
28    yValue = result % result;//assign result modulo result to yValue
29
30    result = result + (xValue + result); //increment result by result added to xValue
31
32    bigResult = result * result * result; //assign result times result times result to bigResult
33
34    result = result + (xValue * yValue); //increment result by xValue times yValue
35
36    std::cout<<result: "<<result<<std::endl;
37
38    std::cout<<"big result: "<< bigResult << std::endl;
39
40    return 0;
41 }
42
```

Output

```
result: 38
big result: 54872
== Code Execution Successful ==
```

Activity 2:

```
main.cpp
1 #include <iostream>
2 #include <iomanip>
3
4 int main()
5 {
6
7     float startValue = 100;
8
9     float interestRate = 0.015;
10
11     float firstYearValue;
12
13     float secondYearValue;
14
15     float thirdYearValue;
16
17     std::cout << std::fixed;
18     std::cout << std::setprecision(5);
19
20     firstYearValue = startValue * (1 + interestRate);
21
22     secondYearValue = firstYearValue * (1 + interestRate);
23
24     thirdYearValue = secondYearValue * (1 + interestRate);
25
26
27
28     std::cout << "After first year: " << firstYearValue << std::endl;
29     std::cout << "After second year: " << secondYearValue << std::endl;
30
31     std::cout << "After third year: " << thirdYearValue << std::endl;
32
33
34     return 0;
35 }
36
```

Output

```
After first year: 101.500000
After second year: 103.022499
After third year: 104.567833
== Code Execution Successful ==
```

8. Conclusion

This activity gave me more insight on how to spot and fix errors on code in the C++ language. I learned about how the arithmetic operations work in this language and how to get the remainder using modulo which is very interesting to find out. This activity also lets me apply what I have learned on the different data types and arithmetic operations in the same language.

9. Assessment Rubric