

Activity No. 1.1

Hands-on Activity 2.1: Data Types and Arithmetic Operations

| | |
|---|---|
| Course Code: CPE010 | Program: Computer Engineering |
| Course Title: Data Structures and Algorithms | Date Performed: |
| Section: CPE11S1 | Date Submitted: |
| Name(s): Tobias, Lawrence C. | Instructor: Engr. Jimlord M. 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.

- The original code had errors such as missing output for the number 7, incorrect use of a comma instead of the <<operator, and no formatting for decimal places. By declaring float variables for the values, using #include <iomanip>, and applying fixed and setprecision(6), the corrected program properly displays the expected output with six decimal places.

```
#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;
}
```

The screenshot shows a C++ online compiler interface. On the left, there is a code editor window titled "main.cpp" containing the following C++ code:

```
1 #include<iostream>
2 #include<iomanip>
3 int main()
4 {
5
6     float cpe11s1 = 7.000000;
7     float cpe11s2 = 8.500000;
8
9     std::cout<<"The value of seven is:" <<std::fixed<<std::setprecision(6)<<cpe11s1<<std::endl
10
11    std::cout<<"The value of eight and a half is: " <<cpe11s2<<std::endl;
12
13
14 }
```

On the right, there are several buttons: a copy icon, a refresh icon, a share icon, and a "Run" button. Below these buttons is an "Output" section. The output text is:
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.

- The original code had errors such as the 70 has a space and it has no decimal points.

```
#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;
```

```
}
```

| | | | | | |
|--|--|--|--|-----|--------|
| main.cpp | | | | Run | Output |
| 1 <code>#include<iostream></code> 2 <code>#include<iomanip></code> 3 <code>int main()</code> 4 <code>{</code> 5 6 <code>float cpe11s1 = 7.000000;</code> 7 <code>float cpe11s2 = 8.500000;</code> 8 9 <code>std::cout<<"The value of seven is:" <<std::fixed<<std::setprecision(6)<<cpe11s1<<std::endl</code> 10 11 <code>std::cout<<"The value of eight and a half is: " <<cpe11s2<<std::endl;</code> 12 <code>return 0;</code> 13 14 <code>}</code> | 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.

- I fixed the code by correcting the variable names, removing spaces in the float values, adding `#include <iomanip>`, and using `fixed` with `setprecision(6)` to format the output. Now, the program correctly displays the values of half and Pi with six decimal places.

```
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;
}
```

main.cpp

Run

Output

The value of half is: 0.500000
The value of Pi is: 3.141593
== Code Execution Successful ==

Example 4: Sample program for Adding Two Integers

- This program asks to input two integers, then adds them together and displays the result. It uses cin to take input and cout to print the sum with a clear message.

```
#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 */
}
```

| | | | |
|--|---|------------|--|
| main.cpp |    | Run | Output |
| <pre> 1 #include <iostream> 2 3 int main() 4 { 5 int integer1; 6 int integer2; 7 int sum = integer1 + integer2; 8 std::cout<<"Enter first integer:\n"; 9 std::cin>>integer1; 10 std::cout<<"Enter second integer:\n"; 11 std::cin>>integer2; 12 sum = integer1 + integer2; 13 std::cout << "Sum is " << sum <<std::endl; 14 15 } 16 </pre> | | | <pre> Enter first integer: 13 Enter second integer: 13 Sum is 26 </pre> <p style="text-align: right;">==== Code Execution Successful ===</p> |

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;
}

```

The screenshot shows a web-based C++ compiler interface. The code in the editor is:

```

main.cpp
1 #include<iostream>
2
3 int main (void)
4 {
5
6     int xValue= 5;
7     int yValue=9;
8     int result;
9     int bigResult;
10
11    xValue += 3;
12    yValue -= xValue;
13    result = xValue * yValue;
14    result += result;
15    result -= 1;
16    yValue = result % result;
17    result += result + xValue;
18    bigResult = result * result * result;
19    result += xValue * yValue;
20
21    std::cout<<"result: "<< bigResult;
22
23    return 0;
24
25 }
26

```

The output window displays:

```

result: 54872
== Code Execution Successful ==

```

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 */

cout<<"After first year: "<<firstYearValue;
cout<<"After second year: "<<secondYearValue; cout<<"After third year: "<<thirdYearValue;
return 0;
}

```

Example output

After first year: 101.500000

After second year: 103.022499

After third year: 104.567833

The screenshot shows a web-based C++ compiler interface. The code in the editor is:

```

main.cpp
1 #include<iostream>
2 int main()
3 {
4     float startValue = 100;
5     float interestRate = 0.015;
6     float firstYearValue;
7     float secondYearValue;
8     float thirdYearValue;
9
10    firstYearValue = startValue * (1 + interestRate);
11    secondYearValue = firstYearValue * (1 + interestRate);
12    thirdYearValue = secondYearValue * (1 + interestRate);
13
14    std::cout<<"After first year: "<<firstYearValue << std::endl;
15    std::cout<<"After second year: "<<secondYearValue << std::endl;
16    std::cout<<"After third year: "<<thirdYearValue << std::endl;
17
18 }

```

The output window shows the results of the execution:

```

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

== Code Execution Successful ==

```

The interface includes a toolbar with file, edit, run, and share buttons, and a sidebar for premium courses.

8. Conclusion

I now know and have learned that when you combine symbols, characters, operators, and other elements, you can create something interesting like math solutions in C++. Even when working with percentages, you. You can still solve problems easily as long as you understand how to use these parts properly. What's important is knowing and learning how each one works and where to place them in your code. It may look hard at first, but once you get the hang of it, solving math problems using code becomes fun and exciting.

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 pts |
| Ⓑ 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 pts |
| Ⓒ 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 pts |
| Ⓓ 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 pts |

Total Points: 24