

EECE 1080C / Programming for ECE

Summer 2019

Laboratory 1: Getting Started with C++

Topics covered:

- Integrated Development Environment (IDE)
- Compiling
- Debugging
- C++ program development practice

Objective:

- To practice program development by using the IDE. The student will create several short programming assignments to create a basic understanding of C++.

Highlights:

- To receive full credit for this laboratory please sign the attendance sheet.
- Please access the laboratory assignment via the canopy/blackboard link. The descriptions for each problem are contained within this document.
- **Submit on Blackboard using the assignment dropbox**

Grading:

- Each part should be worked on separately. You will need a separate project for each part of this assignment when working within your IDE.

Rubric: 100 points

- Part A = 10 points
- Part B = 10 points
- Part C = 15 points
- Part D = 15 points
- Part E = 15 points
- Part F = 15 points
- Part G = 20 points

Tasks:

A. Hello World Modified

- You will need to modify the following Hello World program to take any name as input. Once you have the name, you will output “hello” and said name.
- For example, if the user enters “Bob” the program should output “Hello Bob!” plus a newline (endl).
- For each of the following steps, end the program’s output with a newline.
- Modify the given program to input a string for a name and output that string along with “hello”.
- **Example:**

Enter Name: Bob

Hello Bob!



```
1 #include <iostream>
2 #include <string>
3 using namespace std;
4
5 int main() {
6
7     string helloworld = "Hello World!";
8     /* Type your code here. */
9     // This program outputs Hello World
10    // Modify it to Input a name and output "Hello" plus that name
11
12    cout << helloworld << endl;
13
14    return 0;
15 }
```

Figure 1. Example Hello World code.

B. Rectangle area and perimeter

- Create a program that inputs two variables (width and height) of a rectangle and outputs the area and the perimeter of said rectangle.

- **Example:**

Width: 4

Height: 5

Area: 20

Perimeter: 18

- Note that the input is in **bold** above.
- Note that inputs (4 and 5) do not need to be displayed.



```
1 #include <iostream>
2 using namespace std;
3
4 int main() {
5
6     /* Type your code here. */
7
8     return 0;
9 }
```

Figure 2. Template of basic initial code.

C. Right circular cone

- Your program should receive as inputs the height and radius of a cone and calculate the volume and the total surface area of said cone.

- **Example 1:**

Radius: 2.5

Height: 3.4

Surface area: 52.78

Volume: 22.25

- **Example 2:**

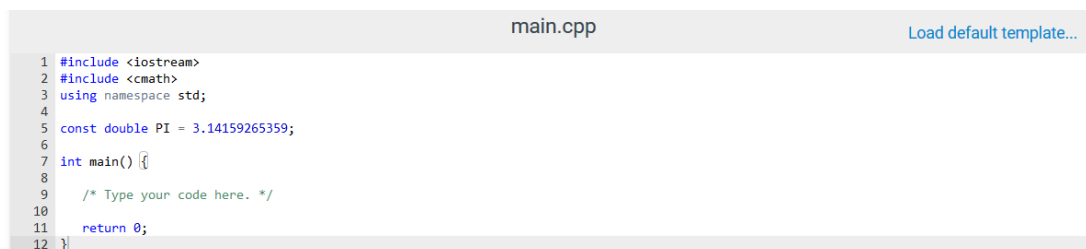
Radius: 1

Height: 1

Surface area: 7.58

Volume: 1.05

- Note that the input is in **bold** above.



```
main.cpp Load default template...
1 #include <iostream>
2 #include <cmath>
3 using namespace std;
4
5 const double PI = 3.14159265359;
6
7 int main() {
8
9     /* Type your code here. */
10
11     return 0;
12 }
```

Figure 3. Example of initial code for Part C.

D. Cube

- Create a program that takes in the edge length of a cube as input and outputs the total surface area and volume of said cube.

- **Example 1:**

Enter edge length: 7

Volume: 343

Surface area: 294

- **Example 2:**

Enter edge length: 1

Volume: 1

Surface area: 6

- Note that the input is in **bold** above.
- Refer to **Figure 2** to start coding.

E. Basic output with variables

- Create a program that takes in the proper inputs and determines the outputs according to the following functionality.
- Note that for convenience in the examples below, the user's input value is shown on the next line. However, such values don't actually appear as outputs when the program runs.
- A variable like "userNum" can store a value like an integer. Extend the program to print "userNum" values as indicated:
 - Output the user's input

```
Enter integer:
```

4

```
You entered: 4
```

- Extend to display the input squared and cubed.

```
Enter integer:
```

4

```
You entered: 4
```

```
4 squared is 16
```

```
And 4 cubed is 64!!
```

- Extend to get a second user input into userNum2. Output the sum and product.

```
Enter integer:
```

4

```
You entered: 4
```

```
4 squared is 16
```

```
And 4 cubed is 64!!
```

```
Enter another integer:
```

5

```
4 + 5 is 9
```

```
4 * 5 is 20
```

- Note that the input is in **bold** above.

F. Grading calculator

- You will be creating a grading calculator that determines the final score of a student.
- The input is expected to be:
 - Three homework (worth 30% total)
 - Three quizzes (worth 10% total)
 - Two midterms (worth 15% each)
 - One final (worth 30% total)
- **Example:**

Homework 1: 100

Homework 2: 0

Homework 3: 75

Quiz 1: 100

Quiz 2: 75

Quiz 3: 0

Midterm 1: 75

Midterm 2: 50

Final Exam: 100

Final Score: 72.0833

- Note that all the **bolded** numbers are inputs to the program. You don't need to display those. They will be displayed automatically.



```
main.cpp Load default template...
1 #include <iostream>
2
3 using namespace std;
4
5 int main() {
6     // Some variables to get you going
7     double homework1 = 0, homework2 = 0, homework3 = 0;
8     double exam1 = 0;
9
10    cout << "Homework 1: ";
11    cin >> homework1;
12
13    return 0;
14 }
```

Figure 4. Example of initial code for Part F.

G. Suspension Test – Code Translation

- On Blackboard, you are given a code written in MATLAB: SuspensionTest.m
- Translate the code into a C++ format
- Note: replace the menu statements with cout/cin commands
- Note: randi doesn't exist in C++. Use the following translation

randi(2)	becomes	rand()%2+1
randi(1000)	becomes	rand()%1000+1
- Refer to **Figure 2** to start coding.
- **Test for a Small Car with a tolerance of 0.15**