Activity No. 6.2	
Built-in Functions	
Course Code: CPE 007	**Program:** Computer Engineering
Course Title: Programming Logic and Design	**Date Performed: 10/27/2025**
Section: CPE11S1	**Date Submitted: 10/30/2025**
Name(s): Canoy Hail B.	**Instructor: Engr. Jimlord M. Quejado**

6. Output

7. Supplementary Activity

1. Create a program that defines a function to compute for the volume of a cube. The formula of the volume of the cube is given as V = s * s * s.

Code :

```
#include <iostream>
using namespace std;

void greetUser();
int volumeComp(int sides);

int main (){
    int s;

    greetUser();

    cout << " Enter the Value of the Sides : ";
    cin >> s;

    cout << " The Volume of the Cube is : " << volumeComp(s);
    return 0;
}

void greetUser(){
    cout << "- - - - - Welcome to Cube Volume Computer - - - - -" << endl;
}
int volumeComp(int sides){
        return sides * sides * sides;
}
```

Output :

```
- - - - - Welcome to Cube Volume Computer - - - - -
 Enter the Value of the Sides : 90
 The Volume of the Cube is : 729000
--------------------------------
Process exited after 3.845 seconds with return value 0
Press any key to continue . . .
```

Analysis :

- In this code I start with include iostream, using namespace std then void greet user so I can input a welcome message for later, then do the volumeComp for the computation later with the int sides. Then moving on in int

main we have the int s to represent the sides, greet user so it will display the greetings in the output, then I cout the guide on what should the user should do then I used cin so the user can enter a value then it will put the s then the cout of results to display it which is the result of volumeComp. Then our last step is the cout for the greetuser for the greet user to have the value and print it then finally the computation of the volumeComp which is side multiplied by itself 3x.

1. Define a function hypotenuse that calculates the length of the hypotenuse of a right triangle when the other two sides are given. Use this function in a program to determine the length of the hypotenuse for each of the following triangles. The function takes two arguments of type double and return the hypotenuse as a double.

Code :

```
#include <iostream>
#include <cmath>
using namespace std;

void greetUser();
double hypotenuse(double side1, double side2);

int main(){
    double s1, s2;
    greetUser();

    cout << "Enter Side 1 : ";
    cin >> s1;

    cout << "Enter Side 2 : ";
    cin >> s2;

    cout << "The Hypotenuse is : " << hypotenuse(s1,s2);
    return 0;
}

void greetUser(){
    cout << "- - - - - Welcome to the Hypotenuse Calculator - - - - -" << endl;
}
double hypotenuse(double side1, double side2){
    return sqrt((side1 * side1) + (side2 * side2));
}
```

Output :

```
- - - - - Welcome to the Hypotenuse Calculator - - - - -
Enter Side 1 : 76
Enter Side 2 : 23
The Hypotenuse is : 79.404
--------------------------------
Process exited after 5.697 seconds with return value 0
Press any key to continue . . .
```

Analysis :

- This one is pretty similar to the first one the first code I did I just added one thing and that's cmath for me to be able to use the function sqrt after that I added the void for greetings then the double hypotenuse then double side1 and side2 for it to display 2 decimal point number. Then added the int main, then double s1 and s2 to

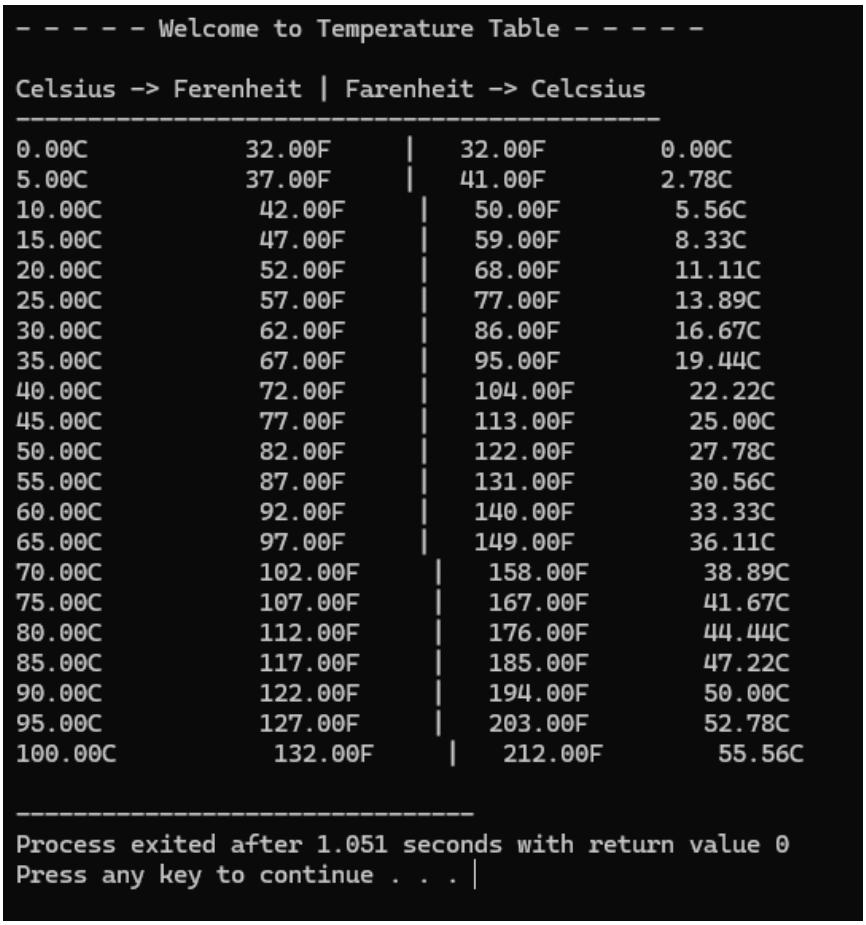
represent side 1 and 2. then I cout the guide for the user to input then cin both s1 and s2 for the value to be kept in s1 and s2 then cout another one for the result then finally void with cout for us to display the greetings and the last one is computation for the hypotenuse which is the square root of the total of side 1 multiplied by itself plus side 2 also multiplied by itself.

1. Implement the following integer functions:
    1. Function celsius returns the Celsius equivalent of a Fahrenheit temperature.
    2. Function fahrenheit returns the Fahrenheit equivalent of a Celsius temperature.
    3. Use these functions to write a program that prints charts showing the Fahrenheit equivalents of a Celsius temperatures from 0 to 100 degrees, and the Celsius equivalents of all Fahrenheit temperatures from 32 to 212 degrees. Print the outputs in a neat tabular format that minimizes the number of lines of output while remaining readable.

Code :

```cpp
#include <iostream>
using namespace std;
#include <iomanip>

void greetUser();
void userChart();
double celsiusFormula(double c);
double farenheitFormula(double f);

int main(){

greetUser();
userChart();

    return 0;
}

void greetUser(){
    cout << "- - - - - Welcome to Temperature Table - - - - -" << endl;
}
void userChart(){
    cout << endl << "Celsius -> Ferenheit | Farenheit -> Celcsius" << endl;
    cout <<  "-------------------------------------------" << endl;
    cout << fixed << setprecision(2);

    double c = 0;
    double f = 32;

    while(c <= 100 && f <= 212){
        cout << c <<"C          "<< f <<"F     |   "<< celsiusFormula(c) <<"F        " << farenheitFormula(f) <<"C"<< endl;
        c += 5;
        f += 5;
    }
}
double celsiusFormula(double c){
    return (c * 9 / 5 ) + 32;
}
double farenheitFormula(double f){
    return (f - 32) * 5 / 9;
}
```

Output :



Analysis :

- So in this code I still used the same things I used in the previews but added iomanip so I can use fixed << setprecision then immediately did the void greetUser and user chart for later, and celsius, fahrenheit formula. after that the int main which only has greetUser and userChart. Then the function for greetings greetuser then “welcome to temperature table” after that the longest function in this code userChart I added the header celsius to fahrenheit, fahrenheit to celsius then the lines, and by using the include iomanip I was able to use fixed << precision(2) so I can set the include decimal point at only 2 numbers. then the double c and f so it can start from their perspective numbers which are the 0 and 32, then looped it that it shouldn't go over 100 and 212 then inside that loop it will print c and f added by 5 because of +=5 then cout the celsius and fahrenheit formula that will increase overtime. Then the last one is added the formula values inside the fahrenheit and celsius formula.

8. Conclusion

- In this activity I learned how to apply the function I learned in class effectively. I also learned more about things like cmath, iomanip. This activity really helped me hone and improve my skills in programming. In the first code it was kinda easy for me but I still had errors but overall I did it with lightwork the second one made me struggle a bit but I still overcame it. The last one was the hardest. It took me a while to do it, days. But I still did it. I just lack knowledge on how to do the loop, and organize it. I keep getting errors and the code wont work but overtime I fixed and improved it. Overall I did well in this task and is actually improving, on the first few days of coding I wasn't like this although its a small improvements and I still have a long way to go I did improve and that's what matters, I just need to keep practicing to keep on improving, not to be perfect but to be better.