**Title**: Exercise 5\_2 – Default parameters

**Objective**: Allow the student to further explore default parameters for methods

**Statement of Problem**: Modify the GetMaximumValue method in the program below to take a default parameter. The default for the parameter should be 1,000,000. Update the method to check for a value < 1,000,000 for this new parameter. If it is, meaning the method was called with a fourth argument, then the method should consider this value in the maximum calculation. Otherwise, the value of the fourth parameter should be ignored. That is, 1,000,000 should NOT be displayed as the maximum value.

Then, run the program again to show that there were no updates in main required to call the same method the same way as before. You should get the same results.

Finally, update main to take a fourth value from the user. Call the GetMaximumValue method with all 4 user values, and show that all 4 values are taken into consideration in the determination of maximum value. Note that in order for this program to run correctly, user input needs to normally be < 1,000,000.

Upload your program, and a screenshot of the program output.

///////////////////////////////////////////////////////////

// CIS554 Object Oriented Programming in C++

// Exercise 5\_2 – Default parameters

///////////////////////////////////////////////////////////

#include <iostream>

using std::cout;

using std::cin;

using std::endl;

class Maximum

{

public:

int GetMaximumValue(int num1, int num2, int num3)

{

int maxValue = num1;

if (num2 > maxValue)

maxValue = num2;

if (num3 > maxValue)

maxValue = num3;

return maxValue;

}

};

int main()

{

int num1;

int num2;

int num3;

Maximum max;

cout << "Enter first number: ";

cin >> num1;

cout << "Enter second number: ";

cin >> num2;

cout << "Enter third number: ";

cin >> num3;

cout << "Max Value = " << max.GetMaximumValue(num1, num2, num3) << endl;

return 0;

}