Complete ALL questions below. Only partial question(s) will be graded. Download the main.cpp from Moodle and begin your work. You may use <cmath> and <algorithm> libraries to help.

1. Write a function named "g\_c\_d" that takes two positive integer arguments and returns as its value the greatest common divisor of those two integers. If the function is passed an argument that is not greater than zero, then the function should return the value 0 as a sentinel value to indicate that an error occurred.

Thus, for example,

cout << g\_c\_d(40,50) << endl; // will print 10

cout << g\_c\_d(256,625) << endl; // will print 1

cout << g\_c\_d(42,6) << endl; // will print 6

cout << g\_c\_d(0,32) << endl; // will print 0 (even though 32 is the g.c.d.)

cout << g\_c\_d(10,-6) << endl; // will print 0 (even though 2 is the g.c.d.)

1. A positive integer n is said to be prime (or, "a prime") if and only if n is greater than 1 and is divisible only by 1 and n . For example, the integers 17 and 29 are prime, but 1 and 38 are not prime. Write a function named "is\_prime" that takes a positive integer argument and returns as its value the integer 1 if the argument is prime and returns the integer 0 otherwise.

Thus, for example,

cout << is\_prime(19) << endl; // will print 1

cout << is\_prime(1) << endl; // will print 0

cout << is\_prime(51) << endl; // will print 0

cout << is\_prime(-13) << endl; // will print 0

1. Write a function named "swap\_floats" that takes two floating point arguments and interchanges the values that are stored in those arguments. The function should return no value. To take an example, if the following code fragment is executed:

float x = 5.8, y = 0.9;

swap\_floats (x, y);

cout << x << " " << y << endl; //then the output will be 0.9 5.8

1. Write a function named "sum\_from\_to" that takes two integer arguments, call them "first" and "last", and returns as its value the sum of all the integers between first and last inclusive.

Thus, for example,

cout << sum\_from\_to(4,7) << endl; // will print 22 because 4+5+6+7 = 22

cout << sum\_from\_to(-3,1) << endl; // will print -5 'cause (-3)+(-2)+(-1)+0+1 = -5

cout << sum\_from\_to(7,4) << endl; // will print 22 because 7+6+5+4 = 22

cout << sum\_from\_to(9,9) << endl; // will print 9

**Submission**

Submit your work to Moodle dropbox before deadline stated on Moodle. Do not compress your files.