**Name: Session:**

**Programming I**

**Lab Exercise 12.2.2019**

For each of the following programs, print out your documented source code and attach it to this sheet and turn in.

1. Write a function that takes one integer named size as a parameter and prints an equilateral triangle composed of asterisks of length size. For example, the call  make\_triangle(4) should result in the following triangle printed:

\*

\* \*

\* \* \*

\* \* \* \*

1. A Pythagorean triple is a set of integers (a, b, and c) such that

a2 + b2 = c2

For example, 3, 4, and 5 for a Pythagorean Triple.

Write a program that has a function that will create and return a list of Pythagorean Triples given a maximum number size.

def generateTriples(num):

triples = [ ]

//insert code here

return triples

1. If you are given three sticks, you may or may not be able to arrange them in a triangle. For example, if one of the sticks is 12 inches long and the other two are one inch long, it is clear that you will not be able to get the short sticks to meet in the middle. For any three lengths, there is a simple test to see if it is possible to form a triangle:

If any of the three lengths is greater than the sum of the other two, then you cannot form a triangle. Otherwise, you can.

Write a function named is\_triangle that takes three integers as arguments, and that returns True or False, depending on whether you can or cannot form a triangle from sticks with the given lengths.

1. Write a function that will return the area of any triangle. The function should be given 3 sides and return the area of that triangle. Use the is\_triangle function to determine if a triangle can be created.

def calcArea(a, b, c):

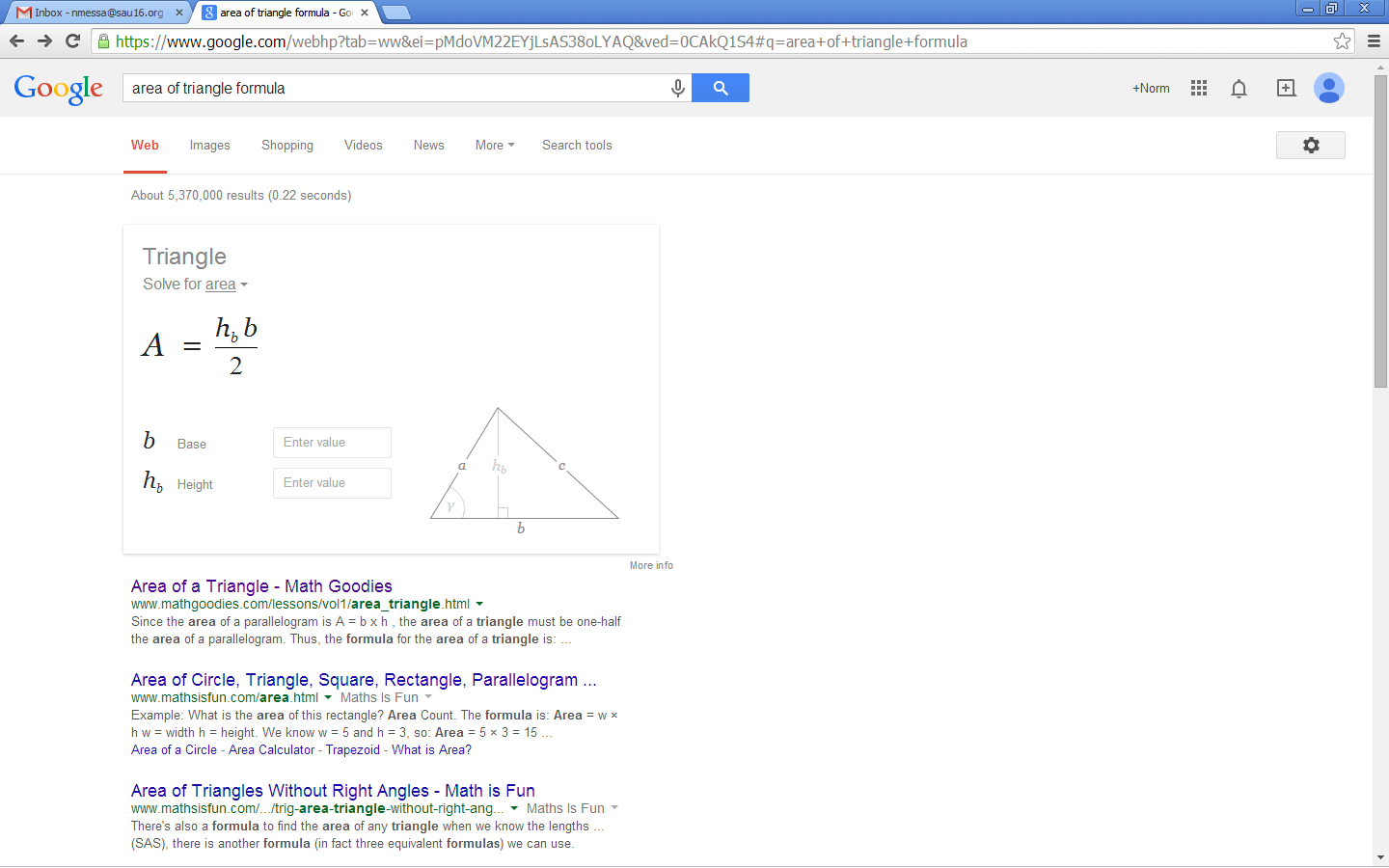
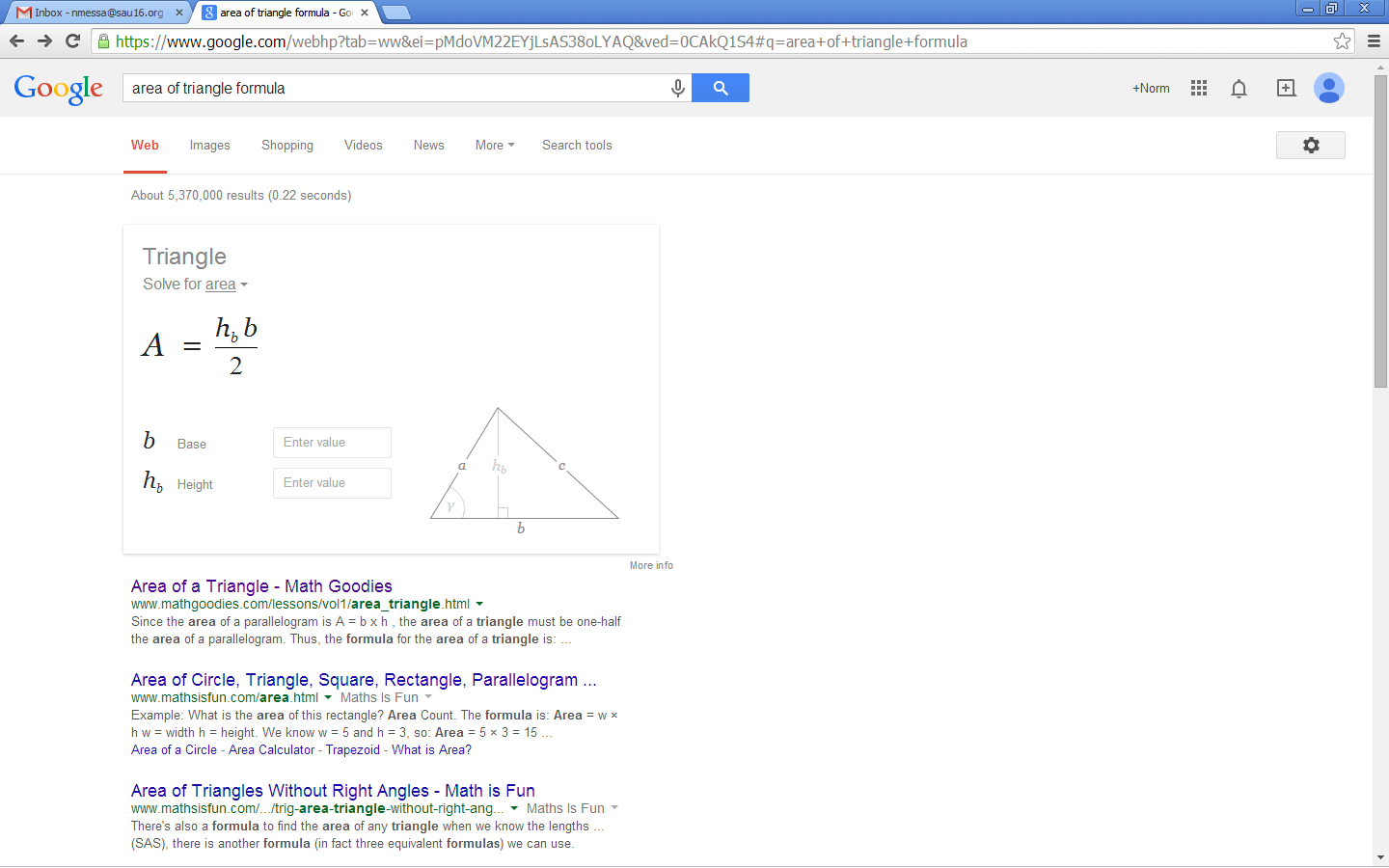
#add code here

return area

Hint: You can use Heron’s Formula to calculate the area. s is the semi-perimeter which is the sum of the sides divided by 2.

A = \sqrt{s(s-a)(s-b)(s-c)},

You could also use

but that is not that easy if you have don’t have a right triangle where one of the sides is the height.

1. Write a function that will print Pascal’s Triangle. The function will receive the parameter of the level of the triangle.

def genPascal (level):

#Add code here

For example genPascal(4) will create the following:

1

1 1

1 2 1

1 3 3 1

1 4 6 4 1

Note: This is equivalent to  