**Assignment 3 –** *Practice: working with functions, for loops, Numbers & turtles*

**Name: \_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_ Student number:** \_\_ \_\_ \_\_ \_\_ \_\_ \_\_

**Name: \_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_ Student number:** \_\_ \_\_ \_\_ \_\_ \_\_ \_\_

***ASSIGNMENT MARK*: \_\_\_\_\_\_\_\_ / 50 Due Date: Friday, Mar. 9th**

As always, all your answers should use functions and every function should have a documentation string explaining the purpose of the function.

**Do not wait until the last minute to do this assignment in case you run into problems.**

Submit your assignment to the instructor by email at: drahmalki@gmail.com

**Part 1 - Simple Function [15 points]**

Write a Python function that asks the user for an integer n. Using n, print a pattern like the one below. n will be the number of lines printed, so the example below is for n=8.

Notice that the middle number on line k is 2k-1 and the middle number on line 1 is 20 =1, and the middle number in line 8 is 27 = 128. – see below.

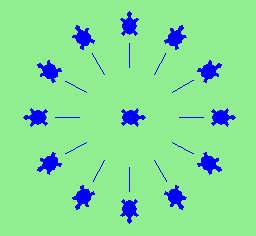


**Part 2 – Simple drawing with turtle [20 points]**

[1] Use for loops to make a turtle, draw these regular polygons (regular means all sides have the same lengths, and all angles are the same): **[6 points, 3 each]**

* An equilateral triangle
* A hexagon (six sides)

[2] Use for loops to make a turtle, draw a clock that looks something like this: **[8 points]**



turtle — Turtle graphics <https://docs.python.org/3/library/turtle.html>

**HINT: start your program as follows:**

import turtle # import the turtle module

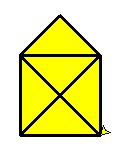
scr = turtle.Screen()

turt = turtle.Turtle()

# put code here to move turtle called turt.

turtle.mainloop() # this must be the last line in code.

[3] Use for loops to make a turtle, draw a yellow colored house as in the following picture. You will write a program to draw a little house, using a turtle. The house can be drawn **without picking up the pen** (i.e., in graph terminology, we can draw the house using an Euler Path). **[6 points]**



**Part 3 – Written [15 points]**

Answer the following questions in this document with same name.  
  
**1.** Convert these binary numbers into decimal assuming they are unsigned integers. Show your calculations. [4 points]

**a.** 1001   
**b.** 011011   
**c.** 01011010   
**d.** 10000000 01111111

**2.** Convert the same binary numbers into decimal assuming they are using signed magnitude representation. Show your calculations. [4 points]  
第一位是正負號 不用轉換  
**3.** Convert these binary numbers into decimal assuming they are using two's complement representation. Show your calculations. [4 points]

4. Subtract each, as a computer would, using binary code and registers of size 8. Show all steps involved in the process. [3 points]

a. 26 -15

b. -31 – 6

**Happy computing!**

**N.B:** Next, you are going to make sure you can use the submission properly. This will be how you turn in your assignments electronically. The submission procedure only accepts **zip files (NOT rar Files)**, so you are going to create a “Asg3\_YourFirstName\_YourStudentID.zip” file containing all the programs you just wrote as well as this document with your name and student number added to it. To create a “.zip” file, find the directory where the files are saved and select all of them. If you right-click on one of the selected files, you should have the option to create an archive by choosing “Send to” then “Compressed “zipped” Folder”. Rename the created “.zip” file and give it the name Asg3\_YourFirstName\_YourStudentID.zip.

***10 marks will be deducted if:***

1. ***the email did not have a full name and a student # in the subject field or***
2. ***the procedure for zipped file is not followed or***
3. ***Python template and documentation are missing or***
4. ***This document, with your names and student numbers on it, is not included with the zipped file***
5. ***Your classmate’s email is not included in CC or Bcc field of the email***

**A score of zero (0) will be given to everyone whose email is empty.**