CMPT 103 – Lab #2

General Information

Python version and IDE:	Python 3 / Wing IDE 101
Allocated lab time:	2 hrs and 50 min
Due date:	At the end of the lab period
Lab weight:	3%

Topics

- ✓ Functions, Looping, and Printing

Submission

- ✓ **All the code files (.py) should be submitted electronically** to your Lab Blackboard site.
- ✓ A portion of the total marks (20%) will be allocated for the programming style. For example, functions should be small; avoid writing duplicate code; names should be meaningful and descriptive; naming convention should be followed consistently; code should be formatted properly; and comments should be accurate and well written.
- ✓ Comments are **required** for:
  - EACH program indicating the student name and program name.
  - EACH function indicating the function purpose, syntax (example usage of the function), parameters, and return value.
  - Any block of code for which the purpose may be unclear (Note: you should always try to write clean code that can be understood easily without comments).

Assignment

For this lab, please put all functions into a file called `Lab2your_initials.py` (e.g., `Lab2FL.py` where F and L are the first letter of your first name and last name).

1) [10 marks] Fill in the blanks with the missing Python code to produce the output sequences shown below. **Do not modify the existing code!** Use the Python shell in Wing 101 to experiment. Put your answers into functions named **`loop_a`** (for part a) and **`loop_b`** (for part b). For this question, you don't need to write any comments about the functions.

Part (a):	Part(b):
>>> for i in range _____ : ... print(i) 20 23 26 29 32	>>> for i in range _____ : ... print(i) 7 5 3 1 -1

2) [40 marks] Write a function named **`hollow_diamond`** that prints a hollow diamond as shown below. Your function is expected to work with only even **`width`** parameters. Assume your code will be tested only on positive even integers. Feel free to write additional helper functions to break down the complexity of your solution.

Purpose:	Print a hollow diamond centered within a square of the specified width
Syntax:	**`hollow_diamond(width)`**
Parameter:	**`width`**: an even integer indicating the width of the square surrounding the diamond
Return value:	None

```
>>> hollow_diamond(8)
********
***  ***
**    **
*      *
*      *
**    **
***  ***
********
```

```
>>> hollow_diamond(6)
******
**  **
*    *
*    *
**  **
******
```

3) [50 marks] Write a function named **`full_diamond`** that prints a full diamond as shown below. Your function is expected to work with only odd **`width`** parameters. Assume your code will be tested only on positive odd integers between 3 and 17. Feel free to write additional helper functions to break down the complexity of your solution.

Purpose:	Print a full diamond with numbers
Syntax:	**`full_diamond(width)`**
Parameter:	**`width`**: an odd integer representing the widest part of the diamond
Return value:	None

```
>>> full_diamond(5)
  1
 212
32123
 212
  1
>>>
>>> full_diamond(7)
   1
  212
 32123
4321234
 32123
  212
   1
```