DSC 430: Python Programming
Assignment 0401: Goldbach Deuce

Given a list of random positive integers, do any two of them sum to a given number? You solved this problem in last week's Goldbach's Conjecture assignment. However, I bet your running time was O(n^2). If not, great! This assignment will be a snap. For the rest of you…

Ask the user for the number of random integers (say 'n') and the target (say 'sum'). Create 'n' number of random integers between 1 and 100 and store them in a list. Then determine if any two of the numbers add up to 'sum' in O(n.log(n)) time. Output should be any one pair of items (in the random list) that add up to 'sum' if exists, or "No numbers sum up to __" (where "___" is the target sum) otherwise.

Hint: **Binary Search**.

Warning: look-ups in dictionaries and lists are not free.

Concession: I will let you sort the list using Python's built in function as a preprocessing step, not counted toward the efficiency of your algorithm. Besides, merge sort runs in O(n.log(n)).

Ensure to provide a sufficient amount of documentation/comments in the code. Points of particular importance are:

- Explain how your approach works.
- Explain why your code runs in O(n.log(n)).

Submission: Submit the source file (.ipynb) and the exported html file (.html) to the D2L. Do not zip or archive the file. Your code must include comments at the top including your name, assignment number, and the honor statement, "I have not given or received any unauthorized assistance on this assignment." Also, each function must include a docstring.