$O(n \log n)$ *vs.* $O(n^2)$

CSCI 112, Lab 7

File names: Names of files, functions, and variables, when specified, must be EXACTLY as specified. This includes simple mistakes such as capitalization.

Individual work: All work must be your own. Do not share code with anyone other than the instructor and teaching assistants. This includes looking over shoulders at screens with the code open. You may discuss ideas, algorithms, approaches, *etc.* with other students but NEVER actual code. Do not use code written by anyone else, in the class or from the internet.

Documentation: Each file should begin with a docstring that includes your name, the class number and name, the lab number, and a short description of the lab, as well as documentation pertinent to that particular file.

The project: Many of the $O(n^2)$ sorts are faster than the $O(n \log n)$ sorts when they array is small because they have less overhead. Also they can be faster when only a few points are out of order, because they can sort in just a few passes. If this is less than $\log n$, then it can be an improvement.

Develop a series of tests ??????????????

The project:

File names: Call your module `tbst.py` and your unit test module `tbst_test.py`, place in a folder `csci112lab07yourname` zip and turn in to canvas.