Project 1 - Source Code (Phase 1)

Due Apr 22 by 11pm **Points** 180 **Submitting** on paper **Available** after Apr 4 at 11am

CS 165 Project #1 — Analyzing Sorting Algorithms (Phase 1)

Project 1 involves implementing and testing various sorting algorithms experimentally to determine their real-world running times. In particular, for Phase 1 you will need to implement each of the following sorting algorithms:

1. **Insertion-sort**
2. **Merge-sort**
3. **The following implementations of Shellsort:**
    1. The original Shell sequence, $[n/2^k]$, ..., 1, for k=1,2,...,log n, where [*] denotes the floor function.
    2. The **A083318** (https://oeis.org/A083318) sequence, $2^k + 1$, for k=log n, ..., 3, 2, 1, plus the value 1.
    3. The **A003586** (https://oeis.org/A003586) sequence, 2^p3^q, ordered from the largest such number less than n down to 1.
    4. The **A033622** (https://oeis.org/A033622) sequence, in reverse order, starting from the largest value less than n, down to 1.
4. **The following implementations of Hybrid sort** (using merge-sort and insertion-sort):
    1. $\boldsymbol{H} = \boldsymbol{n}^{1/2}$
    2. $\boldsymbol{H} = \boldsymbol{n}^{1/4}$
    3. $\boldsymbol{H} = \boldsymbol{n}^{1/6}$

For Phase 1, you need to correctly implement each of the above algorithms in **Python 3.6+**. Phase 2 will involve you performing experiments with this software.

Upload all your source code (but not test data) via **GradeScope** to turn it in and run test cases. The code for each algorithm should be in a different file, as shown below. You may alternatively create and upload one big zip file containing all these files in the top level.

- **insertion_sort.py (https://canvas.eee.uci.edu/courses/55229/files/22253585?wrap=1) (https://canvas.eee.uci.edu/courses/55229/files/22253585/download?download_frd=1)** (provided as an example)
- merge_sort.py

- shell_sort1.py
- shell_sort2.py
- shell_sort3.py
- shell_sort4.py
- hybrid_sort1.py
- hybrid_sort2.py
- hybrid_sort3.py

You should be using and turning in the file **requirements.py (https://canvas.eee.uci.edu/courses/55229/files/22253584?wrap=1) (https://canvas.eee.uci.edu/courses/55229/files/22253584/download?download_frd=1)** , which contains more detailed information. You should eventually also have a file, main.py, that runs all your experiments, but that does not have to be completely done in Phase 1. You may also create other files, but you must submit all your Phase-1 files to Gradescope to be graded.