CS 340 Project I: Sorting (and Heaps)

Description: You are to implement **InsertionSort, MergeSort, HeapSort**, and the linear time **BuildHeap** procedure, and plot all four on 5 sorted and 5 permuted lists.

Specifications: I have provided you with the said subsets of the wordlists, namely 5 subsets of the small permuted wordlist and 5 of the small sorted wordlist. The permuted input files are of the form **perm<size>.txt** where the size indicates how many words (lines) the file has, and the sorted input files are **sorted<size>.txt**. E.g., **Perm30K.txt has 30,000 words, sorted60K.txt has 60,000 words, etc..** You should run each of the three algorithms, **InsertionSort, MergeSort, and HeapSort,** on all 10 input files, both the already sorted inputs and the permuted inputs. Moreover, you will also be measuring the runtime of **BuildHeap** separately, in addition to the total runtime of **HeapSort,** and the runtimes of **InsertionSort** and **MergeSort.**

From the console, you will allow the user to select (i) the sorting algorithm, (ii) file size, (iii) whether the input file is permuted or already sorted. Upon that selection, you will run the appropriate algorithm on the selected input file and report to the user the location of the associated output file in addition to the execution time.

Plots: You will have two sets of plots, one for the permuted inputs and another for the already sorted ones, where you demonstrate the runtime of the four algorithms (InsertionSort, MergeSort, HeapSort, and BuildHeap) as a function of the input size.

What to turn in: You must turn in a single zipped file containing your **source code,** runtime **PLOTS**, a **README** file indicating how to execute your program, and a **Makefile** if needed for compilation. **Read proglag.pdf for further descriptions on allowable programming languages and platforms.**

This assignment is due by MIDNIGHT of Thursday, August 31st. Late submissions carry a -40% per-day late penalty (i.e. -40% for each day late).