EE3980 Algorithms

Homework 5. Better Sorts

Due: Apr. 10, 2021

It has been shown that `heap sort` (Algorithm 2.2.19), `merge sort` (Algorithm 3.2.1) and `quick sort` (Algorithm 3.2.5) have better performances than other sorts. In this homeowrk, please implement these three sorting algorithms in `C` and compare their efficiency using the data sets in hw01. The function declarations should be as following:

```
void HeapSort(char **list, int n);
void MergeSort(char **list, int low, int high);
void QuickSort(char **list, int low, int high);
```

As usual, you should analyze these algorithms for their space and time complexities and correlate the CPU times to the theoretical complexities.

Example of program output is as follows:

```
$ a.out < s1.dat
N = 10
HeapSort CPU time: 0.00132823 s
MergeSort CPU time: 0.00147581 s
QuickSort CPU time: 0.00106192 s
1 anemometry
2 cates
3 cincture
4 homebuilder
5 preestablish
6 roccellaceae
7 seedbed
8 speedboat
9 synclinal
10 unamusing
```

Notes.

1. One executable and error-free `C` source file should be turned in. This source file should be named as `hw05.c`.
2. A report file in `pdf` format is also needed. This file should be named as `hw05a.pdf`.
3. Submit your `hw05.c` and `hw05a.pdf` on EE workstations using the following command:

   `~ee3980/bin/submit hw05 hw05.c hw05a.pdf`

   where `hw05` indicates homework 5.
4. Your report should be clearly written such that I can understand it. The writing, including English grammar, is part of the grading criteria.
5. In comparing two strings, the following library function in the `<string.h>` package can be used.
   ```
   int strcmp(const char *s1, const char *s2);
   ```