EE3410 Algorithms

Homework 6. Better Sorts

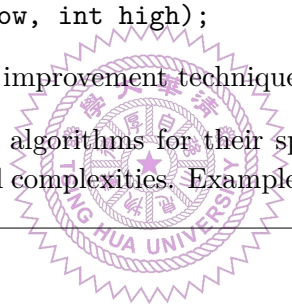
Due: Apr. 7, 2025

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 performance than other sorting algorithms. In addition, `merge sort` can be improved as shown in Algorithm 3.2.3, and the same technique can also be applied to `quick sort`. It has also been argued that randomized quick sort (Algorithm 3.2.7) can avoid worst-case complexity. In this homework, please implement these six 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 MergeSort1(char **list, int low, int high);
void QuickSort(char **list, int low, int high);
void QuickSort1(char **list, int low, int high);
void RQuickSort(char **list, int low, int high);
```

Note that `RQuickSort` applies both improvement techniques to the `QuickSort` algorithm.

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: 3.13997e-07 s
MergeSort CPU time: 2.85625e-07 s
MergeSort1 CPU time: 1.71423e-07 s
QuickSort CPU time: 2.41804e-07 s
QuickSort1 CPU time: 1.88828e-07 s
RQuickSort CPU time: 2.73991e-07 s
1 bandbox
2 dearest
3 eccrine
4 fixable
5 manhole
6 notably
7 stearic
8 pompous
9 stemmed
10 stirred
```

Notes.

1. One executable and error-free `C` source file should be turned in. This source file should be named as `hw06.c`.
2. A report file in `pdf` format is also needed. This file should be named as `hw06a.pdf`.
3. Submit your `hw06.c` and `hw06a.pdf` on EE workstations using the following command:

   ~ee3980/bin/submit hw06 hw06.c hw06a.pdf

   where `hw06` indicates homework 6.
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);
   ```