Work on project. Stage 3/4: Need for speed

198 users solved this problem. Latest completion was 10 days ago.

Project: Phone Book

Hard

Description

Let's use faster methods to sort the list of numbers and to search in the list.

As in the previous stage, you should first sort the list of phone numbers by the owner's name and then search for the numbers of the 500 people from the list given in the first stage. Remember: to get decent and comparable results, you should put all the algorithms in the same initial conditions.

For sorting, use the quick sort algorithm, and for searching, use the binary search algorithm.

4 / 4 Prerequisites

- Recursion basics — Stage 3 — 3
- Divide and conquer — Stage 3 — 3
- Binary search — Stage 3 — 1
- Quick sort — Stage 3 — 1

Example

Output all three approaches one after another and see which one is faster. The output example is shown below. Note that you can get totally different sorting and searching times!

```
Start searching (linear search)...
Found 500 / 500 entries. Time taken: 1 min. 56 sec. 328 ms.

Start searching (bubble sort + jump search)...
Found 500 / 500 entries. Time taken: 9 min. 15 sec. 291 ms.
Sorting time: 8 min. 45 sec. 251 ms.
Searching time: 0 min. 30 sec. 40 ms.

Start searching (quick sort + binary search)...

Found 500 / 500 entries. Time taken: 1 min. 21 sec. 996 ms.

Sorting time: 1 min. 17 sec. 381 ms.

Searching time: 0 min. 4 sec. 615 ms.
```

Report a typo

HINT by MG Ma Go

Passing output for the bubble + jump test can look something like this:

Start searching (bubble sort + jump search)...
Found 500 / 500 entries. Time taken: 01 min. 26 sec. 190 ms.
Sorting time - STOPPED, moved to linear search: 01 min. 18 sec. 967 ms.
Searching time: 00 min. 07 sec. 223 ms.

So you can reuse the previous bubble sort + jump search behavior if you update the timers and output for it. You can also use a smaller directory.txt if you want to actually see the jump search in action.

Requiring the total at the top of the output still seems weird (to me at least), but maybe it's just another one of the many gotchas this project has.

I think the point of this project is to just get a feel for these sorting and searching algorithms since no one is expected to re-invent the wheel here, and I guess someone wanted to make sure we knew just how bad bubble sort sucks.

Was this hint helpful? Yes No Report

This is the last hint available for this problem! Please post your own hint after completing the problem, future learners will thank you.

Write a program

Code Editor IDE

CONNECTION STATUS

✓ IDE is responding IntelliJ IDEA 2020.2.3

✓ EduTools plugin is responding 2021.2-2020.2-1924

Solve in IDE

View solution (100)

Comments (6) Hints (1) Useful links (1) Solutions (22) Show discussion