CPSC 319

Assignment 2

Matteo Messana

30020933

Q1. The 2 sorting algorithms used in the program are merge sort and quick sort algorithms. Both algorithms were provided to me by Michael T. Goodrich, Roberto Tamassia, Michael H. Goldwasser. The merge shift was used for the ordering of the individual words and quick sort was used to order all the letters in a word to compare for anagrams. The merge sort can be applied to files of any size so regardless of the size of the list of words, merge sort will be able to run. Merge sort minimizes the amount of seeking required for the algorithm, thus cutting down on some time and allowing it run at relatively similar speeds for larger files.

The quick sort algorithm was used for sorting the letters in each word in alphabetical order to determine any anagrams in the list. Quick Sort was chosen because of its quick processing time, hence the name. It is good for most scenarios except for when the array is almost already sorted.

Alternative algorithms that could have been used could include bubble sort, which would include more searching but would be much faster for arrays of smaller sizes.

Q2.

The big O for both sorting algorithms would be logarithmic since both algorithms use a halving method to sort the program. The total big O of the program would be f(n) = nlog(n).

|  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- |
| **QuickSort** | | | **Cost** | | **Times** | | **Comments** | |
| if(a >= b) | | | c1=2 | | t1=1 | | single assignment and return | |
| variable definition and instantiation | | | c2=5 | | t2=1 | | 3 assignments, 2 arithmetic operations | |
| while loops | | | c3=11 | | t3=log(L) | | 3 comparisons, 2 arithmetic operations, 2 method calls, 2 element accesses | |
| sorting adjustments | | | c4=10 | | t4=1 | | 5 assignments, 3 element access, 2 arithmetic expressions | |
| final placements | | | c5=5 | | t5=1 | | 3 array accesses, 3 assignments | |
| recursive calls | | | c6=4 | | t6=1 | | 2 method calls, 2 arithmetic expressions | |
|  | | |  | |  | |  | |
|  | | |  | |  | |  | |
| **MergeSort** | | | **Cost** | | **Times** | | **Comments** | |
| if(n<2) | | | c1=2 | | t1=1 | | single comparisson and return | |
| variable definition and instantiation | | | c2=7 | | t2=1 | | 3 assignments, 2 arithmetic operations, 2 method calls | |
| recursive calls | | | c3=3 | | t3=log(N) | | 3 method calls | |
|  | | |  | |  | |  | |
| f(n) | Growth Rate Analysis | | | | | Asymptotic Analysis | |
| Time Units | Prop to | | Rate | | Big Oh | |
| f(n)=3log(N)+9 | 3log(N)+9 | 3log(N) | | Logarithmic | | O(nlog(n)) | |
| f(n)=11log(L)+26 | 11log(L)+26 | 11log(L) | | Logarithmic | | O(nlog(n)) | |

Q3. If there are only 2 words the equation would be f(2) = 2log(2), which would be equal to 0.60206.

References

[1] Rashid, M. (2019). *CPSC 319: Tutorials*. [online] Pages.cpsc.ucalgary.ca. Available at: https://pages.cpsc.ucalgary.ca/~mdmamunur.rashid1/CPSC319-W19.html [Accessed 28 Feb. 2019].

[2] Govens, S. (2019). *Directory Contents*. [online] Pages.cpsc.ucalgary.ca. Available at:

http://pages.cpsc.ucalgary.ca/~sgovens/ [Accessed 28 Feb. 2019].