Comparison of MergeSort,QuickSort and Heapsort

Asymptotic running time of MergeSort TwoParts is T(n)= 2T(n/2) + O(n) and it is equal to O(n lgn) and running time of MergeSort ThreeParts is T(n) = 3T(n/3) + O(n) which is equal to O(n).So from these formulas and the comparison table below we could say that MergeSort ThreeParts takes more running time.

Asymptotic running time complexity for HeapSort is O(nlgn) in all three cases(worst,best,average.)

Fort the Quicksort algorithm the worst case is 0() and that would be equal to picking the pivot first element.Best case isO(nlgn) and that refers to chosing the pivot as the middle element of the array.

For all types of various array types and sizes the Quicksort algorithm and middle element takes the least running time from the 6 of selections from my comparison table.

|  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- |
|  | EQUAL INTEGERS | | | RANDOM INTEGERS | | |
|  | 1000 | 10000 | 100000 | 1000 | 10000 | 100000 |
| mergeSort  TwoParts | 777875 | 2099354 | 21239275 | 751801 | 2195750 | 10436724 |
| mergeSort  ThreeParts | 1084047 | 4519104 | 24451912 | 1465677 | 5136979 | 25563219 |
| heapsort | 237432 | 810666 | 4664486 | 829234 | 2308737 | 16694887 |
| quickSort  First | 7955346 | 28475214 | 2221596489 | 4143006 | 16003531 | 1509513467 |
| quickSort  Random | 5428534 | 20025252 | 1313157431 | 4269030 | 6270014 | 11487982 |
| quickSort  Middle | 385975 | 1377578 | 14771335 | 348444 | 2550119 | 8384777 |

|  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- |
|  | INCREASING INTEGERS | | | DECREASING INTEGERS | | |
|  | 1000 | 10000 | 100000 | 1000 | 10000 | 100000 |
| mergeSort  TwoParts | 195160 | 1259060 | 9793565 | 88494 | 12083734 | 14841656 |
| mergeSort  ThreeParts | 209777 | 1507553 | 31137926 | 123259 | 15049457 | 44446349 |
| heapsort | 86519 | 897974 | 9487393 | 157234 | 779061 | 15291630 |
| quickSort  First | 322370 | 15835629 | 1537849224 | 367802 | 15766493 | 1170459385 |
| quickSort  Random | 109432 | 649481 | 19142291 | 74271 | 743900 | 11601364 |
| quickSort  Middle | 44246 | 313678 | 1839009 | 56888 | 190815 | 7518803 |

Scenario A)

You have a Turkish-English dictionary with a single word for each word in alphabetical order and you want to translate it into an English-Turkish dictionary. For example; if your Tur-Eng dictionary contains [ayı : bear, bardak : glass, elma : apple, kitap : book] then your Eng-Tur dictionary should be [apple : elma, bear : ayı, book : kitap, glass : bardak]. If we think that there are thousands of words in your dictionary, which sorting algorithm do you use to do this translation faster?

Using heapsort sorting algorithm would be the best for this scenario.

Scenario B)

When you inquire Sub-Upper Pedigree, an ordered list of people according to their birth date comes out in the system of e-Devlet. However, you want to rank the people from the youngest to the oldest one. If you are asked to do this operation using a sorting algorithm, which algorithm do you use?

Since ranking people from youngest to oldest would be same as an array in increasing order.Using quicksort sorting algorithm and choosing the pivot as middle element would be the best way to do this operation as seen from the above comparison table of sorting algorithms.

LINKS

1. <http://www.cs.armstrong.edu/liang/intro9e/html/QuickSort.html>
2. https://jordanspencerwu.github.io/randomized-quick-sort/
3. http://www.algolist.net/Algorithms/Sorting/Quicksort
4. https://www.geeksforgeeks.org/3-way-merge-sort/