Divide and Conquer Analysis

Cody L. Strange

Utah Valley University

Divide and Conquer Analysis

This paper provides an in-depth analysis of two popular sorting algorithms Merge Sort and Quick Sort by comparing which algorithm is faster and more efficient, we will be using four case studies as evidence. Each of these case studies will be tested using both sorting algorithms, and with three different list sizes of size 210, 215, and 220. Each case study will resemble a real-life scenario and have differing distributions in their lists. We will run each test case ten times per list size and record the time it takes for each sorting algorithm to complete and record the average of the ten run times to determine which sorting algorithm is faster. We will also be recording the total number of basic operations – number of comparisons – required to complete the sorting algorithms and compare those to determine which sorting algorithm is more efficient.

Elo Simulation

Summary

The first case study will be sorting a list of players from an arbitrary competitive video game by their ELO. Where ELO represents their skill in said game.

Distribution

The list will be split into four different categories each with a lower bound and an upper bound, low(1000-2400), mid(2401-2700), high(2701-2900), and pro(2901-3000). Where 50% of the players are in the low category, 30% are in the mid, 15% are in the high, and 5% are in the pro.

Findings

Beginning with input size 2^{10} we found that quick sort beat merge sort by an average of approximately 2. $6.5*10^{-4}$ seconds. For input size 2^{15}, we found that quick sort once again beat merge sort by an average of approximately $2.34 *10^{-2}$. For input size 2^{20}, we found that merge sort beat quick sort by an average of approximately $4.4105 *10^{1}$ seconds. What this tells us is

that for this level distribution quick sort is negligibly faster at small list sizes while merge sort is significantly faster at larger list sizes.

Data.

Merge Sort(2^{10})

```
elo_sim merge_sort
        Run Times:
                (1) 0.003656300000000001
                (2) 0.0028258000000000033
                (3) 0.0027965000000000073
                (4) 0.002724199999999996
                (5) 0.0028799000000000047
                (6) 0.002860599999999991
                (7) 0.0026248000000000105
                (8) 0.003576800000000005
                (9) 0.002815100000000001
                (10) 0.0036471999999999893
         Op Counts:
                (1) 8926
                (2) 8954
                (3) 8960
                (4) 8928
                (5) 8938
                (6) 8986
                (7) 8934
                (8) 8943
                (9) 8952
                (10) 8929


Average Run Time = 0.003040720000000001
Average Number of Basic Operations = 8945.0
```

Merge Sort(2^{15})

```
elo_sim merge_sort
        Run Times:
                (1) 0.1392779
                (2) 0.15124369999999998
                (3) 0.13306929999999995
                (4) 0.13383499999999993
                (5) 0.12562360000000006
                (6) 0.14420809999999984
                (7) 0.14397460000000017
                (8) 0.12546579999999996
                (9) 0.14516379999999995
                (10) 0.15203389999999994
         Op Counts:
                (1) 449970
                (2) 450175
                (3) 450188
                (4) 450053
                (5) 450141
                (6) 449888
                (7) 450008
                (8) 450042
                (9) 449985
                (10) 450054


Average Run Time = 0.13938957
Average Number of Basic Operations = 450050.4
```

Merge Sort(2^{20})

```
elo_sim merge_sort
        Run Times:
                (1) 9.180140300000001
                (2) 8.217722099999998
                (3) 8.0062602
                (4) 7.571647400000003
                (5) 8.425538999999993
                (6) 9.625389999999996
                (7) 9.517097899999996
                (8) 8.989274599999987
                (9) 9.517901199999997
                (10) 9.32826829999999
         Op Counts:
                (1) 19644875
                (2) 19643501
                (3) 19642990
                (4) 19644073
                (5) 19643698
                (6) 19643706
                (10) 19643662


Average Run Time = 8.837924099999997
Average Number of Basic Operations = 19643894.2
```

Quick Sort(2^{10})

```
elo_sim quick_sort
        Run Times:
                (1) 0.0024826999999999905
                (2) 0.0022183000000000064
                (3) 0.0019184000000000007
                (4) 0.0020084000000000074
                (5) 0.0024765999999999955
                (6) 0.0019940000000000235
                (7) 0.0020638000000000045
                (8) 0.0019374000000000058
                (9) 0.0027195999999999887
                (10) 0.004129500000000008
         Op Counts:
                (1) 11603
                (2) 11244
                (3) 11557
                (4) 11313
                (5) 10844
                (6) 11028
                (7) 11501
                (8) 11170
                (9) 10686
                (10) 11027


Average Run Time = 0.002394870000000003
Average Number of Basic Operations = 11197.3
```

Quick Sort(2^{15})

```
elo_sim quick_sort
        Run Times:
                (1) 0.14341619999999988
                (2) 0.13856559999999973
                (3) 0.2270563000000001
                (4) 0.16936630000000008
                (5) 0.1350804000000001
                (6) 0.1196432999999999
                (7) 0.18471700000000002
                (8) 0.1745490000000003
                (9) 0.15797529999999993
                (10) 0.17743139999999968
         Op Counts:
                (1) 790156
                (2) 764163
                (3) 777694
                (4) 805767
                (5) 819506
                (6) 770671
                (7) 795941
                (8) 773161
                (9) 791755
                (10) 775734


Average Run Time = 0.16278007999999997
Average Number of Basic Operations = 786454.8
```

Quick Sort(2^{20})

```
elo_sim quick_sort
        Run Times:
                (1) 61.84822349999999
                (2) 54.21739579999999
                (3) 45.314307499999984
                (4) 47.574325000000044
                (5) 57.64768460000005
                (6) 62.08513210000001
                (7) 58.95364760000001
                (8) 57.15292720000002
                (9) 48.96740810000006
                (10) 35.669481700000006
         Op Counts:
                (1) 352800326
                (2) 354758549
                (3) 353762489
                (4) 353021359
                (5) 353408040
                (6) 353490317
                (7) 354744888
                (8) 352883889
                (9) 353879126
                (10) 353885307


Average Run Time = 52.94305331000002
Average Number of Basic Operations = 353663429.0
```

References

Last Name, F. M. (Year). Article Title. *Journal Title*, Pages From - To.

Last Name, F. M. (Year). *Book Title.* City Name: Publisher Name.

Footnotes

[1][Add footnotes, if any, on their own page following references. For APA formatting requirements, it's easy to just type your own footnote references and notes. To format a footnote reference, select the number and then, on the Home tab, in the Styles gallery, click Footnote Reference. The body of a footnote, such as this example, uses the Normal text style. *(Note: If you delete this sample footnote, don't forget to delete its in-text reference as well. That's at the end of the sample Heading 2 paragraph on the first page of body content in this template.)*]

Tables

Table 1

[Table Title]

Column Head	Column Head	Column Head	Column Head	Column Head
Row Head	123	123	123	123
Row Head	456	456	456	456
Row Head	789	789	789	789
Row Head	123	123	123	123
Row Head	456	456	456	456
Row Head	789	789	789	789

Note: [Place all tables for your paper in a tables section, following references (and, if applicable, footnotes). Start a new page for each table, include a table number and table title for each, as shown on this page. All explanatory text appears in a table note that follows the table, such as this one. Use the Table/Figure style, available on the Home tab, in the Styles gallery, to get the spacing between table and note. Tables in APA format can use single or 1.5 line spacing. Include a heading for every row and column, even if the content seems obvious. A default table style has been setup for this template that fits APA guidelines. To insert a table, on the Insert tab, click Table.]

Figures title:

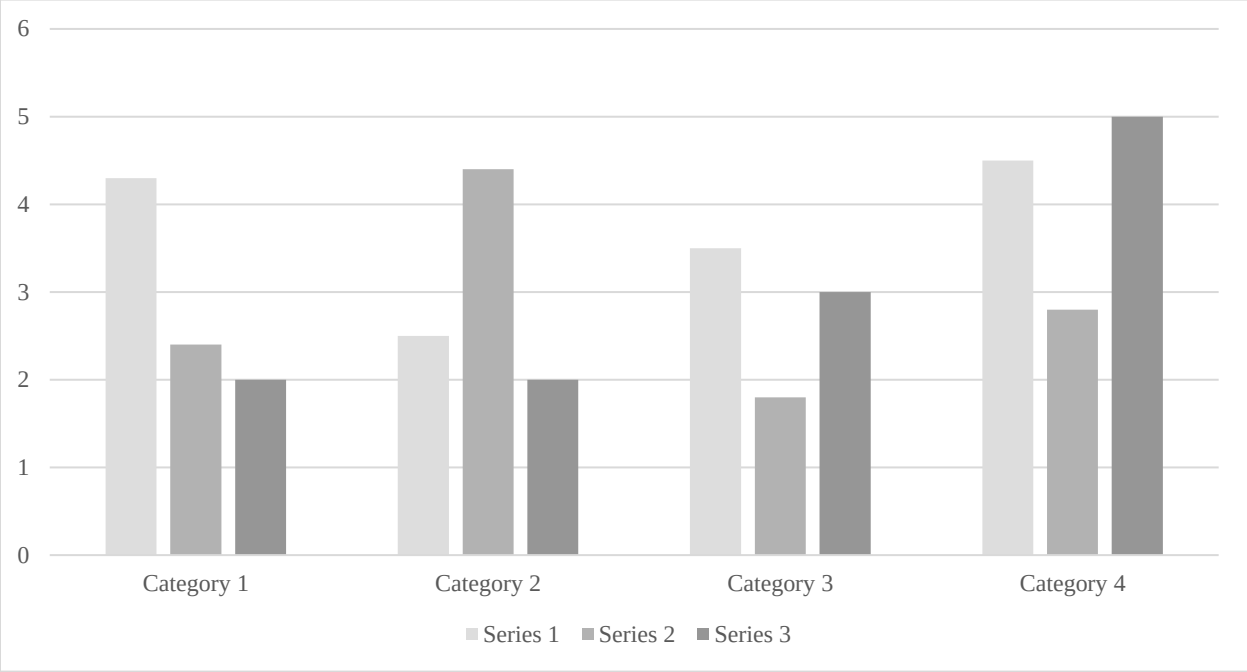

Figure 1. [Include all figures in their own section, following references (and footnotes and tables, if applicable). Include a numbered caption for each figure. Use the Table/Figure style for easy spacing between figure and caption.]

For more information about all elements of APA formatting, please consult the *APA Style Manual, 6th Edition*.