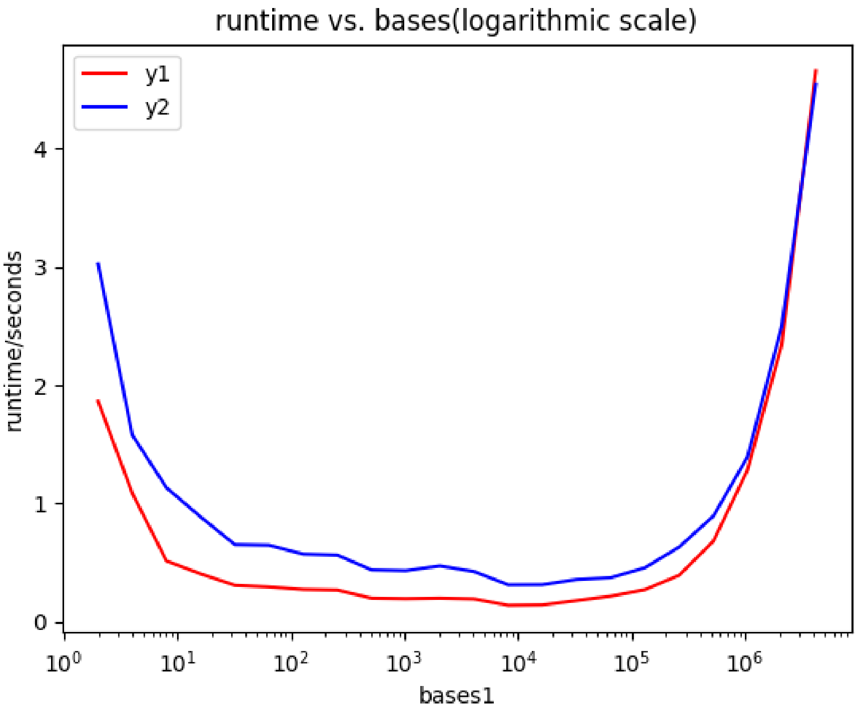
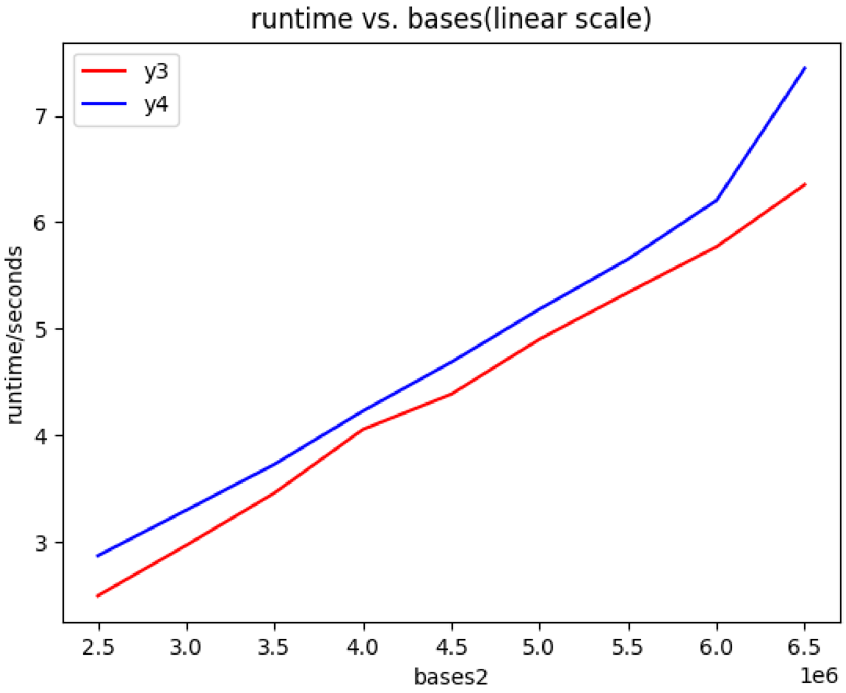
Name: Lahiru Hasaranga Weliwitiya Student ID: 31177654

FIT2004 - Assignment 1 Q2 Answers

The diagrams for the two cases required are given below:





1. Why do the base/time curves for the first two graphs show a U shape? In other words, why are the times high when the base is low and when the base is high, but low when the base is in between? Justify your answer using the complexity of radix sort.

The time complexity of a radix sort algorithm is O((n+b) * k) where "n" is the element of the input, b is the base that is being considered, and k is the number of digits required to represent the largest value of he input sequence.

Due to the nature of the scale used, there is seen to be an inverse relationship between the size of the base and the number of columns that is being used to sort the sequence. This relationship is further highlighted with the use of a log graph.

So the left curve of the U-shape is a result of the number of digits in the base representation increasing as the base decreases, increasing the number of sorting operations that have to take place.

And the right curve has occurred as a result of the number of columns decreasing, but the size of the comparable digits increasing.

2. Why are the times for y2 about twice as long as for y1 when the base is low? Include a mathematical argument based on the complexity of radix sort in your answer.

The values stored in y2 are larger to begin with, compared to y1. It can be seen that 2^16 digits will be put to use, so it is understandable that the times for y2 be as large as they are.

3. Why are the times for y2 not twice as long as for y1 (and in fact are very close) when the base is high? Include a mathematical argument based on the complexity of radix sort in your answer.

The scale of the x axis is given in a logarithmic scale, which means that the larger the bases for y2 get, the more exponentially they increase. This is also due to the increase time taken for processing the columns of the sequence, which grow wider towards the right of the graph

4. Why are the times for y3 and y4 almost the same, despite data2 having twice as many elements as data1? Include a mathematical argument based on the complexity of radix sort in your answer.

The column width increases as the base increases, acting as a compromise for the curve to remain linear.

5. Why do the graphs for y3 and y4 show an almost linear shape? Include a mathematical argument based on the complexity of radix sort in your answer.

Because they are bounded by the same complexity curve, which is O(n). This is because a close compromise is achieved between the base and the column number of the list of data values that requires sorting. This means that the length of the count array increases at a uniform rate with the array generated by every base.

Both are seen to increase at a uniform rate, Thus turning the complexity from O((n+b) * k) into O((n+n)*k) = O((2n)*k) = O(k*n), which is essentially O(n)