

CS361 Algorithm Winter 2017 Lab 3

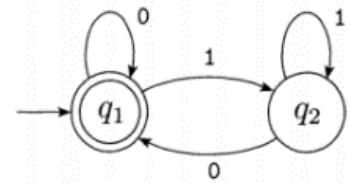
Due 6/8/2017

What to do

1. Implement the Radix sort algorithm and use it to sort roughly 10,000,000 numbers. I am providing a new data file.
2. Implement the Bin sort algorithm and use it to sort roughly 10,000,000 numbers.
3. Make sure the results are actually sorted for 1 and 2.
4. Show the execution time comparison with your either quick sort or merge sort. Also make sure the result of your quick sort or merge sort is actually sorted.
5. Show the screen dump indicate the sorting algorithms are actually sorting correctly.
6. Write a program to implement a generic simulator for DFA. At the time of instantiation, you can specific your DFA simulator to implement a specific DFA by providing it with a 5-tuple $(Q, \Sigma, \delta, q_0, F)$. You can also have a section in your code to specify your machine, or enter your formal definition at run time.

You then need to bound your generic DFA with the one on the right and show me the output (accepting the string or not) for the following strings:

1. 10101
2. 0010
3. 0010100
4. 1000
5. ϵ (the empty string, some people use the letter E for that)



If I were to implement this, I would declare a class and have a member for each of the five tuples. I also need a member to store the current symbol, current state, and the input string. The transition table's column index is used to determine the alphabet, still using index to get to the actual symbol. This calls for a private method to get the symbol for a given index.

7. Run your code for 1~3 three times, record the execution time in milliseconds for each run on each size, enter the milliseconds reading into an Excel spreadsheet, calculate the average execution time in milliseconds and provide your results in both a table and as a line chart.
8. Add comments to your code!!!
9. Write a half to one page report to explain your execution time observation and discuss the problem solving approach you applied. The expected results should be Bin sort is the fastest, then Radix sort, then either quick or merge sort.

What to turn in

A PDF file (not WORD file nor zipped files) that contains the screen dump showing your programs execution times and they are correct, then your report followed by code segments and corresponding outputs (these are 3 and 6).