CS472 Assignment 3: Practice with Big-Oh

CS472 - Analysis of Algorithms

January 19, 2017

1 Overview

The assignment contains a set of practice problems covering asymptotic measures of efficiency and working with recurrence relations.

2 Problems

1. Show directly that $f(n) = n^2 + 3n^3 \in \Theta(\mathrm{n}^3)$. That is, use the definitions of O and Ω to show that $f(n)$ is in both $O(n^3)$ and $\Omega(n^3)$.

2. Consider the following algorithm that checks whether a graph represented by its adjacency matrix is complete:

Algorithm 1: GraphComplete

```
Input: An zero-indexed adjacency matrix A of a undirected graph G
Output: A Boolean value of true if graph is complete, false otherwise
if n = 1 then
    Return true;
else
    if NOT (GC(A[n-2],A[n-2])) then
        Return false;
    else
        for j in [0..n-2] do
            if A[n-1,j] = 0 then
                Return false;
```

What is this order of this algorithm in the worst case? Justify your answer.

3. The authors of modern software libraries are nice enough to include a `sort()` function for use in applications. These libraries claim that the sort functions use an algorithm that is at least $O(n * lg(n))$ in nature. Let's test this claim by writing a small program.

   In your language of choice, use the sort routine provided by the language to sort randomly generated data sets of 5, 10, 50, 100, 500, 1000, 5000, and 10,000 integers. Since we are looking at average data set sizes, you will need to sort at least 10 or more random data sets for each data set size.

   You should implement an algorithm that looks like this:

Algorithm 2: Test the order of the library sort

```
Output: A data set of run times for each data set size
for size ∈ [5, 10, 50, 100, 500, 1000, 5000, 100000] do
    Set totalRunTime leftarrow 0;
    for runNumber ∈ [0 .. NumberOfRuns) do
        GenerateRandomDataSet (dataSet,size);
        Note start time;
        Sort (dataSet);
        Note end time;
        Set elapsed time to difference between end and start times;
        Add elapsed time to totalRunTime;
    Set averageRunTime = totalRunTime / NumberOfRuns;
```

   The function `GenerateRandomDataSet(dataSet, size)` needs to generate a random set of integers of whatever size is passed into the function as an input.

   Create a graph (using Excel or your favorite graphing tool) that graphs data set size versus the average run time for each size. Include a plot of data set size versus $f(n) = n * lg(n)$ as well. Submit your source code, graph, and a short (1-3 paragraph) analysis of the results.

   Hint: you will need to research two items for this question: (1) the functions in your language of choice to deal with function timing, and (2) how to call the `sort()` function in your language of choice.

3 Submission instructions

You will need to attach a PDF file to your problem submission in Blackboard that details your responses to the questions in the previous section.