CS2010 Semester 1 2012/2013

Data Structures and Algorithms II

Tutorial 01 - Review of CS1020 + Intro to CS2010

For Week 03 (27 August - 31 August 2012)

Released: Wednesday, August 22, 2012

Document is last modified on: July 26, 2012

1 Introduction and Objective

The purpose of this tutorial is to review and refresh the basic concepts taught in CS1020 which will be essential foundation to CS2010. Please try out all the questions in the tutorial, and if you encounter difficulties in answering them, do attend the remedial lectures (help sessions) and don't hesitate to clear up all doubts and questions you might have, with the lecturer or tutor.

2 Tutorial 01 Questions

Sorting

Q1. Which sorting algorithm is the best to sort the following sequence in *ascending order*?
{10000000, 9, 8, 7, 6, 5, 4, 3, 2, 1}
Explain your answer!

1. Insertion Sort
2. Quick Sort (note: Quick Sort as taught in CS1020)
3. Bubble Sort
4. Selection Sort
5. Merge Sort

Hashing

Q2. Which of the Following is the best hash function?

1. `Index = (currentTimeMillis() * Key.charAt(0) = 'A') % N;`
2. `Index = (Key.charAt(0) - 'A') % N;`
3. `Index = Key.hashCode() % N;`

where

- Key is a string
- `currentTimeMillis()` is a function in `Java.lang.System` that returns the current time in milliseconds
- `N` is the hash table size, usually a prime number

Order of Growth

Q3. What is the bound of the following function? $\mathbf{F(}n\mathbf{)} = \log\mathbf{(2}^n\mathbf{)} + \sqrt{n} + \mathbf{100000000}$

1. O(n)
2. O($n \log n$)
3. O(n^2)
4. O(1)
5. O(2^n)

Q4. What is the bound of the following function? $\mathbf{F}(n) = n + \mathbf{(1/2)}n + \mathbf{(1/3)}n + \mathbf{...} + \mathbf{1}$

1. $O(n)$
2. $O(n \log n)$
3. $O(n^2)$
4. $O(1)$
5. $O(2^n)$

Stack

Q5. Given a stack $S1$ containing n integers, show how to sort it in ascending order from top to bottom with help of 2 more extra stacks: $S2$ and $S3$? (operations allowed are: `push()`, `pop()`, `peek()`, and `empty()`). One example is shown below:

Finding k-th Smallest Element (Selection Algorithm)

Q6. In the first lecture you have learned about using the concept of quicksort's partitioning algorithm combined with binary search-like algorithm on an unsorted array to find the k-th smallest element in the array. In this tutorial, we will spend some time discussing the details. Before attending this tutorial, please investigate the Internet about this algorithm (`en.wikipedia.org/wiki/Selection_algorithm#Partition-based_general_selection_algorithm` is a good starting point).