

# University of Sargodha

## BS 4<sup>th</sup> Term Examination 2015

Subject: Computer Science

Paper: Design & Analysis of Algorithm (CS: 3143)

\*\*\*\*\*

### **Q1. Write short answers of the following.**

#### **1) Write down steps of designing of an algorithm?**

- **Design:** The first stage is to identify the problem and thoroughly understand it.
- **Analyze:** Once you have the basic framework of the algorithm it's time to start analyzing how efficient the code is in solving the problem.
- **Implement:** Writing and coding the algorithm is the next step in the process.
- **Experiment**

#### **2) Define standard notation?**

Standard notation is when a number is completely written out using numerical digits. Some examples of numbers written in standard notation are 64,100 and 2,000,000. Standard notation is commonly used in everyday math. However, when working with large numbers, it can become cumbersome to write out every single digit.

#### **3) Define role of algorithm in computing?**

**To improve the efficiency of a computer program:** Another way of looking at the efficiency of software is speed. An algorithm can be used to improve the speed at which a program executes a problem. A single algorithm has the potential of reducing the time that a program takes to solve a problem.

**Proper utilization of resources:** The right choice of an algorithm will ensure that a program consumes the least amount of memory. Apart from memory, the algorithm can determine the amount of processing power that is needed by a program.

#### **4) Why we use substitution methods for solving recurrences?**

The substitution method for solving recurrences is famously described using two steps:

- Guess the form of the solution
- Use induction to show that the guess is valid

This method is especially powerful when we encounter recurrences that are non-trivial and unreadable via the master theorem. We can use the substitution method to establish both upper and lower bounds on recurrences.

#### **5) What is Best- and Worst-case complexity of quick sort algo?**

The best case of quick sort is  $n \log(n)$  and the worst case is  $n^2$ .

#### **7) What is Dynamic programming?**

Dynamic Programming refers to a very large class of algorithms. The idea is to break a large problem down (if possible) into incremental steps so that, at any given stage, optimal solutions are known to sub-problems. When the technique is applicable, this condition can be extended incrementally without having to alter previously computed optimal solutions to subproblems. Eventually the condition applies to all of the data and, if the formulation is correct, this together with the fact that nothing remains untreated gives the desired answer to the complete problem.

#### **8) Define minimum spanning tree?**

Given an undirected and connected graph  $G=(V,E)$ , a spanning tree of the graph  $G$  is a tree that spans  $G$  (that is, it includes every vertex of  $G$ ) and is a subgraph of  $G$  (every edge in the tree belongs to  $G$ ). The cost of the

spanning tree is the sum of the weights of all the edges in the tree. There can be many spanning trees. Minimum spanning tree is the spanning tree where the cost is minimum among all the spanning trees. There also can be many minimum spanning trees. Minimum spanning tree has direct application in the design of networks.

### 9) Define sparse Graph?

A graph in which the number of edges is much less than the possible number of edges. A directed graph can have at most  $n(n-1)$  edges, where  $n$  is the number of vertices. An undirected graph can have at most  $n(n-1)/2$  edges.

### Graph: A collection of vertices and edges

- **Vertex:** A simple object that can have a name and other properties
- **Edge:** A connection between two vertices
- **Path:** A list of vertices in which successive vertices are connected by edges in the graph
- **Connected Graph:** A graph in which there is a path from every node to every other node in the graph.
- **Connected Component:** A set of vertices within a graph in which:
  - ✓ there is a path from every vertex within the set to every other vertex in the set, and
  - ✓ there is not a path from any vertex within the set to any vertex not in the set

If a graph is not connected, then it has at least two connected components.

- **Cycle:** A path in which the first and last vertices are the same (i.e., a path from a vertex back to itself)
- **Undirected Graph:** A graph in which the edges are undirected (i.e., bidirectional).
- **Directed Graph:** A graph in which the edges are directed (i.e., unidirectional). If edges are directed, then we speak of the edge as going from one vertex to another vertex.
- **Dense Graph:** Roughly speaking, a graph in which the number of edges is greater than or equal to  $V \lg V$ , where  $V$  is the number of vertices.
- **Sparse Graph:** Roughly speaking, a graph in which the number of edges is  $< V \lg V$ , where  $V$  is the number of vertices

### 10) What is topological sort?

Topological sorting for Directed Acyclic Graph (DAG) is a linear ordering of vertices such that for every directed edge  $uv$ , vertex  $u$  comes before  $v$  in the ordering. Topological Sorting for a graph is not possible if the graph is not a DAG.

For example, a topological sorting of the following graph is "5 4 2 3 1 0". There can be more than one topological sorting for a graph. For example, another topological sorting of the following graph is "4 5 2 3 1 0".

### 11) What is Binary Tree?

A binary search, also known as a half-interval search, is an algorithm used in computer science to locate a specified value (key) within an array. For the search to be binary, the array must be sorted in either ascending or descending order.

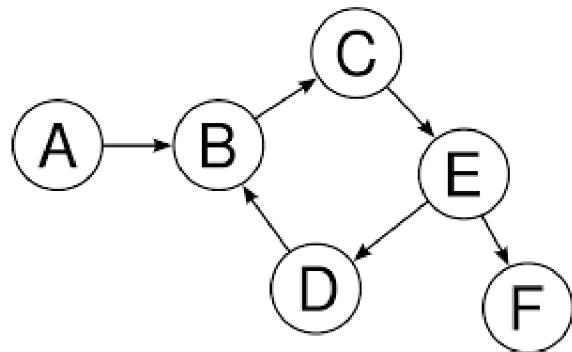
### 12) Difference between Bucket sort and Radix sort?

The initial pass of both RadixSort and BucketSort is exactly the same. The elements are put in buckets (or bins) of incremental ranges (e.g. 0-10, 11-20, ... 90-100), depending on the number of digits in the largest number.

In the next pass, however, BucketSort orders up these 'buckets' and appends them into one array. However, RadixSort appends the buckets without further sorting and 're-buckets' it based on the second digit (ten's place) of the numbers. Hence, BucketSort is more efficient for 'Dense' arrays, while RadixSort can handle sparse (well, not exactly sparse, but spaced-out) arrays well.

Bucket Sort and Radix Sort are like sister sorting algorithms because they are not comparison sorts and the general idea is similar. Also, they both are a bit abstract in implementation.

**14) Draw directed cyclic graph.**



**15) Write elements of Greedy strategy?**

Optimal Substructure: An optimal solution to the problem contains within it optimal solutions to sub-problems.

The 0 - 1 knapsack problem: A thief has a knapsack that holds at most W pounds. Fractional knapsack problem: takes parts, as well as wholes.

**16) What is heap and heap sort?**

A sorting algorithm that works by first organizing the data to be sorted into a special type of binary tree called a heap. The heap itself has, by definition, the largest value at the top of the tree, so the heap sort algorithm must also reverse the order. It does this with the following steps:

1. Remove the topmost item (the largest) and replace it with the rightmost leaf. The topmost item is stored in an array.
2. Re-establish the heap.
3. Repeat steps 1 and 2 until there are no more items left in the heap.

The sorted elements are now stored in an array. A heap sort is especially efficient for data that is already stored in a binary tree. In most cases, however, the quick sort algorithm is more efficient.

**SHAHAB**  
tshahab.blogspot.com