Name : Abdullah Bin Noman Tabid

Student ID : 2120180031

Course Title : Algorithm

Course Code : CSI-227

"Final Exam"

Ans to the Q.No-1(a)

As we study algorithms, we can learn analysis techniques that allow us to compare and contrast solutions based solely on their own characteristics, not the characteristics of the program or computer used to implement them.

There are 4 types of algorithms-

① Brute Force Algorithms:- A brute force algorithm essentially attempts all the chances until an acceptable result is found.

② Recursive Algorithm.

③ Dynamic ~~Programming~~ Programming Algorithm

④ Divide and Conquer Algorithm.

⑤ Greedy Algorithm

⑥ Backtracking Algorithm

⑦ Randomized Algorithm.

Ans to the Q No - 1(b)

Data flow: Data flow analysis is a process for collecting information about the use, definition, and dependencies of data in programs. The data flow analysis algorithm operates on a CFG generated from an AST. We can use a CFG to determine the parts of a

program to which a particular value assigned to a variable might propagate.

Q: Difference between Algorithm and Pseudocode-

Algorithm	Pseudocode
① An algorithm is a well defined.	① It is written by programmers.
② It is a logical approach that comes with a step by step.	② It follows the basic rules of concerned programming language.
③ It follows step by step formula to solve any given program.	④ It follows basic rules to solve any given program.
④ We can express algorithms using flowcharts.	④ We can get to include various control structures using pseudocode.

Ans to the Q No: 1(c)

There are 7 Types of Algorithms

1. Brute Force Algorithm.
2. Recursive Algorithm.
3. Dynamic Programming Algorithm.
4. Divide and Conquer Algorithm.
5. Greedy Algorithm.
6. Back tracking Algorithm.
7. Randomized Algorithm.

Greedy Algorithm:

A greedy algorithm is an algorithm that finds a solution to problems in the

shortest time possible. It picks the path that seems optimal at the moment without regard for the overall optimization of the solution that would be found.

Ans to the Q.No 2(a)

Searching Algorithm: A search algorithm is the step by step procedure used to locate specific data among a collection of data. It is considered a fundamental procedure in computing. In computing science, when searching for data, the difference between a fast application

and a slower one often lies in the use of the proper search algorithm.

Sorting Algorithm: A sorting algorithm is an algorithm that puts elements of a list into an order. The most frequently used orders are numerical order and lexicographical order, and either ascending ~~and~~ ~~or~~ or descending

Ans to the Q.No-2(b)

Regardless of the context in which they are used, algorithms are essentially problem solvers - their purpose is to solve and often automate a solution to a particular problem. Introductory textbooks on algorithms tend to outline their subject broadly, defining an algorithm as a set of steps to accomplish a task.

There are many functions in algorithms. We can use these functions in every algorithm. Algorithm, Procedure that produces the answer to a question or the solution

to a problem, in a finite number of steps. An algorithm that produces a yes or no answer is called a decision procedure; one that leads to a solution is a computation procedure.

Ans to the Q No 2(e)

Mathematical Algorithm: A procedure for solving a mathematical problem (eg of finding the greatest common divisor) in a finite number of steps that frequently involves repetition of an operation.

Graph Algorithm: Graph theory is the study of graphs, which are mathematical structures used to model pairwise relation between objects. A graph in this context is made up of vertices which are connected by edges.

Divide Algorithm: A divide algorithm is an algorithm which, given two integers N and D, computes their quotient and/or remainder, the result of Euclidean division.

Conquer Algorithm: In computer science, divide and conquer is an algorithm design paradigm. A divide and conquer algorithm recursively breaks down a problem into two or more sub problems of the same or related type, until these become simple enough to be solved directly.