

## U-2 - (DAA)

DOMS

Page No.

Date

/ /

Q1-

Worst Case

Best Case

Average Case

calculates.

• upper bound  
↳ running time of algo

• lower bound  
↳ running time of algo.

• all input & ~~output~~ calculate computing time.

• max. no. of oper<sup>n</sup> to be executed.

• min. no. of oper<sup>n</sup> to be executed

• sum of all calculate values & divide sum by total no. of input.

• max time required for program execution.

• min time required for program execution.

• Avg time required for program execution.

• growth will be linear; assuming (n) is odd

• growth is constant, (n) is even.

• even & odd, growth is linear

Q2

$(\neg x \vee y) \vee (x \vee \neg z) \rightarrow$  not satisfiable.



Q.10

Q4

P class problems

- set of problems solved in polynomial time by deterministic algorithm.

- All the P class problems are deterministic

- every problem of P are a subset of NP

- can be solved efficiently

- can be solved & verified in polynomial time

- example - Binary Search, Bubble Sort

NP class problem

- set of problems solved in nondeterministic polynomial time.

- Are non-deterministic

- But NP are superset of P-

- Can it be solved efficiently as P.

- can't be obtained in polynomial time but if solution given can be verified

- example, TSP & Knapsack.



Q5-

NP Hard

Q9-

Q15

NP Complete

- can be ~~solved~~ <sup>reduced</sup> in polynomial time.
- Time is unknown

- To solve  $\rightarrow$  NP Hard  $\rightarrow$  must be in NP class

- Not a decision problem

- Time is unknown

- optimization problem used

- eg - Halting problem, vertex cover etc

- can be solved in poly<sup>n</sup>
- Time is not fixed.

- NP comp  $\rightarrow$  both NP & NP<sup>hard</sup> problem

- Is a decision problem

- Time is fixed.

- decision problem used

- eg - determine whether a graph has a Hamiltonian cycle.

Q6

Asymptotic Notations -

- To compare two algorithm  $\rightarrow$  soft to growth rate
- describes asymptotic efficiency of algo.

Significance  $\rightarrow$  comparative analysis  $\rightarrow$  analysis of algo & dataset  
 $\rightarrow$  diff algo comparison

2) Scalability - Helps us understand performance scal with input size.

3) Algorithm Design - optimize algo to achieve better time & space complexity

4) Predictive Power - provide insights of how an algo will perform for large input without execution



Q7 Measure algorithms's efficiency & algo's running time.

→ efficiency of algo can be measured by time complexity & space complexity

- Time complexity → amount of time taken for algo to run. measured using asymptotic notation.
- Space complexity → amt of memory an algo uses. measured using asymptotic notation.

We measure efficiency of algorithm by, comparing its time complexity & space complexity to another algorithm.

- Running time of algo → can vary  $\begin{cases} \text{specific input} \\ \text{implementation of algo} \end{cases}$
- Efficiency of algo → how well it uses available resource  
more efficient algo  $\begin{cases} \text{run fast} \\ \text{use less memory} \end{cases}$

To measure

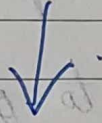
- Counting no. of operation →  $\begin{cases} \uparrow \text{operation} \\ \downarrow \text{efficiency} \end{cases}$
- Using asymptotic notation →  $\begin{cases} \text{compare diff} \\ \text{algo without any} \\ \text{constants} \end{cases}$
- Running algo on diff input → actual running time can be determined



Q8- Formulate order of growth. compare  $\rightarrow n! \& 2^n$ .

- also known as asymptotic complexity
- describes how runtime or resource usage of an algo scales with input sizes.

compare  $n! \& 2^n$



Q11-

## Time space Trade-off

- • Refers to fact that there is often a trade-off b/w amt of time it takes to solve problem & amt of space required to solve
- • Algo  $\rightarrow$   $\downarrow$  time  $\rightarrow$   $\uparrow$  more space & viceversa
- • balancing use of time & space to optimize performance of an algo.
- • It improves performance of algo.  
eg  $\rightarrow$  algo  $\rightarrow$  too long to run  $\rightarrow$  implement in less space.
- • reduce space requirements  
eg  $\rightarrow$  algo  $\rightarrow$  too much space  $\rightarrow$  longer to run.

Q12-

3- Conjunctive Normal Form is NP-complete.



$V_1 \rightarrow$   
 $V_2 \rightarrow 2, 3, 8, 12, 13, 16$

DOMS	Page No.
Date	/ /

Q13 Vertex Cover Problem is NP complete.

Q15 NP Hard & NP complete (Doubt already)

Q16 Prove that Hamiltonian cycle is NP hard problem.

Q14 3 SAT problem.

- particular case of satisfiability problem.
- also known as Boolean satisfiability or propositional satisfiability
- prove difficulty of other problems, such as clique problem
- many algo  $\rightarrow$  proposed for 3 SAT problem
- No known polynomial time algorithm
- has appl<sup>n</sup> in job scheduling & circuit design.
- here the boolean expression has a strict form.