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 & calculate computing time.
•	max. no. of oper^n to be executed.	• min. no. of oper^n to be executed	• sum of all calculated values & divide sum by total no. of inputs
•	max time required for program execution.	• mini time required for program execution.	• Avg time required for program exec^n.
•	growth will be linear; assuming (n) is odd	• growth is constant, (n) is even.	• even & odd any possible, growth is linear

Q2 $(\neg x \vee y) \vee (x \vee \neg z)$ → not satisfiable.

Q4 Q.10

P class problems	NP class prob'm
• set of problems solved in polynomial time by deterministic algorithm.	• set of problems solved in nondeterm polynomial time.
• All the P class problems are deterministic	• Are non-determ
• every problem of P are a subset of NP	• But NP are superset of P.
• can be solved efficiently	• can it be solved efficiently as P.
• can be solved & verified in polynomial time	• can't be obtained in polynomial time but if soln given can be verified
• example - Binary Search, Bubble Sort	• example, TSP & Knapsack.

Q5- Q9- Q15

NP Hard	NP Complete
• can be ~~solved~~ reduced in polynomial time.	• Can be solved in poly^ time.
• ~~Time is unknown~~	• ~~Time is not fixed.~~
• To solve → NP Hard → must be in NP class	• NP comp → both NP & NP Hard problem.
• Not a decision problem	• Is a decision problem
• Time is unknown	• Time is fixed.
• optimization problem used	• decision problem used
• eg- Halting problem, vertex cover etc.	• eg- determine whether a graph has a Hamiltonian cycle.

Q6 Asymptotic Notations -

- To compare two algorithm → refto growth rate
- describes asymptotic efficiency of algo.

Significance → 1) comparitive analysis → analysis of algo & datastruct
↳ diff algo comparision

2) Scalability - Helps us understand performance scalwith i/p size.

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

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

DOMS | Page No.
Date / /

Q1 Measure algorithm'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 → specific input
  → implementation of algo
- efficiency of algo → how well it uses available resource
  More efficient algo → run fast
  → use less memory

To measure
- Counting no. of operation → ↑operation ↓efficiency
- Using asymptotic notation → • compare diff algo without any constants
- Running algo on diff input → actual running time can be determined

Q8. Formulate order of growth. compare → $n!$ & 2^n.

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

compare $n!$ & $2n$

↓

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 → ↓ time → ↑ more space & viceversa
- balancing use of time & space to optimize performance of an algo.
- It improves performance of algo.
  eg → algo → too long to run → implemt in less space.
- reduce space requirement
  eg → algo → too much space → longer to run.

Q12- 3- Conjunctive Normal Form is NP- complete.

U1 →
U2 → 2, 3, 8, 12, 13, 16

DOMS | Page No.
Date / /

Q13 Vertex Cover Problem is NP complete.

Q15 NP Hard & NP complete (Done already)

Q16 Prove that Hamiltonian Cycle is NP hard problem.

Q14 3SAT problem.

- particular case of satisfiability problem).
- also known as Boolean satisfiability or propositional satisfiability
- prove difficulty of the problems, such as clique problem
- many algo → proposed for 3SAT problem
- No know polynomial time algorithm
- has appln in job scheduling & circuit design.
- Here the boolean expression has a strict form.