

## Computer Science Ics Part 1 Chapter 3 Online Test

Sr	Questions	Answers Choice
1	Tractable problem can be solved	<p>A. In exponential time</p> <p>B. Efficiently with large input</p> <p>C. With no algorithm</p> <p>D. Are always NP hard</p>
2	Which is solved by dynamic programming.	<p>A. Coin change</p> <p>B. Fibonacci sequence</p> <p>C. Merge sort</p> <p>D. Puzzle solving</p>
3	Which is an ill-defined problem.	<p>A. Check even number</p> <p>B. Find shortest path</p> <p>C. Reduce poverty</p> <p>D. Count book arrangements</p>
4	Why is bubble sort not for large datasets.	<p>A. Needs more memory</p> <p>B. <math>O(n^2)</math> time complexity</p> <p>C. Works on sorted data only</p> <p>D. Can't handle numbers</p>
5	Which algorithm finds node relationships in a graph.	<p>A. Bubble sort</p> <p>B. Linear Search</p> <p>C. BFS</p> <p>D. Selection sort</p>
6	Knowing if a problem is solvable helps.	<p>A. Increase complexity</p> <p>B. Save time</p> <p>C. Generate random output</p> <p>D. Avoid design</p>
7	Which notation shows the upper bound of time complexity.	<p>A. Omega</p> <p>B. Theta</p> <p>C. Big O</p> <p>D. Lambda</p>
8	What do sorting algorithms mainly do.	<p>A. Network routing</p> <p>B. Arrange data</p> <p>C. Find node links</p> <p>D. Search data</p>
9	What is the purpose of search algorithms	<p>A. Arrange data</p> <p>B. Find specific data</p> <p>C. Analyze nodes</p> <p>D. Save memory</p>
10	What does page rank algorithm analyze.	<p>A. Page links</p> <p>B. Page color</p> <p>C. Page size</p> <p>D. Page font</p>
11	NP hard problems are	<p>A. Easy to solve</p> <p>B. Always solvable</p> <p>C. As hard as NP problem</p> <p>D. In class P</p>
12	Problems in class P are solved in.	<p>A. Exponential time</p> <p>B. Polynomial time</p> <p>C. Non deterministic time</p> <p>D. Infinite time</p>
13	How does Backtracking work.	<p>A. Break into parts</p> <p>B. Build and Backtrack</p> <p>C. Store subproblems</p> <p>D. Make local choices</p>
14	How does Divide and conquer work.	<p>A. Make local choices</p> <p>B. Break, solve, combine</p> <p>C. Store subproblem results</p> <p>D. Try all options, backtrack</p>
15	What is the main advantage of Dynamic programming.	<p>A. Avoid redundant work</p> <p>B. Locally optimal results</p> <p>C. Solve independent problems</p>

		D. <p>Explore alloptions</p>
16	Which is an example of $O(\log n)$ complexity.	A. <p>Sorting numbers</p> B. <p>Pair comparison</p> C. <p>Binary search</p> D. <p>Linear search</p>
17	What is the main advantages of dynamic programming.	A. <p>Avoid redundant work</p> B. <p>Locally optimal results</p> C. <p>Solves independent problems</p> D. <p>Explore alloptions</p>
18	Why are commonly used algorithms important.	A. <p>Reduce hardware cost</p> B. <p>Solve common Problems efficiently</p> C. <p>Remvoe need for data structure</p> D. <p>For theory only</p>
19	Which is solved by Dynaimc programming.	A. <p>Coin Change</p> B. <p>Fibonacci sequence</p> C. <p>Merge sort</p> D. <p>Puzzle solving</p>
20	P vs NP asks if	A. <p>Fast solutions are in NP</p> B. <p>Check complexity</p> C. <p>Debug rrors</p> D. <p>Improve hardware</p>