

## Computer Science Ics Part 1 Chapter 3 Online Test

Sr	Questions	Answers Choice
1	How does selection sort work.	<p>A. &lt;p&gt;Swap adjacent&lt;/p&gt;</p> <p>B. &lt;p&gt;Select minium form unsorted part&lt;/p&gt;</p> <p>C. &lt;p&gt;Use queneue for nodes&lt;/p&gt;</p> <p>D. &lt;p&gt;Store intermediate results&lt;/p&gt;</p>
2	Which is a Greedy Algorithm example	<p>A. &lt;p&gt;Fibonacci sequence&lt;/p&gt;</p> <p>B. &lt;p&gt;Coin change&lt;/p&gt;</p> <p>C. &lt;p&gt;Merge sort&lt;/p&gt;</p> <p>D. &lt;p&gt;Puzzle solving&lt;/p&gt;</p>
3	What is computational problem.	<p>A. &lt;p&gt;solved using algorithm&lt;/p&gt;</p> <p>B. &lt;p&gt;Solved using creativity&lt;/p&gt;</p> <p>C. &lt;p&gt;Has unclear input&lt;/p&gt;</p> <p>D. &lt;p&gt;Has no process&lt;/p&gt;</p>
4	What is output when checking if 8 is even.	<p>A. &lt;p&gt;Found at 2&lt;/p&gt;</p> <p>B. &lt;p&gt;Even&lt;/p&gt;</p> <p>C. &lt;p&gt;Odd&lt;/p&gt;</p> <p>D. &lt;p&gt;Yes&lt;/p&gt;</p>
5	What is limitation Greedy algorithms	<p>A. &lt;p&gt;Too complex&lt;/p&gt;</p> <p>B. &lt;p&gt;Always optimal&lt;/p&gt;</p> <p>C. &lt;p&gt;Not always optimal&lt;/p&gt;</p> <p>D. &lt;p&gt;High memory use&lt;/p&gt;</p>
6	Which problem gives yes /no answer.	<p>A. &lt;p&gt;Search&lt;/p&gt;</p> <p>B. &lt;p&gt;Decision&lt;/p&gt;</p> <p>C. &lt;p&gt;Optimization&amp;nbsp;&lt;/p&gt;</p> <p>D. &lt;p&gt;Counting&lt;/p&gt;</p>
7	The Halting Problem is an example of.	<p>A. &lt;p&gt;Solvable problem&lt;/p&gt;</p> <p>B. &lt;p&gt;Tractable prolem&lt;/p&gt;</p> <p>C. &lt;p&gt;Unsolvable problem&lt;/p&gt;</p> <p>D. &lt;p&gt;NP Complete problem&lt;/p&gt;</p>
8	What does time complexity measrue.	<p>A. &lt;p&gt;Memory use&lt;/p&gt;</p> <p>B. &lt;p&gt;Time growth with input&lt;/p&gt;</p> <p>C. &lt;p&gt;Number of loops&lt;/p&gt;</p> <p>D. &lt;p&gt;Output accuracy&lt;/p&gt;</p>
9	What does page rank algorithm analyze.	<p>A. &lt;p&gt;Page links&lt;/p&gt;</p> <p>B. &lt;p&gt;page color&lt;/p&gt;</p> <p>C. &lt;p&gt;Page size&lt;/p&gt;</p> <p>D. &lt;p&gt;Page font&lt;/p&gt;</p>
10	Which is an NP complete problem.	<p>A. &lt;p&gt;Sorting&amp;nbsp;&lt;/p&gt;</p> <p>B. &lt;p&gt;Knapsack&lt;/p&gt;</p> <p>C. &lt;p&gt;GCD finding&lt;/p&gt;</p> <p>D. &lt;p&gt;Binary search&lt;/p&gt;</p>
11	Why is tractability important.	<p>A. &lt;p&gt;Output color&lt;/p&gt;</p> <p>B. &lt;p&gt;Language choice&lt;/p&gt;</p> <p>C. &lt;p&gt;Solve efficiently&lt;/p&gt;</p> <p>D. &lt;p&gt;Input type&lt;/p&gt;</p>
12	What do sorting algorithm mainly do.	<p>A. &lt;p&gt;Network routing&lt;/p&gt;</p> <p>B. &lt;p&gt;Arrange data&lt;/p&gt;</p> <p>C. &lt;p&gt;Find node links&lt;/p&gt;</p> <p>D. &lt;p&gt;Search data&lt;/p&gt;</p>
13	If $P = NP$ ,this means	<p>A. &lt;p&gt;Some NP unsolvable&lt;/p&gt;</p> <p>B. &lt;p&gt;all NP solvable in polynomial time&lt;/p&gt;</p> <p>C. &lt;p&gt;All NP unsolvable&lt;/p&gt;</p> <p>D. &lt;p&gt;Same space complexity&lt;/p&gt;</p>
14	The meanig of NP in computational complexity is.	<p>A. &lt;p&gt;Non deterministic polynomial time&lt;/p&gt;</p> <p>B. &lt;p&gt;Negative polynomial time&lt;/p&gt;</p> <p>C. &lt;p&gt;No trivial polynomial time&lt;/p&gt;</p> <p>D. &lt;p&gt;Numerical polynomial time&lt;/p&gt;</p>
15	Which alorithm finds node relationships in a aranh.	<p>A. &lt;p&gt;Bubble sort&lt;/p&gt;</p> <p>B. &lt;p&gt;Linear Search&amp;nbsp;&lt;/p&gt;</p>

15	Which algorithm finds node relationships in a graph?	<p>C. &lt;p&gt;BFS&lt;/p&gt;</p> <p>D. &lt;p&gt;Selection sort&lt;/p&gt;</p>
16	The halting problem is.	<p>A. &lt;p&gt;Solvable and tractable&lt;/p&gt;</p> <p>B. &lt;p&gt;Unsolvable and in P&lt;/p&gt;</p> <p>C. &lt;p&gt;Unsolvable and not in NP&lt;/p&gt;</p> <p>D. &lt;p&gt;NP Complete&lt;/p&gt;</p>
17	Problems in NP and NP-hard are called.	<p>A. &lt;p&gt;P&lt;/p&gt;</p> <p>B. &lt;p&gt;BPP&lt;/p&gt;</p> <p>C. &lt;p&gt;NP Complete&lt;/p&gt;</p> <p>D. &lt;p&gt;Tractable&lt;/p&gt;</p>
18	Easy to verify but hard to solve problem are in.	<p>A. &lt;p&gt;NP&lt;/p&gt;</p> <p>B. &lt;p&gt;P&lt;/p&gt;</p> <p>C. &lt;p&gt;NP hard&lt;/p&gt;</p> <p>D. &lt;p&gt;Undecidable&lt;/p&gt;</p>
19	For which problems is Backtrackign suitable.	<p>A. &lt;p&gt;Optimal substructure&lt;/p&gt;</p> <p>B. &lt;p&gt;Explore all combinations&lt;/p&gt;</p> <p>C. &lt;p&gt;One optimal choice&lt;/p&gt;</p> <p>D. &lt;p&gt;No overlapping problems&lt;/p&gt;</p>
20	Whcih is a sorting algorithms.	<p>A. &lt;p&gt;Linear search&lt;/p&gt;</p> <p>B. &lt;p&gt;Qucik sort&lt;/p&gt;</p> <p>C. &lt;p&gt;DFS&lt;/p&gt;</p> <p>D. &lt;p&gt;BFS&lt;/p&gt;</p>