School of Computing
First CIA Exam – Feb 2024
Course Code: CSE318
Course Name: Algorithm Design Strategies & Analysis
Duration: 90 minutes Max Marks: 50

Key for Evaluation

PART A

Answer all the questions **10 x 2 = 20 Marks**

1. How to quantify the efficiency of an algorithm?
   Ans: Time Complexity and Space Complexity
2. Define Theta notation.
   Ans: A function f(n) is said to be in Theta, θ(g(n)), denoted as f(n) ∈ θ(g(n)) or simply, f(n) = θ(g(n)), if f(n) is bounded BELOW and ABOVE by some +ve constant multiples of g(n) for all large 'n', if there exist some +ve constants c_1 and c_2, and some non-negative integer n_0
3. Find the complexity of below codes.

```
function(int n) {
    for (int i=1; i<=n; i++) {
        for (int j=1; j*j<=n; j++) {
            printf("*");
            break;
        }
    }
}
```

   Ans: Since there is a "break" statement in the inner 'for' loop, the complexity of this algorithm is O(n)
4. Compare the divide & conquer approach with dynamic programming approach.
   Ans: Divide-and-conquer algorithms partition the problem into disjoint sub problems, solve the sub problems recursively, and then combine their solutions to solve the original problem. In contrast, dynamic programming applies when the subproblems overlap—that is, when sub problems share subproblems. DP solves the sub problems just once and saves its answer in a table thereby avoiding the work of re-computation.

5. Solve the following recurrence using Master theorem.
   $$T(n) = 8T(n/2) + \Theta(n^3)$$
   Ans: Case 2. Solution: O(n^3 log n)
6. Prove that $(3n^2 + 7n)^2 \in O(n^4)$
   Ans:
   $\boldsymbol{Definition\ of\ Big-Oh}$: $f(n) \le c.g(n), \forall n \ge n_0$
   $g(n) = n^4$
   $f(n) = (3n^2 + 7n)^2 = 9n^4 + 49n^2 + 42n^3$
   $f(n) = 9n^4 + 49n^2 + 42n^3 \le 9n^4 + 49n^2.n^2 + 42n^3.n$
   $f(n) \le 9n^4 + 49n^4 + 42n^4$
   $f(n) \le 100n^4$
   $\boldsymbol{f(n) \le 100.g(n), \forall n \ge 1, with\ c = 100}$ & $\boldsymbol{n_0 = 1}$
   So, it is proved that $(3n^2 + 7n)^2 \in O(n^4)$
7. Find the order of growth of the following sum.
   $$\sum_{i=1}^{n}\sum_{j=1}^{i}(i+j)$$
   Ans:
   $$\sum_{i=1}^{n}\sum_{j=1}^{i}(i+j) = \sum_{i=1}^{n}\sum_{j=1}^{i} i + \sum_{i=1}^{n}\sum_{j=1}^{i} j$$
   $$\sum_{i=1}^{n} i\,.\sum_{j=1}^{i} 1 + \sum_{i=1}^{n}\frac{i\,.(i+1)}{2}$$
   $$\sum_{i=1}^{n} i\,.\,i + \frac{1}{2}\sum_{i=1}^{n} i^2 + i = \sum_{i=1}^{n} i^2 + \frac{1}{2}\sum_{i=1}^{n} i^2 + \frac{1}{2}\sum_{i=1}^{n} i$$
   $$O(n^3) + \frac{1}{2}O(n^3) + \frac{1}{2}O(n^2) \in O(n^3)$$

8. Find the recurrence by analyzing the following simple algorithm.
   Algorithm MyFun(n)
       If n<=2 ***Then***
           Return n
       Else
           Return 2*MyFun(n/3)*MyFun(2*n/3)
       End If
   End MyFun
   Ans:

$T(n) = T(n/3) + T(2n/3) + O(1)$, If $n>2$
$T(n) = O(1)$, If $n<=2$

9. What is optimization problem? Which algorithm design strategy is used mostly for solving optimization problem?
Ans: A computational problem in which the objective is to find the best of all possible solutions. More formally, find a solution in the feasible region which has the minimum (or maximum) value of the objective function. Greedy strategy is used mostly in solving optimization problem.
10. Consider a set of unordered elements. Problem is to search an element from the list. Suggest a best searching algorithm and justify the reason.
Ans: Linear Search is best suitable for searching on unordered set of elements. Since the input is unordered element, we cannot apply binary search directly for the input. We need to sort the elements first before applying binary search. But linear search can be applied for any kind of input with worst case complexity as O(n).

<u>PART B</u>

<u>Answer all the questions</u> **3 x 10 = 30 Marks**

11. By applying divide & conquer strategy algorithm, solve the following maximum sub array problem. Show the step-by-step results of algorithm.

Index	1	2	3	4	5	6	7	8	9	10	11	12	13
Array	-3	-8	1	-2	1	5	-3	-4	3	10	-2	4	-1

Key:
Answer: (9, 12, 15)
Steps in solving – 7 Marks
Correct Answer – 3 Marks

12. (a) Using recursion tree method, solve the following reccurence.
$T(n) = T(n-1) + T(n-2) + O(1)$ if $n>2$
$T(n) = \Theta(1)$ if $n=1$ or $n=2$

Key:

Answer: $O(2^n)$
Recursion Tree – 2 Marks
Steps – 2 Marks
Correct Answer – 1 Mark

(b) Illustrate the greedy algorithm to find a sequence of jobs, which is completed within their deadlines and gives maximum profit for the following input.

n=8	Jobs With Profit & Deadlines							
Jobs	**1**	**2**	**3**	**4**	**5**	**6**	**7**	**8**
Profits	18	31	24	5	53	42	67	39
Deadlines	3	2	1	2	5	5	4	3

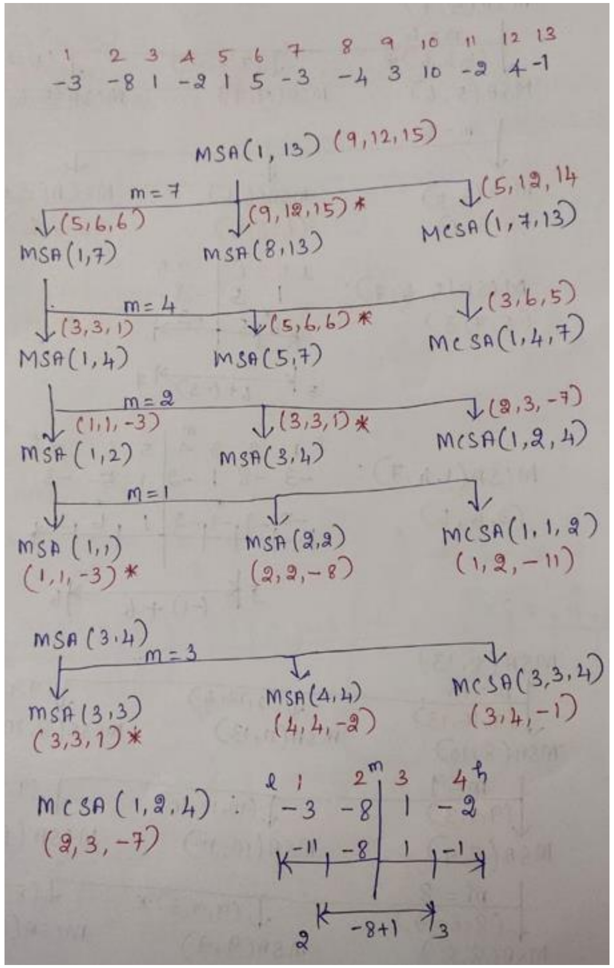
Key:
Answer: Scheduled Jobs: <2, 8, 6, 7, 5>
Profit: 232
Steps – 3 Marks
Correct Answer – 2 Markss

13. Consider a modification of the rod-cutting problem in which, in addition to a price p_i, for each rod, each cut incurs a fixed cost of c. The revenue associated with a solution is now the sum of the prices of the pieces minus the costs of making the cuts. Give a dynamic programming algorithm to solve this modified problem. The algorithm should return the maximum revenue. Using this algorithm, find the maximum revenue for the 5-inch rod with the following price list and the fixed cut cost of Rs.5 per cut.

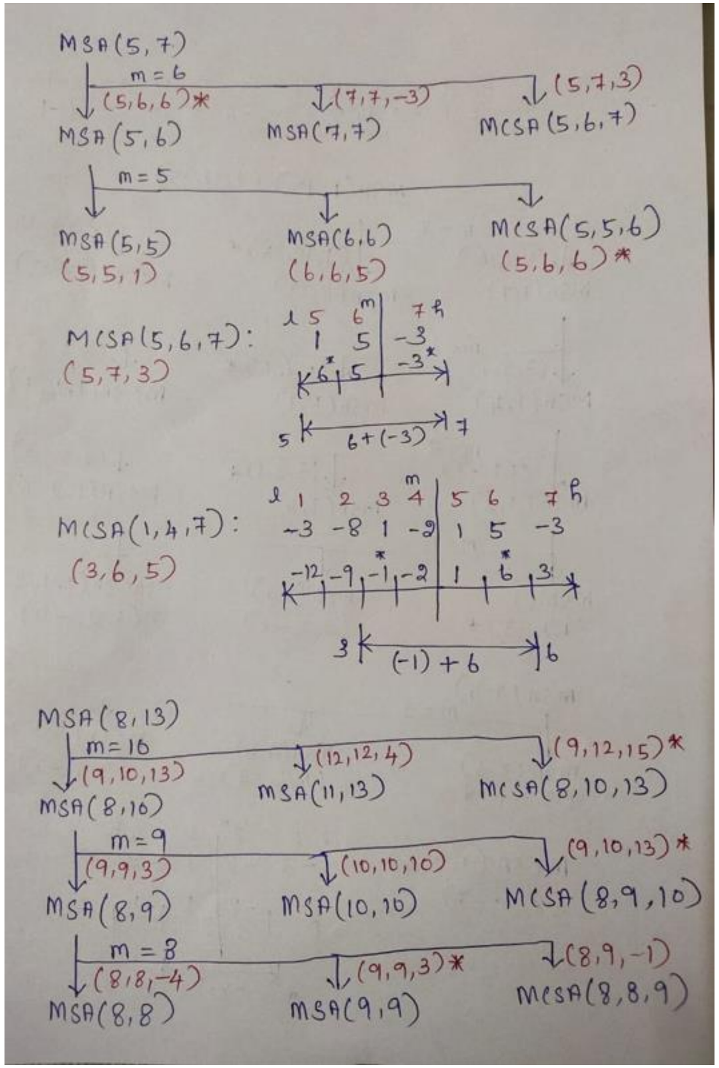
Length of Rod = 5					
Length	**1**	**2**	**3**	**4**	**5**
Price	2	3	7	8	9

Key:
Answer: Profit - 9
Algorithm – 5 Marks
Steps in solving problem – 3 Marks
Correct Answer – 2 Marks

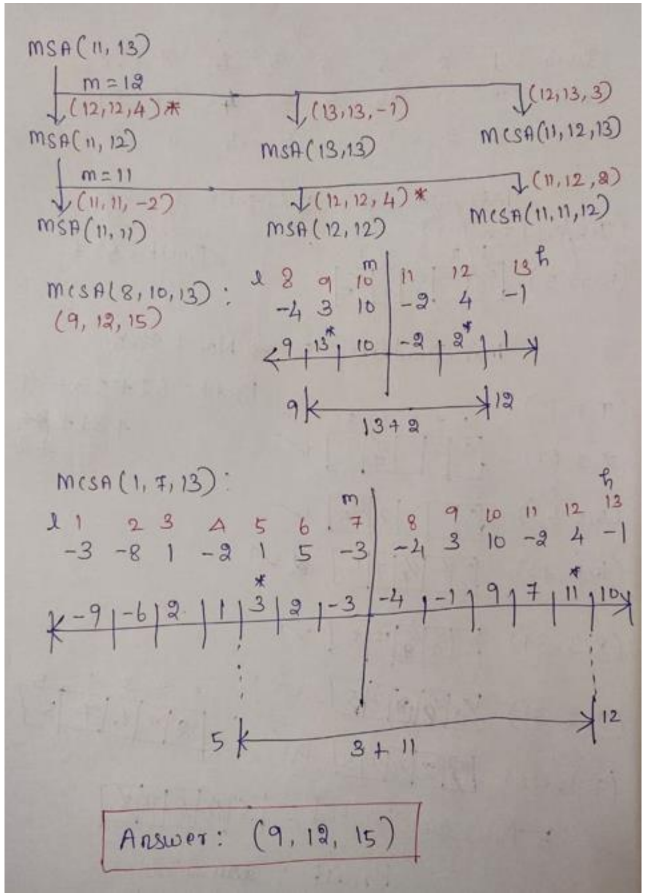
Q.No.: 11 Answer



Q.No.: 11 Answer



Q.No.: 11 Answer



Q. No.: 12 (b)

Job No	1	2	3	4	5	6	7	8
Profit	18	31	24	5	53	42	67	39
Deadline	3	2	1	2	5	5	4	3

Max. Deadline : 5 ⇒ No. of Slots.

Profit : 67 + 53 + 39 + 31 + 42

(J, D, P)

	1	2	3	4	5	
(7, 4, 67)	0	0	0	7	0	✓
(5, 5, 53)	0	0	0	7	5	✓
(6, 5, 42)	0	0	6	7	5	✓
(8, 3, 39)	0	8	6	7	5	✓
(2, 2, 31)	2	8	6	7	5	✓
(3, 1, 24)	2	8	6	7	5	✗

1	2	3	4	5
2	8	6	7	5

5 Jobs are Selected : 2 8 6 7 5

Profit : 232

Q. No.: 13

Length	1	2	3	4	5
Price	2	3	7	8	9

	0	1	2	3	4	5
r	0	2	3	7	8	9

$r[0] = 0$

j = 1:

$i=1 \rightarrow P_1 + r_0 = 2$

j = 2:

$i=1 \rightarrow P_1 + r_1 - 5 = -1$

$i=2 \rightarrow P_2 + r_0 = 3$

j = 3:

$i=1 \rightarrow P_1 + r_2 - 5 = 0$

$i=2 \rightarrow P_2 + r_1 - 5 = 0$

$i=3 \rightarrow P_3 + r_0 = 7$

j = 4:

$i=1 \rightarrow P_1 + r_3 - 5 = 4$

$i=2 \rightarrow P_2 + r_2 - 5 = 1$

$i=3 \rightarrow P_3 + r_1 - 5 = 4$

$i=4 \rightarrow P_4 + r_0 = 8$

j = 5:

$i=1 \rightarrow P_1 + r_4 - 5 = 5$

$i=2 \rightarrow P_2 + r_3 - 5 = 5$

$i=3 \rightarrow P_3 + r_2 - 5 = 5$

$i=4 \rightarrow P_4 + r_1 - 5 = 5$

$i=5 \rightarrow P_5 + r_0 = 9$

Answer : 9