90000

PDSA

Section Id :	64065349261
Section Number :	2
Section type :	Online
Mandatory or Optional :	Mandatory
Number of Questions :	26
Number of Questions to be attempted :	26
Section Marks :	100
Display Number Panel :	Yes
Section Negative Marks :	0
Group All Questions :	No
Enable Mark as Answered Mark for Review and Clear Response :	Yes
Maximum Instruction Time :	0
Sub-Section Number :	1
Sub-Section Id :	640653103254
Question Shuffling Allowed :	No
Is Section Default? :	null

Question Number : 22 Question Id : 640653697596 Question Type : MCQ Is Question Mandatory : No Calculator : None Response Time : N.A Think Time : N.A Minimum Instruction Time : 0

Correct Marks : 0

Question Label : Multiple Choice Question

THIS IS QUESTION PAPER FOR THE SUBJECT "DIPLOMA LEVEL : PROGRAMMING, DATA STRUCTURES AND ALGORITHMS USING PYTHON (COMPUTER BASED EXAM)"

ARE YOU SURE YOU HAVE TO WRITE EXAM FOR THIS SUBJECT?

CROSS CHECK YOUR HALL TICKET TO CONFIRM THE SUBJECTS TO BE WRITTEN.

(IF IT IS NOT THE CORRECT SUBJECT, PLS CHECK THE SECTION AT THE TOP FOR THE SUBJECTS REGISTERED BY YOU)

Options :

6406532329803. ✔ YES

6406532329804. ✖ NO

Sub-Section Number :	2
Sub-Section Id :	640653103255
Question Shuffling Allowed :	Yes
Is Section Default? :	null

Question Number : 23 Question Id : 640653697597 Question Type : MCQ Is Question Mandatory : No Calculator : None Response Time : N.A Think Time : N.A Minimum Instruction Time : 0

Correct Marks : 4

Question Label : Multiple Choice Question

Consider the following code.

```
def fun(n):
    total = 0
    for i in range(n):
        for j in range(n):
            for k in range(j, n):
                total = total + 1
    return total
```

What is the time complexity of the function `fun` in terms of `n`?

Options :

6406532329805. ✖ $O(n^2)$

6406532329806. ✖ $O(n^2 \log n)$

6406532329807. ✖ $O(n \log n)$

6406532329808. ✔ $O(n^3)$

Question Number : 24 Question Id : 640653697598 Question Type : MCQ Is Question Mandatory : No Calculator : None Response Time : N.A Think Time : N.A Minimum Instruction Time : 0

Correct Marks : 4

Question Label : Multiple Choice Question

If the input list of `n` elements is already sorted in ascending order, what is the time complexity of applying the **Selection Sort** algorithm to sort the list in ascending order?

Options :

6406532329809. ✔ $O(n^2)$

6406532329810. ✖ $O(n \log n)$

6406532329811. ✖ $O(\log n)$

6406532329812. ✖ $O(n)$

Question Number : 25 Question Id : 640653697599 Question Type : MCQ Is Question Mandatory : No Calculator : None Response Time : N.A Think Time : N.A Minimum Instruction Time : 0

Correct Marks : 4

Question Label : Multiple Choice Question

Consider the following **Insertion sort** algorithm:

```
def insertionsort(L):
    n = len(L)
    if n < 1:
        return(L)
    for i in range(n):
        j = i
        while(j > 0 and L[j] < L[j-1]):
            (L[j],L[j-1]) = (L[j-1],L[j]) # swap operation
            j = j-1
    return(L)
```

To sort the input list `L = [6, 3, 4, 1, 2, 5]`, How many swap operation will be performed by the given algorithm?

Options :

6406532329813. ✖ 4

6406532329814. ✖ 6

6406532329815. ✔ 9

6406532329816. ✖ 10

Question Number : 26 Question Id : 640653697600 Question Type : MCQ Is Question Mandatory : No Calculator : None Response Time : N.A Think Time : N.A Minimum Instruction Time : 0

Correct Marks : 4

Question Label : Multiple Choice Question

Consider that **Quick sort** is applied to a list of `n` distinct elements that is already sorted in the required sorting order. What will be the Worst case time complexity of Quick sort if the pivot is taken to be

$I)$ First element $II)$ Last element

Choose the correct option corresponding to the correct pair of complexities for both pivots.

Options :

6406532329817. ✔ $I : O(n^2)$ and $II : O(n^2)$

6406532329818. ✖ $I : O(n^2)$ and $II : O(n)$

6406532329819. ✖ $I : O(n \log n)$ and $II : O(n \log n)$

6406532329820. ✖ $I : O(n \log n)$ and $II : O(n^2)$

Question Number : 27 Question Id : 640653697601 Question Type : MCQ Is Question Mandatory : No Calculator : None Response Time : N.A Think Time : N.A Minimum Instruction Time : 0

Correct Marks : 4

Question Label : Multiple Choice Question

Consider the following code.

```
L = ['A', 'B', 'C', 'D', 'E']
for i in range(5):
  S.Push( L[i] )
for i in range(4):
  v = S.Pop()
  Q.Enqueue(v)
for i in range(2):
  v = Q.Dequeue()
  S.Push(v)
x = S.Pop()
print(x)
```

Assume `S` is a stack and `Q` is a queue. `Push` and `Pop` operations are usual stack operations, `Enqueue` and `Dequeue` are usual queue operations.

What would be the output of the given code?

Options :

6406532329821. ✖ A

6406532329822. ✖ B

6406532329823. ✖ C

6406532329824. ✔ D

Question Number : 28 Question Id : 640653697602 Question Type : MCQ Is Question Mandatory : No Calculator : None Response Time : N.A Think Time : N.A Minimum Instruction Time : 0

Correct Marks : 4

Question Label : Multiple Choice Question

Linear probing is an open addressing scheme in computer programming for resolving hash collisions in hash tables. Linear probing operates by taking the original hash index and adding successive values linearly until a free slot is found.

A hash table of size **10** (indexed from 0 to 9) initialized with `None`, uses linear probing to resolve collisions. The key values are integers and the hash function used is `key mod 10`. Values **23, 36, 72, 12, 54,** and **83** are stored in the given order in the hash table.

What is the sequence of elements (from index 0 to 9) in the hash table?

Options :

6406532329825. ✔ `None, None, 72, 23, 12, 54, 36, 83, None, None`

6406532329826. ✖ `None, None, 12, 83, 72, 54, 36, 23, None, None`

6406532329827. ✖ `None, None, 72, 12, 23, 54, 36, 83, None, None`

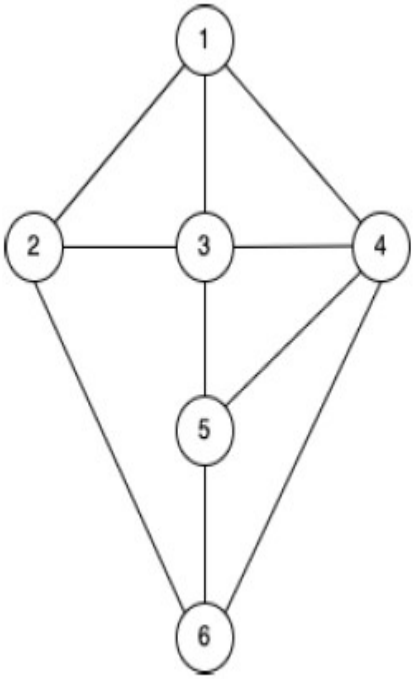
6406532329828. ✖ `None, None, 72, 23, 54, 12, 36, 83, None, None`

Question Number : 29 Question Id : 640653697604 Question Type : MCQ Is Question Mandatory : No Calculator : None Response Time : N.A Think Time : N.A Minimum Instruction Time : 0

Correct Marks : 4

Question Label : Multiple Choice Question

Consider the below Graph.



If we run **Depth First Search(DFS)** on the given graph starting from vertex `1`, which of the following is the possible order of visiting the nodes?

Options :

6406532329830. ✖ 1, 4, 3, 6, 5, 2

6406532329831. ✖ 1, 4, 5, 6, 3, 2

6406532329832. ✔ 1, 4, 6, 5, 3, 2

6406532329833. ✖ 1, 4, 6, 3, 5, 2

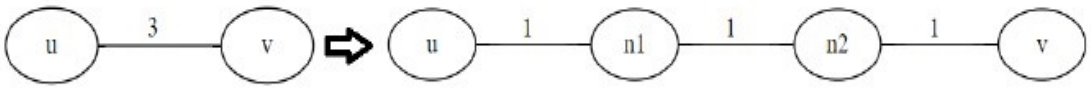
Question Number : 30 Question Id : 640653697606 Question Type : MCQ Is Question Mandatory : No Calculator : None Response Time : N.A Think Time : N.A Minimum Instruction Time : 0

Correct Marks : 4

Question Label : Multiple Choice Question

Consider the following strategy to solve the single source shortest path problem with positive integer edge weights from a source vertex **s**:

Replace each edge in the graph with weight `w` by `w` edges of weight 1 connected by new `w-1` intermediate nodes. For example:



Run **BFS(s)** on the modified graph to find the shortest path to each of the original vertices in the graph.

Which of the following statements is true?

Options :

6406532329839. ✖ This strategy will not solve the problem correctly.

6406532329840. ✖ This strategy will only work if the graph is acyclic.

6406532329841. ✖ This strategy will solve the problem correctly and is as efficient as Dijkstra's algorithm.

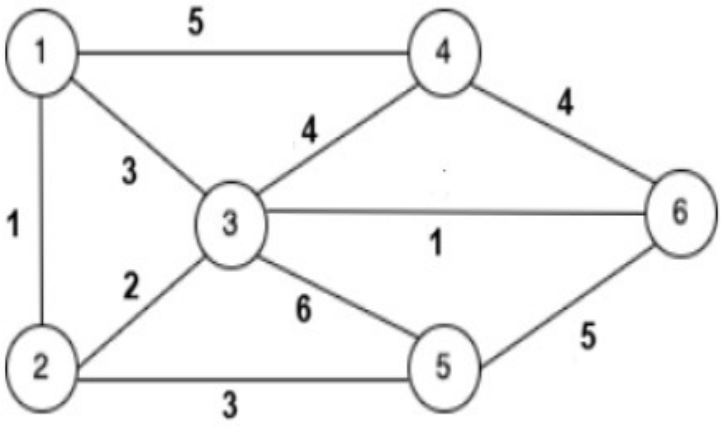
6406532329842. ✔ This strategy will solve the problem correctly, but is not as efficient as Dijkstra's algorithm.

Question Number : 31 Question Id : 640653697608 Question Type : MCQ Is Question Mandatory : No Calculator : None Response Time : N.A Think Time : N.A Minimum Instruction Time : 0

Correct Marks : 4

Question Label : Multiple Choice Question

Consider the following graph



Which of the following is the correct sequence of edges added to the minimum spanning tree when **prim's algorithm** is applied on this graph with `1` as the source vertex?

Options :

6406532329847. ✖ `(1, 2), (3, 6), (2, 3), (2, 5), (3, 4)`

6406532329848. ✔ `(1, 2), (2, 3), (3, 6), (2, 5), (3, 4)`

6406532329849. ✖ `(1, 2), (3, 6), (2, 3), (3, 4), (2, 5)`

6406532329850. ✖ `(1, 2), (2, 3), (3, 6), (3, 4), (2, 5)`

Question Number : 32 Question Id : 640653697610 Question Type : MCQ Is Question Mandatory : No Calculator : None Response Time : N.A Think Time : N.A Minimum Instruction Time : 0

Correct Marks : 4

Question Label : Multiple Choice Question

Let's consider a max heap **[30, 20, 25, 5, 15, 23, 10, 3, 2]**. What would be the final max heap after performing the following operations on a given max heap?

Note:- Each operation generates the max heap after inserting or deleting an element, and the next operation is performed on the updated max heap by the previous operation.

```
delete_max()
delete_max()
insert(40)
```

Options :

6406532329852. ✖ [40, 20, 5, 15, 23, 10, 3, 2]

6406532329853. ✖ [40, 23, 10, 15, 20, 2, 3, 5]

6406532329854. ✖ [40, 23, 20, 15, 10, 2, 3, 5]

6406532329855. ✔ [40, 23, 10, 20, 15, 2, 3, 5]

Question Number : 33 Question Id : 640653697614 Question Type : MCQ Is Question Mandatory : No Calculator : None Response Time : N.A Think Time : N.A Minimum Instruction Time : 0

Correct Marks : 4

Question Label : Multiple Choice Question

Consider the following implementation for Median of Medians(MoM).

```
def MoM(L): # Median of medians
  if len(L) <= 5:
    L.sort()
    return(L[len(L)//2])
  # Construct list of block medians
  M = []
  for i in range(0,len(L),5):
    X = L[i:i+5]
    X.sort()
    M.append(X[len(X)//2])
  return(MoM(M))
```

Let **L = [8, 19, 5, 14, 1, 3, 6, 1, 10, 7, 16, 6, 15, 22, 7, 21, 5, 16, 32, 2]**. What is the returned value of **MoM(L)** using the list **L**?

Options :

6406532329863.

✖ 19

6406532329864. ✔ 15

6406532329865. ✖ 16

6406532329866. ✖ 21

Question Number : 34 Question Id : 640653697615 Question Type : MCQ Is Question Mandatory : No Calculator : None Response Time : N.A Think Time : N.A Minimum Instruction Time : 0

Correct Marks : 4

Question Label : Multiple Choice Question

In a list `L`, two elements `L[i]` and `L[j]` form a **significant inversion** if `L[i]` > `2 * L[j]` and `i < j`. The total number of significant inversions for **L = [1, 11, 6, 3, 5, 2]** is___.

Options :

6406532329867. ✖ 4

6406532329868. ✔ 5

6406532329869. ✖ 6

6406532329870. ✖ 7

Question Number : 35 Question Id : 640653697616 Question Type : MCQ Is Question Mandatory : No Calculator : None Response Time : N.A Think Time : N.A Minimum Instruction Time : 0

Correct Marks : 4

Question Label : Multiple Choice Question

A **subsequence** is a sequence that can be derived from the given sequence by deleting zero or more elements without changing the order of the remaining elements.

Suppose you are given two strings S_1 and S_2:

$S_1 = a_0, a_1, \ldots, a_{n-1}$

$S_2 = b_0, b_1, \ldots, b_{m-1}$

Your task is to find out the length of the longest common subsequence in S_1 and S_2

Consider the following initialization of a two-dimensional array DP of size $n+1, m+1$.

DP	0	1	2	..	j	..	$n-1$	n
0								0
1								0
2								0
..								0
i								0
..								0
..								0
$m-1$								0
m	0	0	0	0	0	0	0	0

Consider that we start at the bottom right ($DP[n-1][m-1]$) and fill DP array row by row or column by column and want to get the length of the longest common subsequence for string S_1 and S_2 at ($DP[0][0]$).

Which of the following inductive structures is correct to fill array DP?

Options :

6406532329871. ✖

$$DP[i,j] = \begin{cases} DP[i+1, j+1], & if \ \ a_i = b_j \\ 1 + min(DP[i+1, j], DP[i, j+1]), & if \ \ a_i \neq b_j \end{cases}$$

6406532329872. ✖

$$DP[i,j] = \begin{cases} DP[i+1, j+1], & if \ \ a_i = b_j \\ 1 + max(DP[i+1, j], DP[i, j+1]), & if \ \ a_i \neq b_j \end{cases}$$

6406532329873. ✖

$$DP[i,j] = \begin{cases} 1 + DP[i+1, j+1], & if \ \ a_i = b_j \\ 1 + min(DP[i+1, j], DP[i, j+1]), & if \ \ a_i \neq b_j \end{cases}$$

6406532329874. ✔

$$DP[i,j] = \begin{cases} 1 + DP[i+1, j+1], & if \ a_i = b_j \\ max(DP[i+1, j], DP[i, j+1]), & if \ a_i \neq b_j \end{cases}$$

Question Number : 36 Question Id : 640653697618 Question Type : MCQ Is Question Mandatory : No Calculator : None Response Time : N.A Think Time : N.A Minimum Instruction Time : 0

Correct Marks : 4

Question Label : Multiple Choice Question

```
def kmp_fail(p):
    m = len(p)
    fail = [0 for i in range(m)]
    j, k = 1, 0
    while j < m:
        if p[j] == p[k]:
            fail[j] = k + 1
            j, k = j + 1, k + 1
        elif k > 0:
            k = fail[k - 1]
        else:
            j = j + 1
    return(fail)
```

Which of the following options represents the fail function (or prefix function) for pattern P = `aabaabaab` returned by the given `kmp_fail(p)` function?

Options :

6406532329876. ✖ `[0, 1, 2, 1, 2, 3, 4, 5, 6]`

6406532329877. ✔ `[0, 1, 0, 1, 2, 3, 4, 5, 6]`

6406532329878. ✖ `[0, 1, 1, 1, 1, 2, 3, 4, 5]`

6406532329879. ✖ `[0, 1, 2, 1, 1, 2, 3, 4, 5]`

Question Number : 37 Question Id : 640653697619 Question Type : MCQ Is Question Mandatory : No Calculator : None Response Time : N.A Think Time : N.A Minimum Instruction Time : 0

Correct Marks : 4

Question Label : Multiple Choice Question

Which of the following combination of input text T and pattern P will exhibit the worst case running time behavior for **Boyer-Moore skipping heuristic**?

Options :

6406532329880. ✖ T = `baabaabaabaabaa` and P = `abb`

6406532329881. ✖ T = `aaaaaaaaaaaaaaa` and P = `abb`

6406532329882. ✔ T = `aaaaaaaaaaaaaaa` and P = `baa`

6406532329883. ✖ T = `aaaaaaaaaaaaaaa` and P = `bbb`

Question Number : 38 Question Id : 640653697620 Question Type : MCQ Is Question Mandatory : No Calculator : None Response Time : N.A Think Time : N.A Minimum Instruction Time : 0

Correct Marks : 4

Question Label : Multiple Choice Question

First kind of cake requires 200g of flour and 25g of fat, and second kind of cake requires 100g of flour and 50g of fat. Formulate this problem as a linear programming problem to find the maximum number of cakes that can be made from 5 kg of flour and 1 kg of fat, assuming that there is no shortage of the other ingredients used in making the cakes.

The above problem is to be formulated as a linear programming problem. Let x and y be the number of cake of kind first and second, respectively. Objective function to maximize the number of cakes $z = x + y$.

Which of the following is **not a valid** constraint for the given problem?

Options :

6406532329884. ✖ 2x + y <= 50

6406532329885. ✔ x + 2y <= 50

6406532329886. ✖ x >= 0 , y >= 0

6406532329887. ✖ x + 2y <= 40

Question Number : 39 Question Id : 640653697621 Question Type : MCQ Is Question Mandatory : No Calculator : None Response Time : N.A Think Time : N.A Minimum Instruction Time : 0

Correct Marks : 4

Question Label : Multiple Choice Question

In a school, there are n teachers and $2n$ subjects. Each subject taught by only one teacher and each teacher is required to teach exactly 2 subjects. However, teachers have their preferences for subjects they would like to teach, and the school wants to maximize overall satisfaction by assigning subjects to teachers based on their preferences.

The preferences of teachers are modeled as a directed graph G, where there exists an edge from a teacher node T_i to a subject node S_j in G if teacher T_i prefers teaching subject S_j. How can this problem be modelled as a network flow problem?

Options :

6406532329888. ✖ It can be modelled as a network flow problem, where the source node is connected to every teacher node in *G* with capacity of n, and every subject node in *G* is connected to the sink node with capacity of 2n.

6406532329889. ✖ It can be modelled as a network flow problem, where the source node is connected to every teacher node in G, and every subject node in *G* is connected to the sink node. All edges in the network flow graph have equal capacity.

6406532329890. ✔ It can be modelled as a network flow problem, where the source node is connected to every teacher node in *G* with capacity of 2, and every subject node in *G* is connected to the sink node with capacity of 1.

6406532329891. ✖ It can be modelled as a network flow problem, where the source node is connected to every teacher node in *G* with capacity of 1, and every subject node in *G* is connected

to the sink node with capacity of 2.

Sub-Section Number : 3

Sub-Section Id : 640653103256

Question Shuffling Allowed : Yes

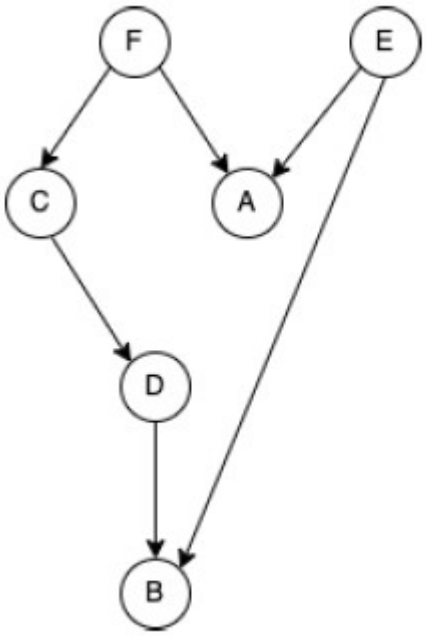
Is Section Default? : null

Question Number : 40 Question Id : 640653697605 Question Type : MSQ Is Question Mandatory : No Calculator : None Response Time : N.A Think Time : N.A Minimum Instruction Time : 0

Correct Marks : 4 Max. Selectable Options : 0

Question Label : Multiple Select Question

Consider the below Graph.



Which of the following is/are **valid** topological orderings of the given graph?

Options :

6406532329834. ✔ E - F - C - D - B - A

6406532329835. ✔ F - E - A - C - D - B

6406532329836. ✖ F - E - A - D - C - B

6406532329837. ✔ E - F - C - D - A - B

6406532329838. ✖ F - C - D - A - E - B

Question Number : 41 Question Id : 640653697607 Question Type : MSQ Is Question Mandatory : No Calculator : None Response Time : N.A Think Time : N.A Minimum Instruction Time : 0

Correct Marks : 4 Max. Selectable Options : 0

Question Label : Multiple Select Question

Which of the following is/are **true** about the **Floyd-Warshall algorithm**?

Options :

6406532329843. ✔ It can detect negative weight cycles in the graph.

6406532329844. ✔ Time complexity of Floyd-Warshall is $O(V^3)$, where V is the number of vertices in the graph.

6406532329845. ✖ It works if the graph has negative edge weight cycles.

6406532329846. ✔ The formula to compute the shortest path from vertex i to j in Floyd-Warshall algorithm is:-

$$SP^k[i,j] = min[SP^{k-1}[i,j], SP^{k-1}[i,k] + SP^{k-1}[k,j]]$$

Question Number : 42 Question Id : 640653697612 Question Type : MSQ Is Question Mandatory : No Calculator : None Response Time : N.A Think Time : N.A Minimum Instruction Time : 0

Correct Marks : 4 Max. Selectable Options : 0

Question Label : Multiple Select Question

Which of the following is/are **true** about **AVL Tree**? Assume that the height of the empty tree is 0.

Options :

6406532329857. ✔ Let `s(h)` denote the minimum number of nodes in an AVL tree of height `h` then `s(h) = s(h-1)+ s(h-2) + 1`, where `s(0) = 0` and `s(1) = 1`.

6406532329858. ✔ In AVL tree, the absolute difference between the height of the left subtree and the height of the right subtree of any node can't be more than 1.

6406532329859. ✖ The complexity of searching element in an AVL tree is $O(n)$.

6406532329860. ✖ If the height of an AVL tree is `h`, the maximum number of nodes will be $2^h + 1$.

6406532329861. ✖ The complexity of both insertion and deletion in AVL tree is $O(n)$.

Sub-Section Number :	4
Sub-Section Id :	640653103257
Question Shuffling Allowed :	Yes
Is Section Default? :	null

Question Number : 43 Question Id : 640653697603 Question Type : SA Calculator : None
Response Time : N.A Think Time : N.A Minimum Instruction Time : 0
Correct Marks : 4

Question Label : Short Answer Question

Consider a simple undirected connected graph *G* with **65** edges with the least number of vertices possible. What will be the number of vertices in graph *G*?

Response Type : Numeric

Evaluation Required For SA : Yes

Show Word Count : Yes

Answers Type : Equal

Text Areas : PlainText

Possible Answers :

12

Question Number : 44 Question Id : 640653697609 Question Type : SA Calculator : None
Response Time : N.A Think Time : N.A Minimum Instruction Time : 0
Correct Marks : 4

Question Label : Short Answer Question

In a binary tree T of **25** nodes, if the number of nodes with two children is **6**, then the number of nodes with one child is ___ .

Response Type : Numeric

Evaluation Required For SA : Yes

Show Word Count : Yes

Answers Type : Equal

Text Areas : PlainText

Possible Answers :

12

Question Number : 45 Question Id : 640653697611 Question Type : SA Calculator : None Response Time : N.A Think Time : N.A Minimum Instruction Time : 0

Correct Marks : 4

Question Label : Short Answer Question

The **pre-order traversal** of a binary search tree is:

4, 1, 3, 2, 7, 5, 6, 8

What would be the sum of elements stored in the leaf nodes of a binary search tree?

Response Type : Numeric

Evaluation Required For SA : Yes

Show Word Count : Yes

Answers Type : Equal

Text Areas : PlainText

Possible Answers :

16

Question Number : 46 Question Id : 640653697613 Question Type : SA Calculator : None Response Time : N.A Think Time : N.A Minimum Instruction Time : 0

Correct Marks : 4

Question Label : Short Answer Question

You are given the following list of **8** meeting requests with start time, and end time.

Meeting Id	Start Time	End Time
1	1	4
2	1	2
3	5	8
4	7	10
5	6	8
6	11	16
7	15	20
8	14	17

Each meeting requires its own conference room. Your goal is to schedule all meetings in the minimum number of conference rooms. Assume that if any meeting ends at time `t` then another meeting can start at time `t` or afterwards in the same room.

The minimum number of conference rooms required to schedule all meetings is___.

Response Type : Numeric

Evaluation Required For SA : Yes

Show Word Count : Yes

Answers Type : Equal

Text Areas : PlainText

Possible Answers :

3

Question Number : 47 Question Id : 640653697617 Question Type : SA Calculator : None

Response Time : N.A Think Time : N.A Minimum Instruction Time : 0

Correct Marks : 4

Question Label : Short Answer Question

There are N stones, numbered $0, 1, 2, \ldots, N-1$. For each $i (0 \leq i \leq N-1)$, the height of Stone i is h_i .

There is a frog who is initially on Stone 0. He will repeat the following action some number of times to reach Stone N

If the frog is currently on stone i, can jump to Stone $i+1$ or Stone $i+2$. Here, a cost of $| h_i - h_j |$ is incurred, where j is the stone to land on.

Find the minimum possible total cost to reach stone 5 from stone 0 for the following sequence of heights for 6 stones.

```
10, 15, 30, 20, 28, 36
```

Response Type : Numeric

Evaluation Required For SA : Yes

Show Word Count : Yes

Answers Type : Equal

Text Areas : PlainText

Possible Answers :

26

AppDev1

Section Id :	64065349262
Section Number :	3
Section type :	Online
Mandatory or Optional :	Mandatory
Number of Questions :	31
Number of Questions to be attempted :	31
Section Marks :	100
Display Number Panel :	Yes
Section Negative Marks :	0
Group All Questions :	No
Enable Mark as Answered Mark for Review and Clear Response :	Yes