Number of Questions :	17
Number of Questions to be attempted :	17
Section Marks :	50
Display Number Panel :	Yes
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 :	64065382585
Question Shuffling Allowed :	No
Is Section Default? :	null

Question Number : 85 Question Id : 640653577836 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 :

6406531929587. ✔ YES

6406531929588. ✖ NO

Sub-Section Number :	2
Sub-Section Id :	64065382586

Question Shuffling Allowed : Yes

Is Section Default? : null

Question Number : 86 Question Id : 640653577837 Question Type : MSQ Is Question Mandatory : No Calculator : None Response Time : N.A Think Time : N.A Minimum Instruction Time : 0

Correct Marks : 3 Max. Selectable Options : 0

Question Label : Multiple Select Question

Consider the following functions:

- $f(n) = 102n^4 + 26n^3$
- $g(n) = 103n^3 + 20n^2$
- $h(n) = 110n^3 \log n + 36n^2$

Which of the following is/are true?

Options :

6406531929589. $f(n) = O(g(n))$

6406531929590. $g(n) = O(h(n))$

6406531929591. $f(n) = O(h(n))$

6406531929592. $h(n) = O(g(n))$

6406531929593. $h(n) = O(f(n))$

Question Number : 87 Question Id : 640653577842 Question Type : MSQ Is Question Mandatory : No Calculator : None Response Time : N.A Think Time : N.A Minimum Instruction Time : 0

Correct Marks : 3 Max. Selectable Options : 0

Question Label : Multiple Select Question

Unimodal List: A list `L[0 . . . n-1]` of distinct elements is *unimodal* if it consists of an increasing sequence followed by a decreasing sequence. More precisely, there is an index `m ∈ 1, 2, . . . , n-2` such that:

- `L[i] < L[i + 1]` for all `0 <= i < m`, and
- `L[i] > L[i + 1]` for all `m <= i < n-1`.

Suppose the middle element of a unimodal list is `x`, and the elements to the left and right of `x` are `p` and `q`, respectively. Which of the following facts must be used to find the maximum element in $O(\log n)$ time?

Options :

6406531929610. If `p < x > q`, then `x` is the maximum in the list.

6406531929611. If `p < x < q`, then the maximum element is in the left half of the list.

6406531929612. If `p < x < q`, then the maximum element is in the right half of the list.

6406531929613. If `p > x > q`, then the maximum element is in the left half of the list.

6406531929614. If `p > x > q`, then the maximum element is in the right half of the list.

Question Number : 88 Question Id : 640653577845 Question Type : MSQ Is Question Mandatory : No Calculator : None Response Time : N.A Think Time : N.A Minimum Instruction Time : 0

Correct Marks : 3 Max. Selectable Options : 0

Question Label : Multiple Select Question

```python
class Node:
    def __init__(self,data):
        self.data = data
        self.next = None
```

Consider an implementation of a **singly linked list**, where each node is created using the given class `Node`. Suppose it has only a `head` pointer that points to the first node of the linked list.

Which of the following statement(s) is/are **true**? Assume we are using the most efficient algorithms.

Options :

6406531929619. Finding an item in a sorted linked list of n items takes $O(n)$ time.

6406531929620. Finding an item in a sorted linked list of n items takes $O(logn)$ time

6406531929621. Adding a new item to the end of the linked list of n items takes $O(n)$ time.

6406531929622. Removing an item from the end of the linked list of n items takes $O(1)$ time.

6406531929623. Removing duplicate items from the sorted linked list of n items takes $O(n)$ time.

Question Number : 89 Question Id : 640653577847 Question Type : MSQ Is Question Mandatory : No Calculator : None Response Time : N.A Think Time : N.A Minimum Instruction Time : 0

Correct Marks : 3 Max. Selectable Options : 0

Question Label : Multiple Select Question

Linear probing is an open addressing scheme in computer programming for resolving hash collisions in hash tables. Linear probing takes the original hash index and increments the value by 1 until a free slot is found.

Consider the given hash table with hash function `h(key) = key mod 5` which uses linear probing for solving collisions.

Index	Key
0	45
1	51
2	60
3	18
4	34

Which among the following options corresponds to possible orders of insertion of values in the hash table?

Options :

6406531929628. 51, 18, 45, 60, 34

6406531929629. 34, 45, 18, 60, 51

6406531929630. 18, 45, 34, 60, 51

6406531929631. 34, 45, 18, 51, 60

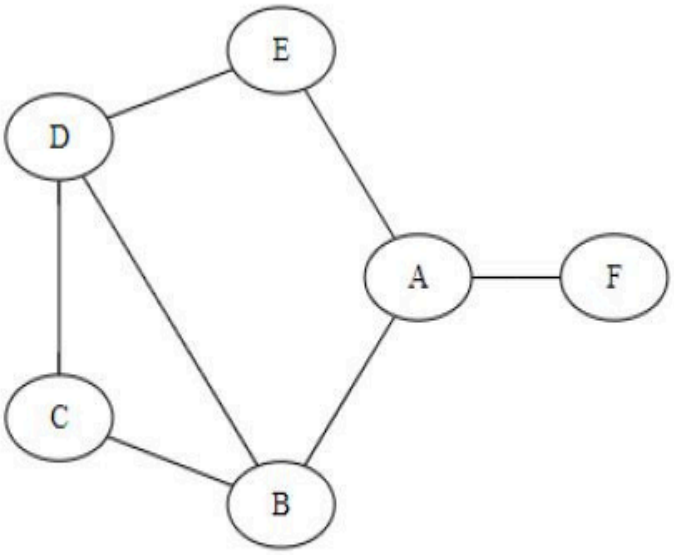
6406531929632. 18, 34, 51, 45, 60

Question Number : 90 Question Id : 640653577850 Question Type : MSQ Is Question Mandatory : No Calculator : None Response Time : N.A Think Time : N.A Minimum Instruction Time : 0

Correct Marks : 3 Max. Selectable Options : 0

Question Label : Multiple Select Question

Consider the following graph:



If we run breadth first search(BFS) on the given graph starting at any vertex, which of the following is/are possible order of visiting the nodes?

Note : When a node has multiple neighbours, BFS would visit alphabetically.

Options :

6406531929639. A B E C D F

6406531929640. B A C D E F

6406531929641. C B D A E F

6406531929642. D B C E A F

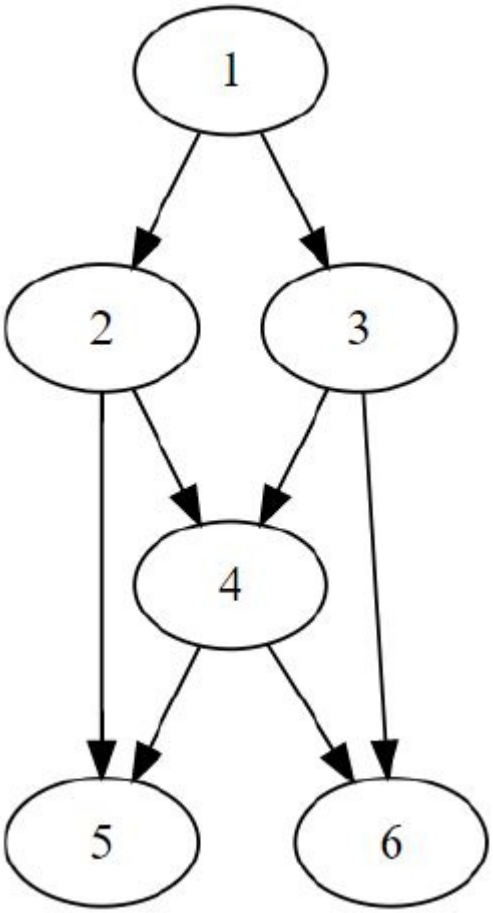
6406531929643. E A B D F C

Question Number : 91 Question Id : 640653577852 Question Type : MSQ Is Question Mandatory : No Calculator : None Response Time : N.A Think Time : N.A Minimum Instruction Time : 0

Correct Marks : 3 Max. Selectable Options : 0

Question Label : Multiple Select Question

Consider the following DAG



Which of the following is/are **not a valid** topological orderings for the given DAG?

Options :

6406531929648. `1 2 3 4 5 6`

6406531929649. `1 3 4 2 5 6`

6406531929650. `1 3 2 4 5 6`

6406531929651. `1 3 2 5 4 6`

6406531929652. `1 2 4 3 5 6`

Sub-Section Number :	3
Sub-Section Id :	64065382587
Question Shuffling Allowed :	Yes

Is Section Default? : null

Question Number : 92 Question Id : 640653577838 Question Type : MCQ Is Question Mandatory : No Calculator : None Response Time : N.A Think Time : N.A Minimum Instruction Time : 0

Correct Marks : 3

Question Label : Multiple Choice Question

```
def fun(n):
    count = 0
    for i in range(n):
        j = 1
        while j < n:
            count += 1
            j *= 2
    return count
```

What is the time complexity of the function `fun` given above?

Options :

6406531929594. $O(1)$

6406531929595. $O(n)$

6406531929596. $O(n \log n)$

6406531929597. $O(n^2)$

Question Number : 93 Question Id : 640653577839 Question Type : MCQ Is Question Mandatory : No Calculator : None Response Time : N.A Think Time : N.A Minimum Instruction Time : 0

Correct Marks : 3

Question Label : Multiple Choice Question

We have an input list of three-dimensional points `[(7, 8, 1), (3, 7, 5), (6, 8, 1), (6, 7, 5), (0, 5, 2), (9, 9, 0)]`. We sort these in ascending order by the third coordinate. Which of the following corresponds to a stable sort of this input?

Options :

6406531929598. `[(9, 9, 0), (6, 8, 1), (7, 8, 1), (0, 5, 2), (6, 7, 5), (3, 7, 5)]`

6406531929599. `[(9, 9, 0), (7, 8, 1), (6, 8, 1), (0, 5, 2), (3, 7, 5), (6, 7, 5)]`

6406531929600. `[(9, 9, 0), (6, 8, 1), (7, 8, 1), (0, 5, 2), (3, 7, 5), (6, 7, 5)]`

6406531929601. `[(9, 9, 0), (7, 8, 1), (6, 8, 1), (0, 5, 2), (6, 7, 5), (3, 7, 5)]`

Question Number : 94 Question Id : 640653577840 Question Type : MCQ Is Question Mandatory : No Calculator : None Response Time : N.A Think Time : N.A Minimum Instruction Time : 0

Correct Marks : 3

Question Label : Multiple Choice Question

Consider the following Implementation for insertion sort

```
1  def insertionsort(L):
2      n = len(L)
3      if n < 1:
4          return(L)
5      for i in range(n):
6          j = i
7          while(j > 0 and L[j] < L[j-1]):
8              (L[j],L[j-1]) = (L[j-1],L[j])
9              j = j-1
10     return(L)
```

Suppose L is a list of distinct integer elements. Let x, y and z be the largest, second largest, and third largest elements in the list L. Suppose z appears before x in the list. Which of the following is true, with respect to the implementation above?

Options :

6406531929602. x and z are always compared in a run of insertion sort, regardless of the position of y.

6406531929603. x and z are compared in a run of insertion sort if and only if y appears before z in the list L.

6406531929604. x and z are compared in a run of insertion sort if and only if y appears after x in the list L.

6406531929605. x and z are compared in a run of insertion sort if and only if y appears after z but before x in the list L.

Question Number : 95 Question Id : 640653577841 Question Type : MCQ Is Question Mandatory : No Calculator : None Response Time : N.A Think Time : N.A Minimum Instruction Time : 0

Correct Marks : 3

Question Label : Multiple Choice Question

3-way-Merge Sort: Suppose that instead of dividing the input list L in half at each step of Merge Sort, you divide L into three equal parts, sort each parts, and finally combine all of them using an efficient three-way merge (merge three sorted lists instead of two).

What is the overall asymptotic running time of the **3-way-Merge Sort** algorithm?

Options :

6406531929606. $O(n^2)$

6406531929607. $O(n^2 log\, n)$

6406531929608. $O(n(log\, n)^2)$

6406531929609. $O(n\, log\, n)$

Question Number : 96 Question Id : 640653577844 Question Type : MCQ Is Question Mandatory : No Calculator : None Response Time : N.A Think Time : N.A Minimum Instruction Time : 0

Correct Marks : 3

Question Label : Multiple Choice Question

Which of the following statements is **not true** about Quicksort?

Options :

6406531929616. For every fixed strategy to choose a pivot for Quicksort, we can construct a worst-case input that requires time $O(n^2)$.

6406531929617. If we could find the median in time $O(n)$, Quicksort would have worst-case complexity $O(n\ log\ n)$

6406531929618. If we randomly choose a pivot element each time, Quicksort will always terminate in time $O(n\ log\ n)$.

Question Number : 97 Question Id : 640653577846 Question Type : MCQ Is Question Mandatory : No Calculator : None Response Time : N.A Think Time : N.A Minimum Instruction Time : 0

Correct Marks : 3

Question Label : Multiple Choice Question

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, and `isEmpty()` is a method that returns true if either the stack or the queue is empty. Assume that stack `S` and Queue `Q` are empty initially.

```
for i in range(5):
    S.Push(i)
    Q.Enqueue(i)

while not Q.isEmpty():
    S.Push(Q.Dequeue())

while not S.isEmpty():
    Q.Enqueue(S.Pop())

while not Q.isEmpty():
    print (Q.Dequeue(),end = " ")
```

What is the output of the given code snippet?

Options :

6406531929624. `0 1 2 3 4 4 3 2 1 0`

6406531929625. `4 3 2 1 0 0 1 2 3 4`

6406531929626. `4 3 2 1 0 4 3 2 1 0`

6406531929627. `0 1 2 3 4 0 1 2 3 4`

Question Number : 98 Question Id : 640653577849 Question Type : MCQ Is Question Mandatory : No Calculator : None Response Time : N.A Think Time : N.A Minimum Instruction Time : 0

Correct Marks : 3

Question Label : Multiple Choice Question

Consider a graph G. Let T be a BFS tree of G with root r. Let $d(r, v)$ denote the length of the shortest path between the nodes r and v. If vertex x is visited before vertex y in the breadth first search traversal, which of the following statements is true?

Options :

6406531929634. $d(r, x) > d(r, y)$

6406531929635. $d(r, x) = d(r, y)$

6406531929636. $d(r, x) < d(r, y)$

6406531929637. $d(r, x) \leq d(r, y)$

6406531929638. $d(r, x) \geq d(r, y)$

Question Number : 99 Question Id : 640653577851 Question Type : MCQ Is Question Mandatory : No Calculator : None Response Time : N.A Think Time : N.A Minimum Instruction Time : 0

Correct Marks : 3

Question Label : Multiple Choice Question

Consider the following statements:

1. While creating a DFS tree for a directed graph, among non-tree edges, only back edges correspond to cycles.
2. The depth of any DFS tree rooted at a vertex is at least as much as the depth of any BFS tree rooted at the same vertex.

Choose the correct option.

Options :

6406531929644. Only statement 1 is true

6406531929645. Only statement 2 is true

6406531929646. Both statements 1 and 2 are true

6406531929647. Both statements 1 and 2 are false

Sub-Section Number : 4

Sub-Section Id : 64065382588

Question Shuffling Allowed : Yes

Is Section Default? : null

Question Number : 100 Question Id : 640653577843 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 the given list `L = [9, 14, 17, 37, 57, 62, 82, 92, 97]` . After applying the **Quick-sort partition** algorithm once, the list is modified to : `[14, 9, 17, 37, 62, 57, 82, 97, 92]` .

The number of elements that could have been chosen as a pivot in the first round is ___ ?

Response Type : Numeric

Evaluation Required For SA : Yes

Show Word Count : Yes **Answers**

Type : Equal **Text Areas :**

PlainText **Possible Answers :**

Question Number : 101 Question Id : 640653577848 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

A connected, simple, undirected graph *G* has 1225 edges. The minimum number of vertices in G is _____.

Response Type : Numeric

Evaluation Required For SA : Yes

Show Word Count : Yes

Answers Type : Equal

Text Areas : PlainText

Possible Answers :

AppDev1

Section Id :	64065339072
Section Number :	7 Online
Section type :	Mandatory
Mandatory or Optional :	16 16 50 Yes
Number of Questions :	No
Number of Questions to be attempted :	
Section Marks :	
Display Number Panel :	
Group All Questions :	
Enable Mark as Answered Mark for Review and Clear Response :	Yes
Maximum Instruction Time :	0
Sub-Section Number :	1
Sub-Section Id :	64065382589
Question Shuffling Allowed :	No
Is Section Default? :	null

Question Number : 102 Question Id : 640653577853 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 : MODERN APPLICATION