6406531561894. ✖ Name of the designation with salary more than 80000.

# PDSA

| | |
|---|---|
| **Section Id :** | 64065330332 |
| **Section Number :** | 6 |
| **Section type :** | Online |
| **Mandatory or Optional :** | Mandatory |
| **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 :** | 64065367674 |
| **Question Shuffling Allowed :** | No |
| **Is Section Default? :** | null |

**Question Number : 86 Question Id : 640653470012 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"**

**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 :**

6406531561911. ✔ YES

6406531561912. ✖ NO

| | |
|---|---|
| **Sub-Section Number :** | 2 |
| **Sub-Section Id :** | 64065367675 |
| **Question Shuffling Allowed :** | Yes |
| **Is Section Default? :** | null |

**Question Number : 87 Question Id : 640653470013 Question Type : MCQ Is Question Mandatory : No Calculator : None Response Time : N.A Think Time : N.A Minimum Instruction Time : 0**

**Correct Marks : 2**

Question Label : Multiple Choice Question

$f1(n) = 3n^2 + 2n$

$f2(n) = 3n + (\log n)^2$

$f3(n) = \log(\log n)$

$f4(n) = 10 \log n$

$f5(n) = 3n \log n$

Arrange the above functions in increasing order of asymptotic complexity.

**Options :**

6406531561913. ✖ $f3(n), f4(n), f2(n), f1(n), f5(n)$

6406531561914. ✖ $f3(n), f2(n), f1(n), f5(n), f4(n)$

6406531561915. ✖ $f4(n), f3(n), f2(n), f1(n), f5(n)$

6406531561916. ✔ $f3(n), f4(n), f2(n), f5(n), f1(n)$

**Sub-Section Number :** 3

**Sub-Section Id :** 64065367676

**Question Shuffling Allowed :** Yes

**Is Section Default? :** null

**Question Number : 88 Question Id : 640653470014 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

Given the following sorted list :

```
[16, 53, 59, 81, 94, 99, 121, 150, 162, 170]
```

If we use binary search algorithm to search the element `99` in the list, then which of the following option corresponds to the correct sequence of comparison done in this process ?

*Note: Assume here binary search will compute the midpoint by using* $(firstindex + lastindex)//2$

**Options :**

6406531561917. ✖ `94, 99`

6406531561918. ✖ `16, 99`

6406531561919. ✖ `94, 150, 121, 99`

6406531561920. ✔ `94, 150, 99`

6406531561921. ✖ None of these

**Question Number : 89 Question Id : 640653470015 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

A list of `n` strings, each of length `n` is sorted in **lexicographical order** using the Merge Sort algorithm. What is its time complexity? (Assume that comparing strings lexicographically takes $O(n)$)

**Options :**

6406531561922. ✖ $O(nlogn)$

6406531561923. ✔ $O(n^2logn)$

6406531561924. ✖ $O(n^2)$

6406531561925. ✖ $O(logn)$

**Question Number : 90 Question Id : 640653470016 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

4 sorted lists each of length $n/2$ are merged into a single sorted list of $2n$ elements using two way merging. What will be the minimum number of element comparisons needed for this process ?

**Options :**

6406531561926. ✖ $n-1$

6406531561927. ✖ $2n-1$

6406531561928. ✔ $4n-3$

6406531561929. ✖ $4n-1$

**Question Number : 91 Question Id : 640653470017 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

Select the most appropriate data structure for the following applications.

| Application | Data Structure |
|---|---|
| 1. To implement undo - redo operations in a text editor | a. Array |
| 2. Matrix operations | b. Graph |
| 3. To implement a music playlist feature which plays songs in sequence | c. Stack |
| 4. To represent communication networks | d. Queue |

**Options :**

6406531561930. ✖ 1-d, 2-a, 3-c, 4-b

6406531561931. ✖ 1-d, 2-b, 3-c, 4-a

6406531561932. ✖ 1-d, 2-a, 3-b, 4-c

6406531561933. ✔ 1-c, 2-a, 3-d, 4-b

**Question Number : 92 Question Id : 640653470019 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 that **Quick sort** is applied on a list of size `n` which is already sorted. What will be the asymptotic running time of Quick sort if the pivot is taken to be

$I)$ Middle element $II)$ Last element

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

**Options :**

6406531561935. ✖ $I : O(n^2)$ and $II : O(n \log n)$

6406531561936. ✖ $I : O(n^2)$ and $II : O(n)$

6406531561937. ✖ $I : O(n \log n)$ and $II : O(n \log n)$

6406531561938. ✔ $I : O(n \log n)$ and $II : O(n^2)$

**Question Number : 93 Question Id : 640653470020 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 which returns true if either the stack or the queue are empty.
Now consider the function `fun` given below:

```
def fun(Q):
    while(!Q.isEmpty()):
        temp = Q.Dequeue()
        S.Push(temp)
    while(!S.isEmpty()):
        temp = S.Pop()
        Q.Enqueue(temp)
```

Suppose the initial content of the queue `Q` is `[26, 78, 45, 10, 19, 56]` and the stack `S` is empty initially . If `fun` is invoked on `Q` then what will be the content of `Q` after `fun` finishes its execution?

**Options :**

6406531561939. ✖ `[26, 78, 45, 10, 19, 56]`

6406531561940. ✖ `[56, 19, 10, 26, 78, 45]`

6406531561941. ✔ `[56, 19, 10, 45, 78, 26]`

6406531561942. ✖ None of these

**Question Number : 94 Question Id : 640653470022 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 function:

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

Which of the following option correctly mentions the return value ( `total` ) of function `fun` in terms of $n$ ? Consider that input $n = 2^k$ where $k$ is a positive integer.

**Options :**

6406531561948. ✖ $n^2$

6406531561949. ✖ $(\log n)^2/4$

6406531561950. ✖ $(n \log n)/2$

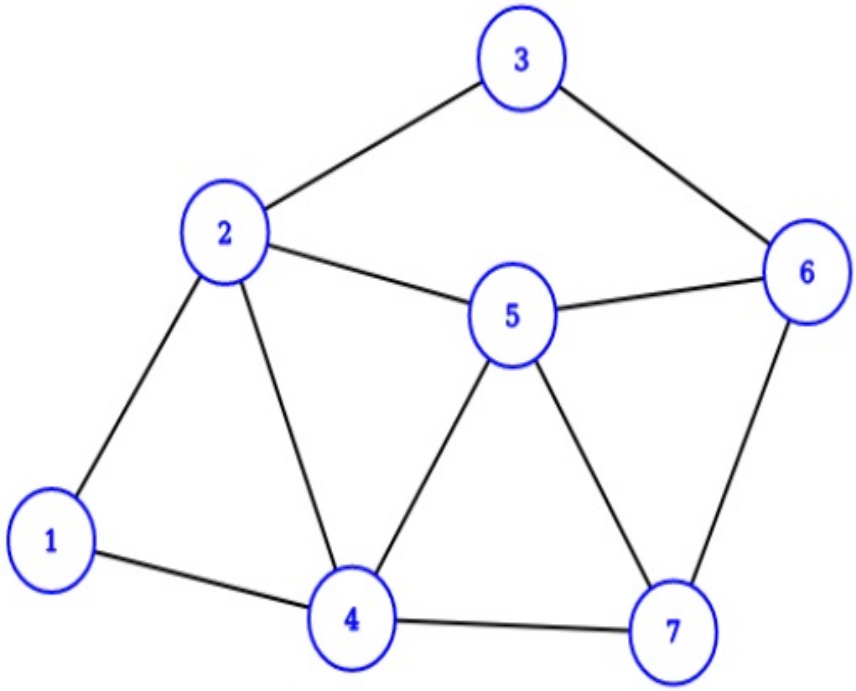
6406531561951. ✔ $(n^2 \log n)/4$

**Question Number : 95 Question Id : 640653470025 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 graph




Which of the following vertex sequence/(s) are possible **BFS traversals** on the graph started from node **5**?

*Note : When a node has multiple neighbours, BFS would visit the numerically smaller valued node first.*

**Options :**

6406531561957. ✔ 5,2,4,6,7,1,3

6406531561958. ✖ 5,2,3,1,4,7,6

6406531561959. ✖ 5,2,1,4,7,6,3

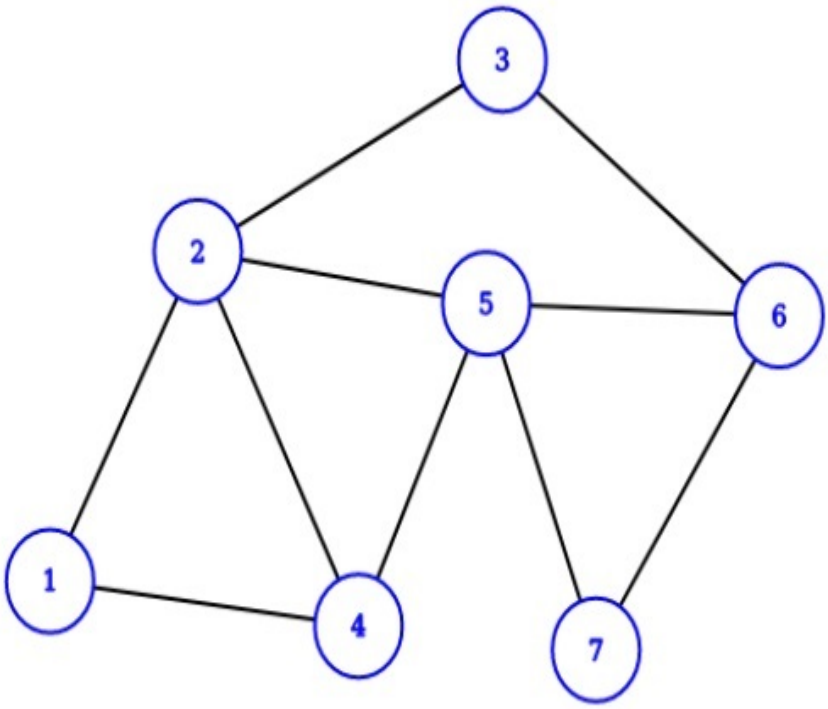
6406531561960. ✖ 5,2,1,3,4,6,7

**Question Number : 96 Question Id : 640653470026 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 graph




**DFS algorithm** is applied starting from vertex **5** on the given graph. What will be the maximum possible height till which the stack will grow while making the DFS traversal ?

*Note : When a node has multiple neighbours, DFS would visit the numerically smaller valued node first.*

**Options :**

6406531561961. ✖ 3

6406531561962. ✖ 4

6406531561963. ✔ 5

6406531561964. ✖ 6

6406531561965. ✖ 7

| | |
|---|---|
| **Sub-Section Number :** | 4 |
| **Sub-Section Id :** | 64065367677 |
| **Question Shuffling Allowed :** | Yes |
| **Is Section Default? :** | null |

**Question Number : 97 Question Id : 640653470018 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 following input list:

`[38, 28, 43, 22, 112, 33, 39]`

What will be the number of swaps that the following **Insertion sort** will make to sort this given list?

```
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])
            j = j-1
    return(L)
```

**Response Type :** Numeric

**Evaluation Required For SA :** Yes

**Show Word Count :** Yes

**Answers Type :** Equal

**Text Areas :** PlainText

**Possible Answers :**

9

**Question Number : 98 Question Id : 640653470023 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

**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.

A hash table contains 8 buckets indexed from 0 to 7 and uses linear probing to resolve collisions. The key values are integers and the hash function used is `key mod 8`. If `key` values 91, 27, 16, 64, 41 are inserted in to the table in the given order, in what location would the key value 160 be inserted after them?

**Response Type :** Numeric

**Evaluation Required For SA :** Yes

**Show Word Count :** Yes

**Answers Type :** Equal

**Text Areas :** PlainText

**Possible Answers :**

5

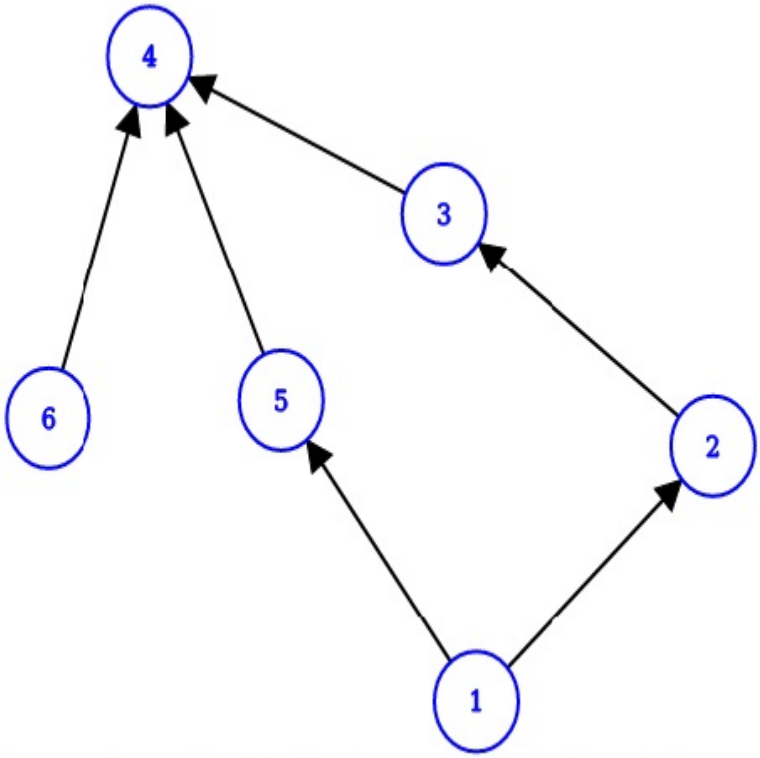
**Question Number : 99 Question Id : 640653470027 Question Type : SA Calculator : None**

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

**Correct Marks : 4**

Question Label : Short Answer Question

Consider the following graph




The number of different **topological orderings** possible for this DAG starting from vertex **1** is ___.

**Response Type :** Numeric

**Evaluation Required For SA :** Yes

**Show Word Count :** Yes

**Answers Type :** Equal

**Text Areas :** PlainText

**Possible Answers :**

12

**Sub-Section Number :** 5

**Sub-Section Id :** 64065367678

**Question Shuffling Allowed :** Yes

**Is Section Default? :** null

**Question Number : 100 Question Id : 640653470021 Question Type : MSQ Is Question Mandatory : No Calculator : None Response Time : N.A Think Time : N.A Minimum Instruction Time : 0**

**Correct Marks : 3 Selectable Option : 0**

Question Label : Multiple Select Question

```
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 a `head` pointer that points to the first node of the linked list and a `tail` pointer that points to the last element of the linked list.

Which of the following operation can be implemented in constant (**O(1)**) time with given representation of the linked list ?

**Options :**

6406531561943. ✔ Insertion of the new node at the front of the linked list.

6406531561944. ✔ Insertion of the new node at the end of the linked list.

6406531561945. ✔ Deletion of the first node of the linked list.

6406531561946. ✖ Deletion of the last node of the linked list.

6406531561947. ✔ Deletion of the second node (from starting) of the linked list

**Question Number : 101 Question Id : 640653470024 Question Type : MSQ Is Question Mandatory : No Calculator : None Response Time : N.A Think Time : N.A Minimum Instruction Time : 0**

**Correct Marks : 3 Selectable Option : 0**

Question Label : Multiple Select Question

Which of the following is/are possible degree sequence/(s) of vertices of a connected undirected graph with six vertices?

Note: Degree sequence is a series of positive integer $a_1, a_2, \ldots, a_n$ where each $a_i$ is the degree of the $i^{th}$ vertex of the graph.

**Options :**

6406531561953. ✖ 1,1,1,1,1,1

6406531561954. ✔ 2,2,2,2,2,2

6406531561955. ✔ 1,1,2,2,3,5

6406531561956. ✖ 1,2,2,2,3,3

**Question Number : 102 Question Id : 640653470028 Question Type : MSQ Is Question Mandatory : No Calculator : None Response Time : N.A Think Time : N.A Minimum Instruction Time : 0**

**Correct Marks : 3 Selectable Option : 0**

Question Label : Multiple Select Question

Which of the following statement(s) is/are **true** about **Depth First Search (DFS)** on an undirected and connected graph?

**Options :**

6406531561967. ✔ DFS can be used to detect cycles in the graph.

6406531561968. ✔ DFS can be used to identify connected components in an undirected graph.

6406531561969. ✔ Using an adjacency list instead of an adjacency matrix can improve the worst case complexity to $O(V+E)$, where $V$ is number of vertices and $E$ is number of edges.

6406531561970. ✖ In an unweighted graph, DFS can be used to identify the shortest path from a starting vertex `s` to every other vertex in the graph

6406531561971. ✔ DFS always produces the same number of tree edges irrespective of the order in which the vertices are considered for DFS.

# AppDev1

| | |
|---|---|
| **Section Id :** | 64065330333 |
| **Section Number :** | 7 |
| **Section type :** | Online |
| **Mandatory or Optional :** | Mandatory |
| **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 :** | 64065367679 |
| **Question Shuffling Allowed :** | No |
| **Is Section Default? :** | null |

**Question Number : 103 Question Id : 640653470029 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 DEVELOPMENT I"**

**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**