

OUTPUT:

Program 1:

```
C:\Users\admin\Desktop\1BM X + v
--- Stack Operations ---
1. Push
2. Pop
3. Display
4. Exit
Enter your choice: 1
Enter the value to push: 5
5 pushed into stack.

--- Stack Operations ---
1. Push
2. Pop
3. Display
4. Exit
Enter your choice: 1
Enter the value to push: 6
6 pushed into stack.

--- Stack Operations ---
1. Push
2. Pop
3. Display
4. Exit
Enter your choice: 3
Stack elements are:
6
5

--- Stack Operations ---
1. Push
2. Pop
3. Display
4. Exit
Enter your choice: 2
6 popped from stack.

--- Stack Operations ---
1. Push
2. Pop
3. Display
4. Exit
Enter your choice: 3
Stack elements are:
5
```

Program 2:

```
C:\Users\admin\Desktop\1BM X + v
Enter infix expression: A+(B*C)
Postfix expression: ABC*+

Process returned 0 (0x0)   execution time : 20.963 s
Press any key to continue.
|
```

Program 3a:

```
C:\Users\admin\Desktop\1BM X + v

--- Queue Operations ---
1. Insert
2. Delete
3. Display
4. Exit
Enter your choice: 1
Enter the value to insert: 5
5 inserted into queue.

--- Queue Operations ---
1. Insert
2. Delete
3. Display
4. Exit
Enter your choice: 1
Enter the value to insert: 6
6 inserted into queue.

--- Queue Operations ---
1. Insert
2. Delete
3. Display
4. Exit
Enter your choice: 3
Queue elements are:
5 6

--- Queue Operations ---
1. Insert
2. Delete
3. Display
4. Exit
Enter your choice: 2
5 deleted from queue.

--- Queue Operations ---
1. Insert
2. Delete
3. Display
4. Exit
Enter your choice: 3
Queue elements are:
6
```

Program 3b:

```
C:\Users\admin\Desktop\1BM X + v

--- Circular Queue Operations ---
1. Insert
2. Delete
3. Display
4. Exit
Enter your choice: 1
Enter the value to insert: 5
5 inserted into queue.

--- Circular Queue Operations ---
1. Insert
2. Delete
3. Display
4. Exit
Enter your choice: 1
Enter the value to insert: 6
6 inserted into queue.

--- Circular Queue Operations ---
1. Insert
2. Delete
3. Display
4. Exit
Enter your choice: 3
Queue elements are:
5 6

--- Circular Queue Operations ---
1. Insert
2. Delete
3. Display
4. Exit
Enter your choice: 2
5 deleted from queue.

--- Circular Queue Operations ---
1. Insert
2. Delete
3. Display
4. Exit
Enter your choice: 3
Queue elements are:
6
```

Program 4:

```
C:\Users\admin\Desktop\IBM X + v

--- Singly Linked List Menu ---
1. Create Linked List
2. Insert at Beginning
3. Insert at End
4. Insert at Position
5. Display
6. Exit
Enter choice: 1
Enter number of nodes: 2
Enter data: 5
Enter data: 6

--- Singly Linked List Menu ---
1. Create Linked List
2. Insert at Beginning
3. Insert at End
4. Insert at Position
5. Display
6. Exit
Enter choice: 2
Enter data: 9

--- Singly Linked List Menu ---
1. Create Linked List
2. Insert at Beginning
3. Insert at End
4. Insert at Position
5. Display
6. Exit
Enter choice: 3
Enter data: 8

--- Singly Linked List Menu ---
1. Create Linked List
2. Insert at Beginning
3. Insert at End
4. Insert at Position
5. Display
6. Exit
Enter choice: 4
Enter position: 3
Enter data: 7
```

```
--- Singly Linked List Menu ---
1. Create Linked List
2. Insert at Beginning
3. Insert at End
4. Insert at Position
5. Display
6. Exit
Enter choice: 5
Linked list: 9 -> 5 -> 7 -> 6 -> 8 -> NULL
```

Program 5:

```
C:\Users\admin\Desktop\1BM X + v

--- Singly Linked List Menu ---
1. Create List
2. Delete First
3. Delete Specified
4. Delete Last
5. Display
6. Exit
Enter choice: 1
Enter data: 5
Add another node? (1/0): 1
Enter data: 6
Add another node? (1/0): 1
Enter data: 7
Add another node? (1/0): 1
Enter data: 8
Add another node? (1/0): 1
Enter data: 9
Add another node? (1/0): 0

--- Singly Linked List Menu ---
1. Create List
2. Delete First
3. Delete Specified
4. Delete Last
5. Display
6. Exit
Enter choice: 2
First node deleted

--- Singly Linked List Menu ---
1. Create List
2. Delete First
3. Delete Specified
4. Delete Last
5. Display
6. Exit
Enter choice: 3
Enter element to delete: 2
Element not found

--- Singly Linked List Menu ---
1. Create List
2. Delete First
3. Delete Specified
4. Delete Last
5. Display
6. Exit
Enter choice: 4
Last node deleted
```

```

--- Singly Linked List Menu ---
1. Create List
2. Delete First
3. Delete Specified
4. Delete Last
5. Display
6. Exit
Enter choice: 3
Enter element to delete: 7
Node deleted

--- Singly Linked List Menu ---
1. Create List
2. Delete First
3. Delete Specified
4. Delete Last
5. Display
6. Exit
Enter choice: 5
Linked List: 6 -> 8 -> NULL

```

Program 6a:

```

C:\Users\admin\Desktop\1BM X
Create First Linked List
Enter data: 3
Add another node? (1/0): 1
Enter data: 8
Add another node? (1/0): 1
Enter data: 1
Add another node? (1/0): 0

Create Second Linked List
Enter data: 2
Add another node? (1/0): 1
Enter data: 4
Add another node? (1/0): 0

--- Menu ---
1. Display List 1
2. Sort List 1
3. Reverse List 1
4. Concatenate List 1 and List 2
5. Exit
Enter choice: 1
3 -> 8 -> 1 -> NULL

```

```
--- Menu ---
1. Display List 1
2. Sort List 1
3. Reverse List 1
4. Concatenate List 1 and List 2
5. Exit
Enter choice: 2
List sorted
1 -> 3 -> 8 -> NULL

--- Menu ---
1. Display List 1
2. Sort List 1
3. Reverse List 1
4. Concatenate List 1 and List 2
5. Exit
Enter choice: 3
List reversed
8 -> 3 -> 1 -> NULL

--- Menu ---
1. Display List 1
2. Sort List 1
3. Reverse List 1
4. Concatenate List 1 and List 2
5. Exit
Enter choice: 4
Lists concatenated
8 -> 3 -> 1 -> 2 -> 4 -> NULL
```

Program 6b:

```
C:\Users\admin\Desktop\1BM X + v

--- Menu ---
1. Push (Stack)
2. Pop (Stack)
3. Display Stack
4. Enqueue (Queue)
5. Dequeue (Queue)
6. Display Queue
7. Exit
Enter choice: 1
Enter value: 5
Pushed 5 into stack

--- Menu ---
1. Push (Stack)
2. Pop (Stack)
3. Display Stack
4. Enqueue (Queue)
5. Dequeue (Queue)
6. Display Queue
7. Exit
Enter choice: 1
Enter value: 6
Pushed 6 into stack

--- Menu ---
1. Push (Stack)
2. Pop (Stack)
3. Display Stack
4. Enqueue (Queue)
5. Dequeue (Queue)
6. Display Queue
7. Exit
Enter choice: 3
Stack: 6 -> 5 -> NULL

--- Menu ---
1. Push (Stack)
2. Pop (Stack)
3. Display Stack
4. Enqueue (Queue)
5. Dequeue (Queue)
6. Display Queue
7. Exit
Enter choice: 2
Popped element: 6
```



```
--- Menu ---
1. Push (Stack)
2. Pop (Stack)
3. Display Stack
4. Enqueue (Queue)
5. Dequeue (Queue)
6. Display Queue
7. Exit
Enter choice: 4
Enter value: 1
Enqueued 1 into queue
```

```
--- Menu ---
1. Push (Stack)
2. Pop (Stack)
3. Display Stack
4. Enqueue (Queue)
5. Dequeue (Queue)
6. Display Queue
7. Exit
Enter choice: 4
Enter value: 2
Enqueued 2 into queue
```

```
--- Menu ---
1. Push (Stack)
2. Pop (Stack)
3. Display Stack
4. Enqueue (Queue)
5. Dequeue (Queue)
6. Display Queue
7. Exit
Enter choice: 6
Queue: 1 -> 2 -> NULL
```

```
--- Menu ---
1. Push (Stack)
2. Pop (Stack)
3. Display Stack
4. Enqueue (Queue)
5. Dequeue (Queue)
6. Display Queue
7. Exit
Enter choice: 5
Dequeued element: 1
```

Program 7:

```
C:\Users\admin\Desktop\1BM X + v

1.Insert at beginning
2.Insert at end
3.Delete node
4.Display forward
5.Display backward
Enter your choice:1
enter the value to be inserted:5

1.Insert at beginning
2.Insert at end
3.Delete node
4.Display forward
5.Display backward
Enter your choice:2
enter the value to be inserted:6

1.Insert at beginning
2.Insert at end
3.Delete node
4.Display forward
5.Display backward
Enter your choice:4
Forward: 5 6

1.Insert at beginning
2.Insert at end
3.Delete node
4.Display forward
5.Display backward
Enter your choice:5
Backward: 6 5
```

Program 8:

```
C:\Users\admin\Desktop\1BM X + v

---Binary Search Tree Menu---

1. Insert Element
2. Traversal (Pre-order, In-order, Post-order)
3. Display
4. Delete element
5. Exit
Enter the choice:1
Enter the value5
---Binary Search Tree Menu---
```

1. Insert Element
2. Traversal (Pre-order, In-order, Post-order)
3. Display
4. Delete element
5. Exit

Enter the choice:1

Enter the value6

---Binary Search Tree Menu---

1. Insert Element
2. Traversal (Pre-order, In-order, Post-order)
3. Display
4. Delete element
5. Exit

Enter the choice:1

Enter the value4

---Binary Search Tree Menu---

1. Insert Element
2. Traversal (Pre-order, In-order, Post-order)
3. Display
4. Delete element
5. Exit

Enter the choice:1

Enter the value7

---Binary Search Tree Menu---

1. Insert Element
2. Traversal (Pre-order, In-order, Post-order)
3. Display
4. Delete element
5. Exit

Enter the choice:1

Enter the value3

---Binary Search Tree Menu---

1. Insert Element
2. Traversal (Pre-order, In-order, Post-order)
3. Display
4. Delete element
5. Exit

Enter the choice:2

Pre-order Traversal:

54367In-order Traversal:

34567Post-order Traversal:

34765---Binary Search Tree Menu---

```

1. Insert Element
2. Traversal (Pre-order, In-order, Post-order)
3. Display
4. Delete element
5. Exit
Enter the choice:3
Tree elements (In-order traversal);34567
---Binary Search Tree Menu---

1. Insert Element
2. Traversal (Pre-order, In-order, Post-order)
3. Display
4. Delete element
5. Exit
Enter the choice:4
Enter the key:7
Node deleted if present
---Binary Search Tree Menu---

1. Insert Element
2. Traversal (Pre-order, In-order, Post-order)
3. Display
4. Delete element
5. Exit
Enter the choice:3
Tree elements (In-order traversal);3456
---Binary Search Tree Menu---

1. Insert Element
2. Traversal (Pre-order, In-order, Post-order)
3. Display
4. Delete element
5. Exit
Enter the choice:Tree elements (In-order traversal);3456
---Binary Search Tree Menu---

```

Program 9a:

```
C:\Users\amrut\OneDrive\De: X + v
Enter number of vertices: 5
Enter adjacency matrix:
0 1 1 0 0
1 0 1 0 0
1 1 0 1 1
0 0 1 0 0
0 0 1 0 0
Enter starting vertex (0 to 4): 0
BFS Traversal: 0 1 2 3 4
Process returned 0 (0x0)    execution time : 69.461 s
Press any key to continue.
|
```

Program 9b:

```
C:\Users\amrut\OneDrive\De: X + v
Enter number of vertices: 5
Enter adjacency matrix:
0 1 1 0 0
1 0 1 0 0
1 1 0 1 1
0 0 1 0 0
0 0 1 0 0
The graph is CONNECTED

Process returned 0 (0x0)    execution time : 42.799 s
Press any key to continue.
|
```

Program 10:

C:\Users\amrut\OneDrive\De: X

+

✓

```
Enter size of hash table (m): 5
Enter number of employee records (N): 5
Enter 4-digit key: 1234
Enter 4-digit key: 2345
Enter 4-digit key: 3456
Enter 4-digit key: 4567
Enter 4-digit key: 5678
```

Hash Table Contents:

Address	Key
0	2345
1	3456
2	4567
3	5678
4	1234

```
Process returned 0 (0x0)    execution time : 21.969 s
Press any key to continue.
```

|