Competitive Coding

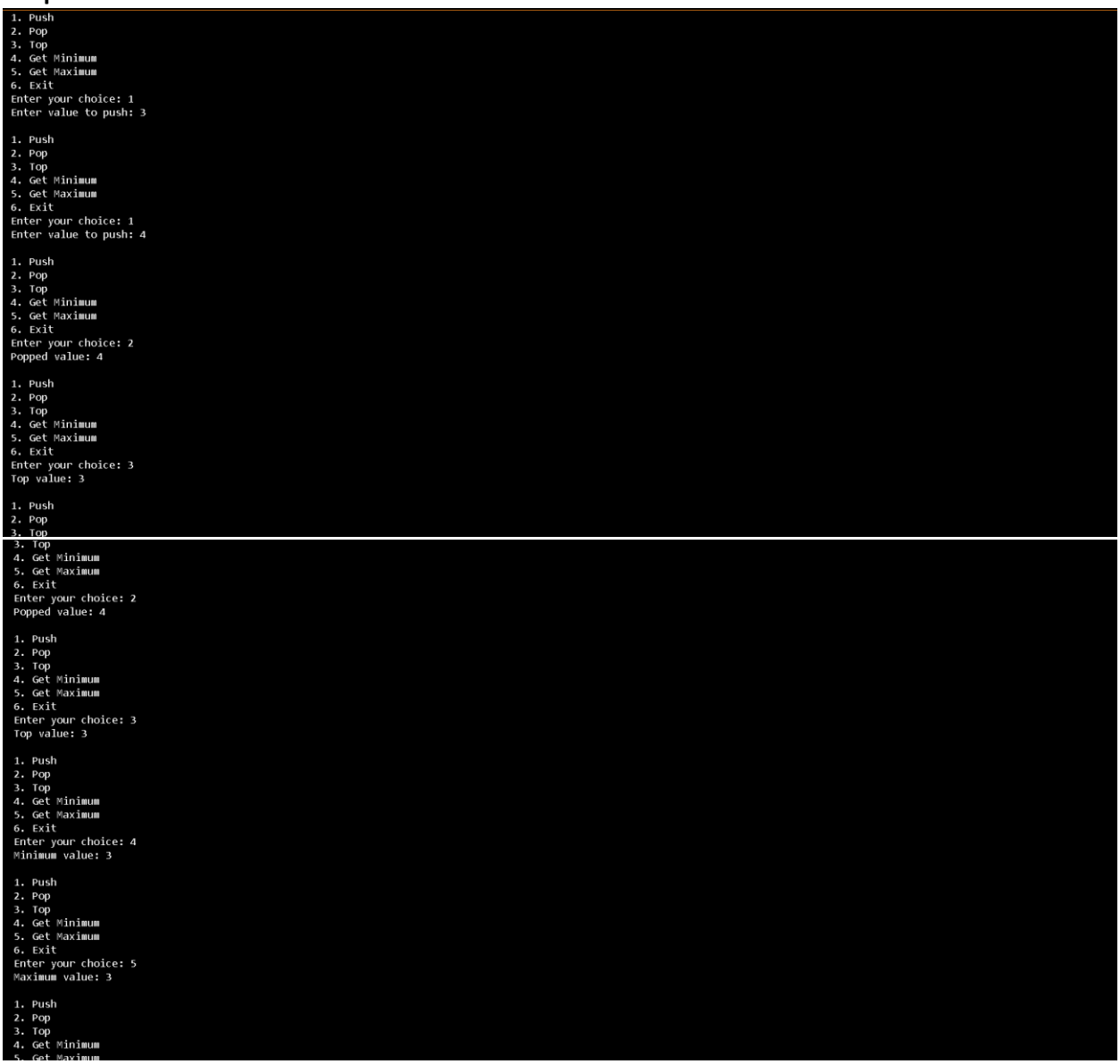
Level 0 output:

```
1. Insert at head
2. Insert at tail
3. Traverse forward
4. Traverse backward
5. Exit
Enter your choice: 1
Enter data to insert at head: 10

1. Insert at head
2. Insert at tail
3. Traverse forward
4. Traverse backward
5. Exit
Enter your choice: 1
Enter data to insert at head: 40

1. Insert at head
2. Insert at tail
3. Traverse forward
4. Traverse backward
5. Exit
Enter your choice: 1
Enter data to insert at head: 60

1. Insert at head
2. Insert at tail
3. Traverse forward
4. Traverse backward
5. Exit
Enter your choice: 2
Enter data to insert at tail: 1

1. Insert at head
2. Insert at tail
3. Traverse forward
4. Traverse backward
```

```
Enter your choice: 2
Enter data to insert at tail: 1

1. Insert at head
2. Insert at tail
3. Traverse forward
4. Traverse backward
5. Exit
Enter your choice: 2
Enter data to insert at tail: 5

1. Insert at head
2. Insert at tail
3. Traverse forward
4. Traverse backward
5. Exit
Enter your choice: 2
Enter data to insert at tail: 8

1. Insert at head
2. Insert at tail
3. Traverse forward
4. Traverse backward
5. Exit
Enter your choice: 3
Forward Traversal:
60 -> 40 -> 10 -> 1 -> 5 -> 8 -> NULL

1. Insert at head
2. Insert at tail
3. Traverse forward
4. Traverse backward
5. Exit
Enter your choice: 4
Backward Traversal:
8 -> 5 -> 1 -> 10 -> 40 -> 60 -> NULL

1. Insert at head
2. Insert at tail
```

Level 1:

Output:



Level 2:

Output:

```
1. Add interval
2. Get intervals
3. Exit
Enter your choice: 1
Enter start of interval: 1 5
Enter end of interval:
1. Add interval
2. Get intervals
3. Exit
Enter your choice: 1
Enter start of interval: 6 8
Enter end of interval:
1. Add interval
2. Get intervals
3. Exit
Enter your choice: 1
Enter start of interval: 4 7
Enter end of interval:
1. Add interval
2. Get intervals
3. Exit
Enter your choice: 2
Current intervals: [[1, 8]]

1. Add interval
2. Get intervals
3. Exit
Enter your choice:
```

Level 3:

Output:

```
1. Set key-value pair
2. Get value by key
3. Print cache entries
4. Exit
Enter your choice: 1
Enter key: my_key
Enter value: Hello,Avishek!
Enter expiry time in seconds: 60

1. Set key-value pair
2. Get value by key
3. Print cache entries
4. Exit
Enter your choice: 2
Enter key: my_key
Value for key 'my_key': Hello,Avishek!

1. Set key-value pair
2. Get value by key
3. Print cache entries
4. Exit
Enter your choice: 3
Index 0:
Index 1:
Index 2:
Index 3:
Index 4:
Key: my_key, Value: Hello,Avishek!, Expiry Time: 1741368909
Index 5:
Index 6:
Index 7:
Index 8:
Index 9:

1. Set key-value pair
2. Get value by key
3. Print cache entries
4. Exit
Enter your choice:
```