BSDS – Assignment 1

Problem-1 (10 Marks)

- Implement a BST in your preferred language (C++/C).
- Augment the BST to support the following operations:
    1. Insert, Delete, Search
    2. Find floor and ceiling of a given key
    3. Range query: Print all keys between [L, R]
- Input Format: A sequence of operations (I for insert, D for delete, S for search, Q for range query, F for floor, C for ceiling).
- Output Format: Appropriate results after each query.

Test Case 1

Input:

```
I 40
I 20
I 60
I 10
D 20
S 20
F 25
C 25
Q 5 50
```

Output:

```
Inserted 40
Inserted 20
Inserted 60
Inserted 10
Deleted 20
Not Found
Floor of 25 = 10
Ceiling of 25 = 40
Keys in range [5,50] = 10 40
```

Test Case 2

Input:

```
I 100
I 50
I 200
I 150
I 250
D 100
S 150
F 180
C 180
Q 120 260
```

Output:

```
Inserted 100
Inserted 50
Inserted 200
Inserted 150
Inserted 250
Deleted 100
Found
Floor of 180 = 150
Ceiling of 180 = 200
Keys in range [120,260] = 150 200 250
```

Test Case 3

Input:

```
I 30
I 20
I 40
I 35
D 30
S 30
F 38
C 38
Q 10 45
```

Output:

```
Inserted 30
Inserted 20
Inserted 40
Inserted 35
Deleted 30
Not Found
Floor of 38 = 35
Ceiling of 38 = 40
Keys in range [10,45] = 20 35 40
```

Problem-2 (10 Marks):

Given the heights of bars in a histogram, use a stack to compute the **largest rectangular area** possible.

Test Case 1 – Increasing heights

Input:

```
[1, 2, 3, 4, 5]
```

Output:

```
9   (bars [3,4,5] → 3*3 = 9)
```

Test Case 2 – Decreasing heights

Input:

```
[5, 4, 3, 2, 1]
```

Output:

```
9   (bars [5,4,3] → 3*3 = 9)
```

Test Case 3 – Mixed heights

Input:

```
[6, 2, 5, 4, 5, 1, 6]
```

Output:

```
12   (bars [5,4,5] → 3*4 = 12)
```
