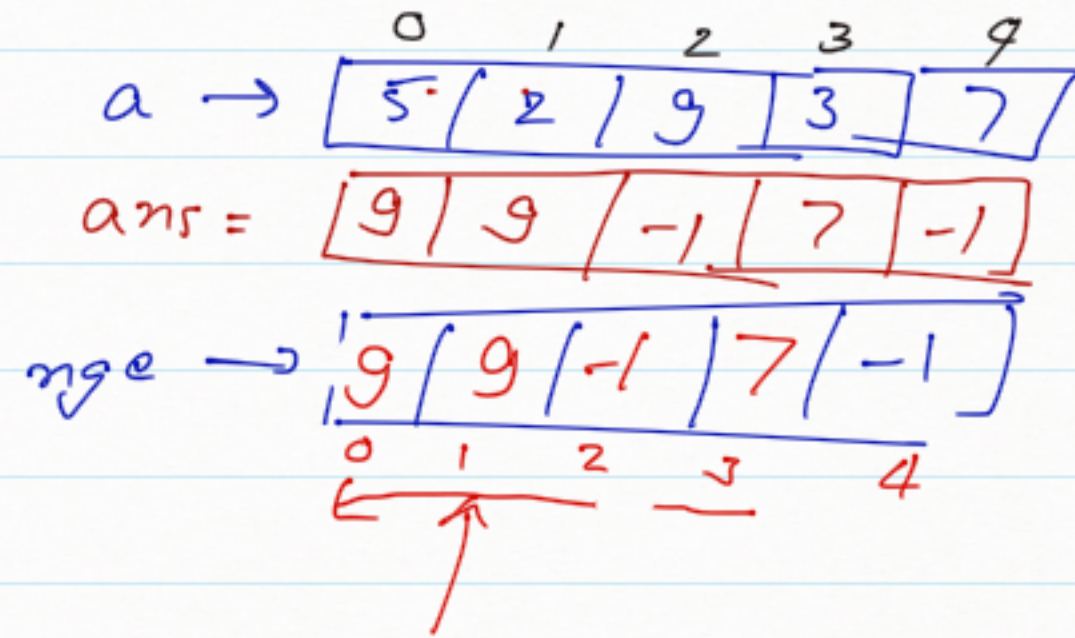
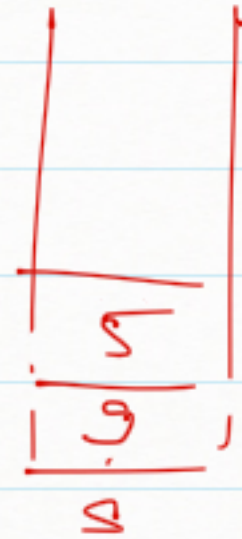


1) Stack.

2) for next greater on the right → start from right.  
" " " left → start from left.



Next Greater element on the right.



```

✓ s.push(a[a.length-1]);
nge[a.length-1] = -1;
for(i = a.length-2; i >= 0; i--)
{
    while(!s.isEmpty() && s.top() < a[i])
    {
        s.pop();
    }
    if(s.isEmpty()) { nge[i] = -1; }
    else { nge[i] = s.top(); } s.push(a[i]);
}
    
```



```
int nge[] = new int[a.length];
Stack<Integer> s = new Stack<>();
```

```
s.push(a[0]);
```

```
nge[0] = -1;
```

a = 

5	8	2	1	4
---	---	---	---	---

nge 

-1	-1	8	2	8
----	----	---	---	---

```
foo(int i=1; i <= 4; i++)
```

```
{ while(!s.isEmpty() && a[i] > s.top())
```

```
{ s.pop();
```

```
}
```

```
if (s.isEmpty())
```

```
{ nge[i] = -1;
```

```
}
```

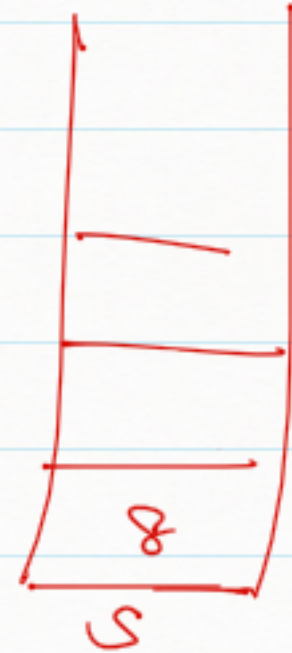
```
else
```

```
{ nge[i] = s.top(); ✓
```

```
}
```

```
s.push(a[i]);
```

```
}
```





## Stock Span

```
int a[] = {5, 3, 4, 8, 4, 6, 7};
```

```
int[] span = new int[a.length];
Stack<Integer> s = new Stack<>();
s.push(0); ✓
```

```
span[0] = 1;
```

```
for(int i=1; i<=6; i++)
```

```
{
    while (!s.isEmpty() && a[i] > a[s.top()])
```

```
{
    s.pop();
```

```
}
```

```
if (s.isEmpty())
```

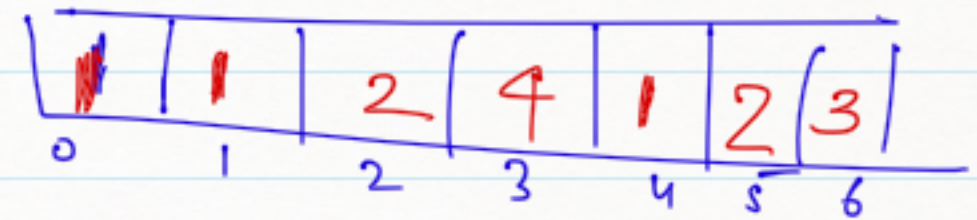
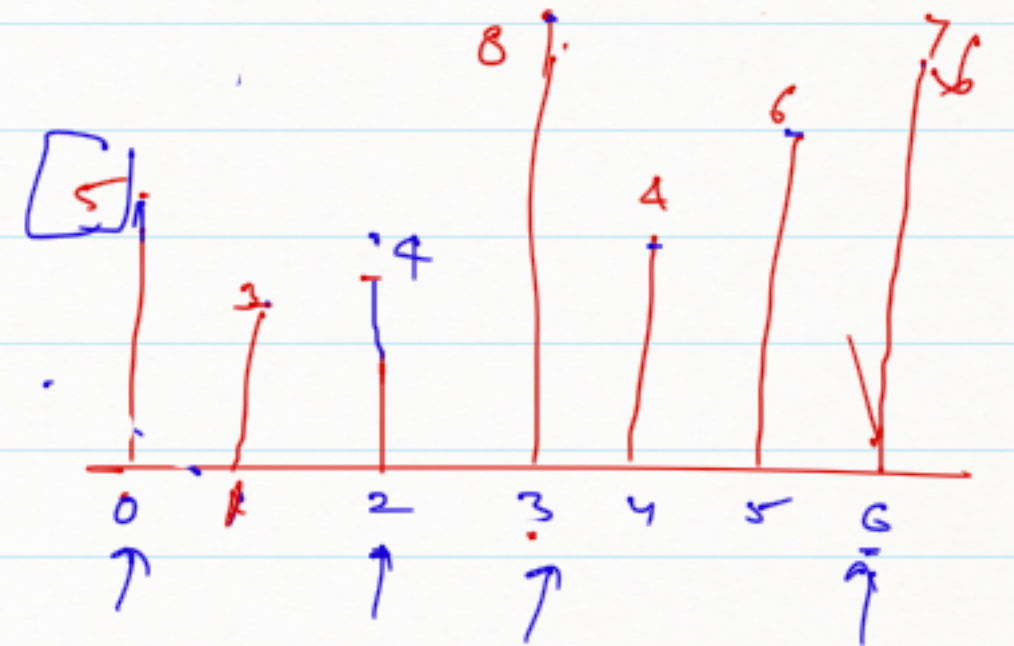
```
span[i] = i+1; ✓
```

```
else
```

```
{
    span[i] = i - s.top(); ✓✓
```

```
}
    s.push(i);
```

```
}
```



```
for(i=0; i<span.length; i++) span
```

```
{
    print span[i];
}
```

ans = [1, 1, 2, 4, 1, 2, 3]



Given a histogram, find the area of largest rectangle

```
int a[] = {2, 6, 5, 4, 5, 1};
int rb[] = new int[a.length];
int lb[] = new int[a.length];
```

```
Stack<Integer> s = new Stack<>();
```

```
s.push(0);
```

```
lb[0] = -1
```

```
for (int i = 1; i <= 5; i++)
```

```
{ while (!s.isEmpty() && a[i] < a[s.top()])
```

```
{ s.pop();
```

```
}
```

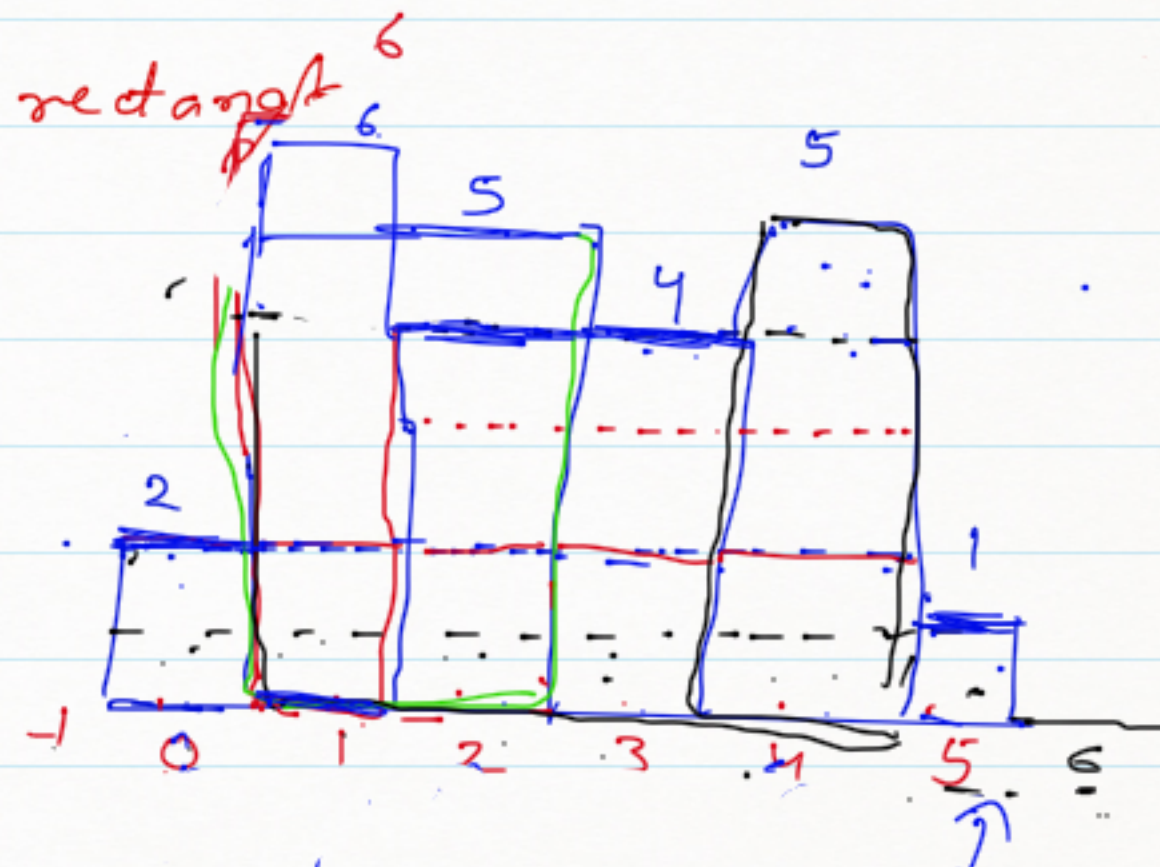
```
if (s.isEmpty())
```

```
{ lb[i] = -1;
```

```
else
```

```
{ lb[i] = s.top();
```

```
s.push(i);
```



rb	5	2	3	5	5	6
lb	-1	0	0	0	3	-1

lb	-1	0	0	0	3	-1
	0	1	2	3	4	5