

**Day-31**

**Python DSA**

## **Square Root**

<https://www.geeksforgeeks.org/problems/square-root/0>

class Solution:

```
def floorSqrt(self, n):
```

```
    if n==0 or n==1:
```

```
        return n
```

```
    low, high=1,n
```

```
    ans=0
```

```
    while low<=high:
```

```
        mid = (low + high)//2
```

```
        if mid* mid==n:
```

```
            return mid
```

```
        elif mid*mid<n:
```

```
            ans=mid
```

```
            low=mid +1
```

```
        else:
```

```
            high = mid-1
```

```
    return ans
```

TC-  $O(\log N)$

SC-  $O(1)$

## Find nth root of m

<https://www.geeksforgeeks.org/problems/find-nth-root-of-m5843/1>

class Solution:

```
    def nthRoot(self, n, m):  
        if m==0:  
            return 0  
  
        if m==1:  
            return 1  
  
        low, high = 1, m  
  
        while low <= high:  
            mid = (low+high)//2  
  
            power = mid**n  
  
            if power == m:  
                return mid  
  
            elif power < m:  
                low = mid+1  
  
            else:  
                high = mid-1  
  
        return -1
```

TC-  $O(\log N)$

SC-  $O(1)$

