**Root to leaf path sum:-**

Given a binary tree and an integer S, check whether there is root to leaf path with its sum as S.

**Example 1:**

**Input:**

Tree =

1

/ \

2 3

S = 2

**Output:** 0

**Explanation:**

There is no root to leaf path with sum 2.

**Example 2:**

**Input:**

Tree =

1

/ \

2 3

S = 4

**Output:** 1

**Explanation:**

The sum of path from leaf node 3 to root 1 is 4.

**Your Task:**  
You dont need to read input or print anything. Complete the function**hasPathSum()** which takes root node and target sum S as input parameter and returns true if path exists otherwise it returns false.

**Expected Time Complexity:**O(N)  
**Expected Auxiliary Space:** O(height of tree)

**Constraints:**  
1 ≤ N ≤ 10^4  
1 ≤ S ≤ 10^6