Paths from root with a specified sum

Given a Binary tree and a sum **S**, print all the paths, **starting from root**, that sums upto the given sum. Path **may not** end on a leaf node.

Example 1:

```
Input :
sum = 8
Input tree
          1
        /   \
      20      3
            /    \
          4        15
         /  \      /  \
        6    7    8    9

Output :
1 3 4
Explanation :
Sum of path 1, 3, 4 = 8.
```

Example 2:

```
Input :
sum = 38
Input tree
           10
         /     \
      28         13
               /     \
           14         15
         /    \      /  \
        21    22    23   24
```

```
Output :
10 28
10 13 15
Explanation :
Sum of path 10, 28 = 38 and
Sum of path 10, 13, 15 = 38.
```

Your task :

You don't have to read input or print anything. Your task is to complete the function **printPaths()** that takes the root of the tree and sum as input and returns a vector of vectors containing the paths that lead to the sum.

Expected Time Complexity: $O(N^2)$

Expected Space Complexity: $O(N)$

Your Task :

$1 <= N <= 2*10^3$

$-10^3 <= sum, Node.key <= 10^3$