Solution

Approach 1: Set [Accepted]

Intuition:

Find all the possible integers that can be generated from the binary string S which are less than N and put it in set. Then check if the set contains all the integers from 1 to N, If is contains all the integers from 1 to N then answer is “true” else answer is “false”.

Algorithm:

Iterate through the string S, and if the character at index i is ‘1’ then generate all the integers whose binary representation is same as that of substring starting from this index.

Code:



Complexity Analysis:

Time Complexity:

Worst Case will occur when all the characters in the binary substring is ‘1’. So worst case time complexity will be O(noOfBits(N) \* length(S)). The biggest value of N is 10^9 and number of bits required to represent it is 30, so value of noOfBits(N) doesn’t exceed 30. Moreover noOfBits(N) = log2(N) + 1, to represent N in it’s binary representation. Hence time complexity of above algorithm is O( M \* log(N) ), where M = len|S|.

Space Complexity:

As we are storing numbers from 1 to N inside Set, space required would be O(N) in worst case.