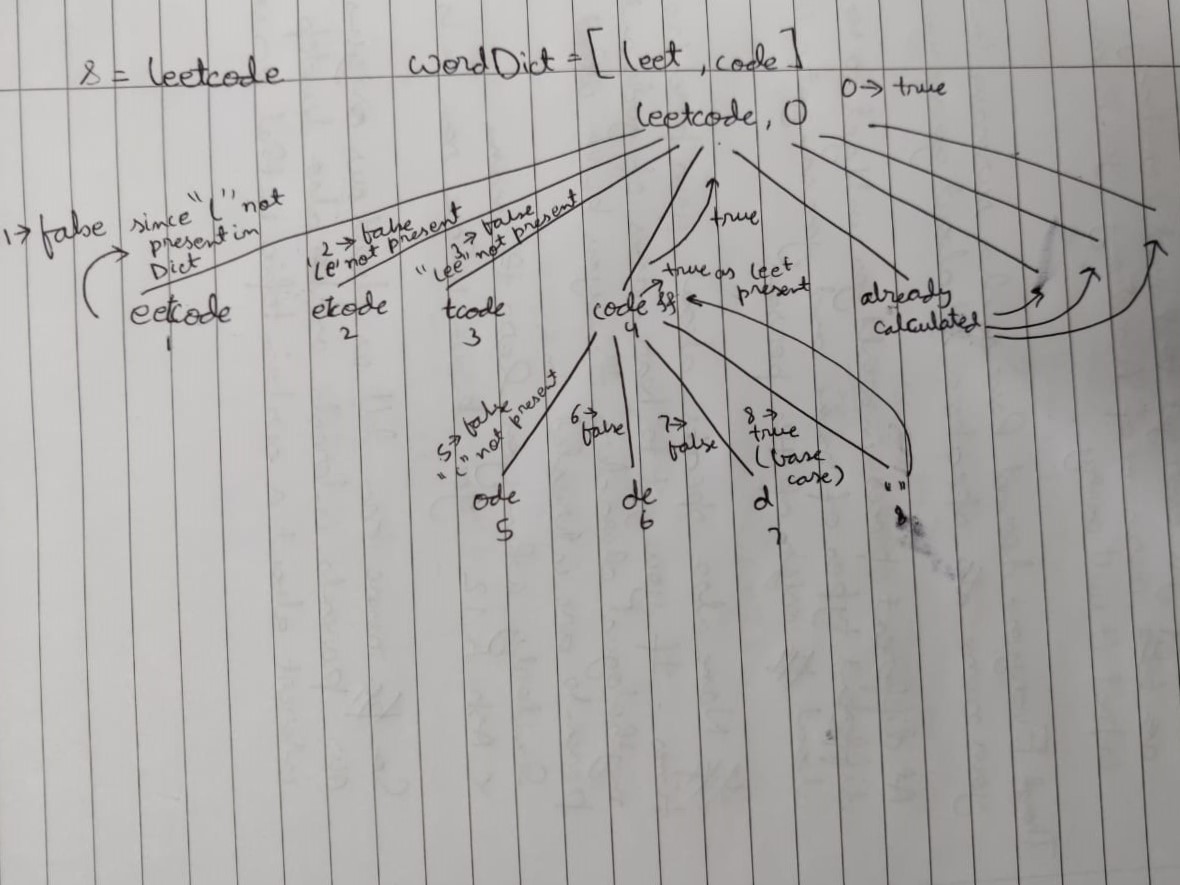
Question: https://leetcode.com/problems/word-break/

Create a helper function which will check if substring starting from ith position can be formed by wordDict elements.

To optimize store ith pos as key and helper value as value.



Code:  
class Solution {

public boolean wordBreak(String s, List<String> wordDict) {

HashMap<Integer, Boolean> mem = new HashMap<>();

mem.put(0,wordBreakChecker(s, wordDict, 0, mem));

return mem.get(0);

}

public boolean wordBreakChecker(String s, List<String> wordDict, int start, HashMap<Integer, Boolean> mem){

//for last position by default set true

if(s.length()==start){

mem.put(start, true);

return true;

}

//check if the sub problem has already been solved

if(mem.containsKey(start)){

return mem.get(start);

}

for(int i=start; i<s.length(); i++){

// if the first sub string is present in wordDict and the left over string can be formed by wordDict elements

if(wordDict.contains(s.substring(start, i+1)) && wordBreakChecker(s, wordDict, i+1, mem)){

mem.put(start, true);

return true;

}

}

mem.put(start, false);

return false;

}

}

Github Link :<https://lnkd.in/ecwtJeaz>