Word Break II

Try to solve the Word Break II problem.

We'll cover the following

- Statement
- Examples
- Understand the problem
- Figure it out!
- Try it yourself

Statement

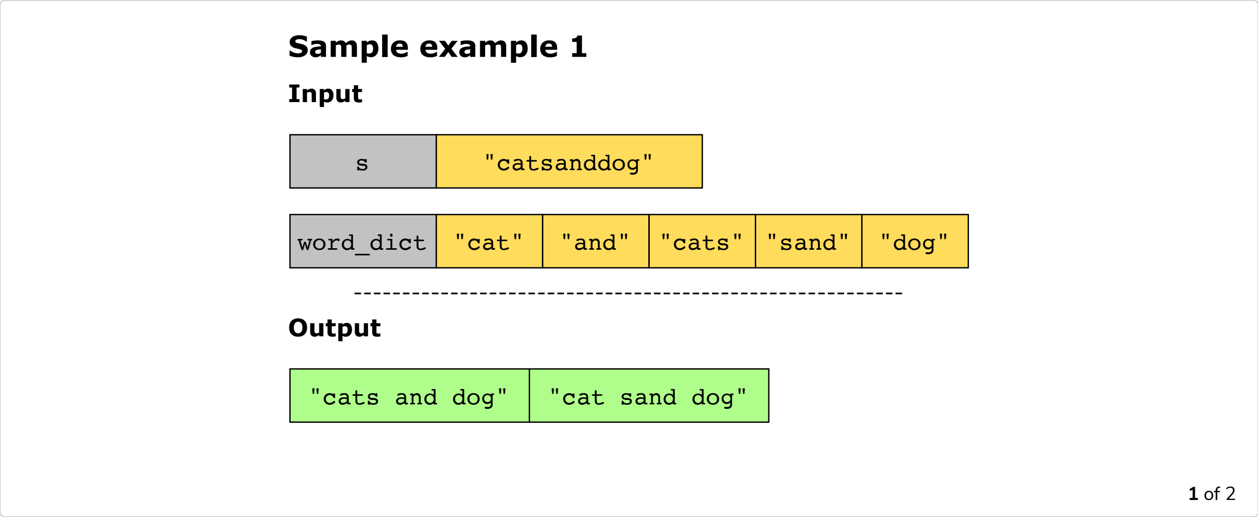
You are given a string, `s`, and an array of strings, `wordDict`, representing a dictionary. Your task is to add spaces to `s` to break it up into a sequence of valid words from `wordDict`. We are required to return an array of all possible sequences of words (sentences). The order in which the sentences are listed is not significant.

> **Note:** The same dictionary word may be reused multiple times in the segmentation.

Constraints:

- $1 \leq$ `s.length` ≤ 20
- $1 \leq$ `wordDict.length` ≤ 1000
- $1 \leq$ `wordDict[i].length` ≤ 10
- `s` and `wordDict[i]` consist of only lowercase English letters.
- All the strings of `wordDict` are unique.

Examples



Sample example 1

Input

s	"catsanddog"

word_dict	"cat"	"and"	"cats"	"sand"	"dog"

Output

"cats and dog"	"cat sand dog"

1 of 2

Understand the problem

Let’s take a moment to make sure you’ve correctly understood the problem. The quiz below helps you check if you’re solving the correct problem:

Word Break II

1 What is the output if the following input string and dictionary are provided as input?

s = “pineapplepenapple”

word_dict = [“apple”, “pen”, “applepen”, “pine”, “pineapple”]

A) [“pine apple pen apple”, “pineapple pen apple”, “pine applepen apple”]

B) [“pine apple pen apple”, “pine applepen apple”]

C) [“pineapple pen apple”]

D) []

Submit Answer

< Question 1 of 3 >
0 attempted

Reset Quiz

Figure it out!

We have a game for you to play. Rearrange the logical building blocks to develop a clearer understanding of how to solve this problem.

Drag and drop the cards to rearrange them in the correct sequence.

Create a 2D table where each entry corresponds to a prefix of the input string and store an array of all possible sentences that can be formed using that substring.

Iterate over all prefixes of the input string.

For each prefix, iterate over its suffixes and check whether the suffix is a valid word in the dictionary.

If the suffix is a valid word, combine it with all possible sentences that can be formed using the prefix to the left of it.

Store the array of all possible sentences that can be formed using the current prefix in the corresponding entry of the table.

After processing all prefixes of the input string, return the list of all possible sentences that can be formed using the complete string.

Reset Show Solution Submit

Try it yourself

Implement your solution in the following coding playground:

usercode > main.java

```java
import java.util.*;
class Main {
  public static List<String> wordBreak(String s, List<String> WordDict) {

    // Replace this placeholder return statement with your code
    return new ArrayList<String>();
  }
}
```

Powered by AI Submit

Test Cases Results

Case 1 Case 2 Case 3

Input #1

"magiclly"

Input #2

```
["ag","al","icl","mag","magic","ly","lly"]
```

Word Break II

← Back

Solution: Partition Equ...

Next →

Solution: Word Break II

☑ Mark as Completed