Problem 271: Encode and Decode Strings

Problem Information

Difficulty: Medium
Acceptance Rate: 50.66%
Paid Only: Yes
Tags: Array, String, Design

Problem Description

Design an algorithm to encode **a list of strings** to **a string**. The encoded string is then sent over the network and is decoded back to the original list of strings.

Machine 1 (sender) has the function:

string encode(vector<string> strs) { // ... your code return encoded_string; }

Machine 2 (receiver) has the function:

vector<string> decode(string s) { //... your code return strs; }

So Machine 1 does:

string encoded_string = encode(strs);

and Machine 2 does:

vector<string> strs2 = decode(encoded_string);

`strs2` in Machine 2 should be the same as `strs` in Machine 1.

Implement the `encode` and `decode` methods.

You are not allowed to solve the problem using any serialize methods (such as `eval`).

Example 1:

Input: dummy_input = ["Hello","World"] **Output:** ["Hello","World"] **Explanation:**
Machine 1: Codec encoder = new Codec(); String msg = encoder.encode(strs); Machine 1
---msg---> Machine 2 Machine 2: Codec decoder = new Codec(); String[] strs =
decoder.decode(msg);

Example 2:

Input: dummy_input = [""] **Output:** [""]

Constraints:

* `1 <= strs.length <= 200` * `0 <= strs[i].length <= 200` * `strs[i]` contains any possible
characters out of `256` valid ASCII characters.

Follow up: Could you write a generalized algorithm to work on any possible set of
characters?

Code Snippets

C++:

```
class Codec {
public:

// Encodes a list of strings to a single string.
string encode(vector<string>& strs) {


}


// Decodes a single string to a list of strings.
vector<string> decode(string s) {


}
};


// Your Codec object will be instantiated and called as such:
// Codec codec;
// codec.decode(codec.encode(strs));
```

Java:

```java
public class Codec {

// Encodes a list of strings to a single string.
public String encode(List<String> strs) {

}

// Decodes a single string to a list of strings.
public List<String> decode(String s) {

}
}

// Your Codec object will be instantiated and called as such:
// Codec codec = new Codec();
// codec.decode(codec.encode(strs));
```

Python3:

```python
class Codec:
def encode(self, strs: List[str]) -> str:
"""Encodes a list of strings to a single string.
"""


def decode(self, s: str) -> List[str]:
"""Decodes a single string to a list of strings.
"""



# Your Codec object will be instantiated and called as such:
# codec = Codec()
# codec.decode(codec.encode(strs))
```