

Encode and Decode Strings

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.

Note:

- The string may contain any possible characters out of 256 valid ascii characters. Your algorithm should be generalized enough to work on any possible characters.
- Do not use class member/global/static variables to store states. Your encode and decode algorithms should be stateless.
- Do not rely on any library method such as `eval` or serialize methods. You should implement your own encode/decode algorithm.

Solution 1

```
public class Codec {  
  
    // Encodes a list of strings to a single string.  
    public String encode(List<String> strs) {  
        StringBuilder sb = new StringBuilder();  
        for(String s : strs) {  
            sb.append(s.length()).append('/').append(s);  
        }  
        return sb.toString();  
    }  
  
    // Decodes a single string to a list of strings.  
    public List<String> decode(String s) {  
        List<String> ret = new ArrayList<String>();  
        int i = 0;  
        while(i < s.length()) {  
            int slash = s.indexOf('/', i);  
            int size = Integer.valueOf(s.substring(i, slash));  
            ret.add(s.substring(slash + 1, slash + size + 1));  
            i = slash + size + 1;  
        }  
        return ret;  
    }  
}
```

written by [qianzhige](#) original link [here](#)

Solution 2

The rule is, for each str in strs, encode it as + '@' + str

```
class Codec {
public:

    // Encodes a list of strings to a single string.
    string encode(vector<string>& strs) {
        string encoded = "";
        for (string &str: strs) {
            int len = str.size();
            encoded += to_string(len) + "@" + str;
        }

        return encoded;
    }

    // Decodes a single string to a list of strings.
    vector<string> decode(string s) {
        vector<string> r;
        int head = 0;
        while (head < s.size()) {
            int at_pos = s.find_first_of('@', head);
            int len = stoi(s.substr(head, at_pos - head));
            head = at_pos + 1;
            r.push_back(s.substr(head, len));
            head += len;
        }

        return r;
    }
};

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

written by [sculd](#) original link [here](#)

Solution 3

Is anyone else getting this error? I get it for any input that doesn't have syntax errors, even if I just try to compile the blank problem skeleton.

written by [omg_zozobra](#) original link [here](#)

From [LeetCoder](#).