🎉 Master algorithms together on Binary Search! Create a room, invite your friends, and race to finish the problems.   ✕

Daily Coding Problem

Blog

Daily Coding Problem #29

Problem

This problem was asked by Amazon.

Run-length encoding is a fast and simple method of encoding strings. The basic idea is to represent repeated successive characters as a single count and character. For example, the string "AAAABBBCCDAA" would be encoded as "4A3B2C1D2A".

Implement run-length encoding and decoding. You can assume the string to be encoded have no digits and consists solely of alphabetic characters. You can assume the string to be decoded is valid.

Solution

We can implement `encode` by iterating over our input string and keeping a current count of whatever the current

character is, and once we encounter a different one, appending the count (as a string) and the actual character to our result string.

```python
def encode(s):
    if not s:
        return ''

    result = ''
    current_char = s[0]
    current_count = 1
    for i, char in enumerate(s, 1):
        if char == current_char:
            current_count += 1
        else:
            result += str(current_count) + current_char
            current_char = char
            current_count = 1
    result += str(current_count) + current_char
    return result
```

We can implement decode by iterating over the encoded string and checking each character for a digit. If it is, then calculate the correct count, and once we find its corresponding character, extend the result with the character count number of times and then reset the count.

```python
def decode(s):
    count = 0
    result = ''
    for char in s:
        if char.isdigit():
            count = count * 10 + int(char)
```

```
        else:
            # char is alphabetic

            result += char * count

            count = 0
    return result
```