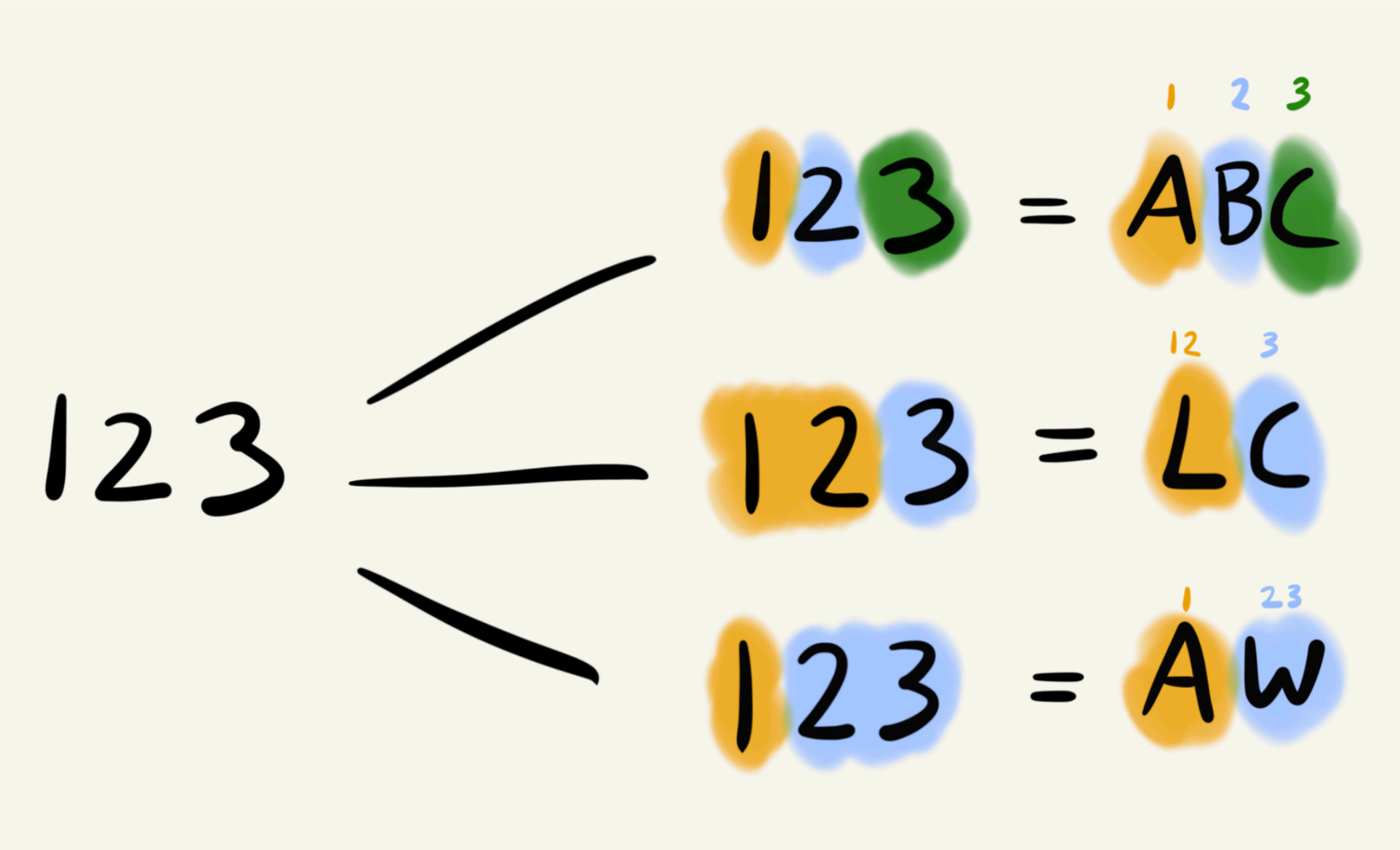
Decode Ways Problem

We are given a message which has been encoded into a string of digits, using the following mapping:

```
A  =  1
B  =  2
   ...
Z  =  26
```

Our goal is to find the number of possible ways we can decode the string of digits.



Analysis

If we have ‘123’, what are all of the possible options for a first letter?

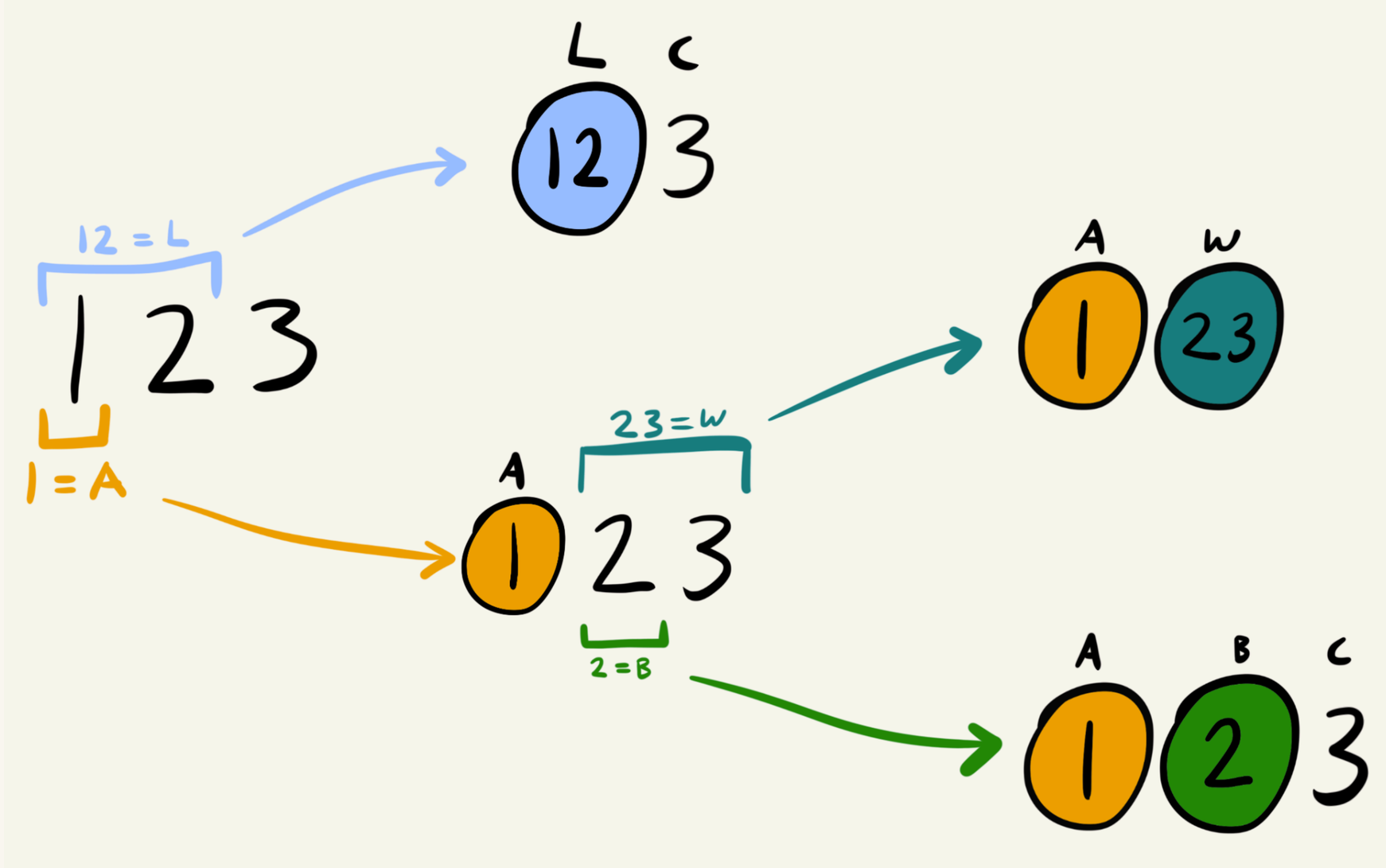
- **1 = A**
- **12 = L**

If we choose 1 (A), then we have the further two options:

- **2 = B**
- **23 = W**

If we choose 12 (L), then we only have one option:

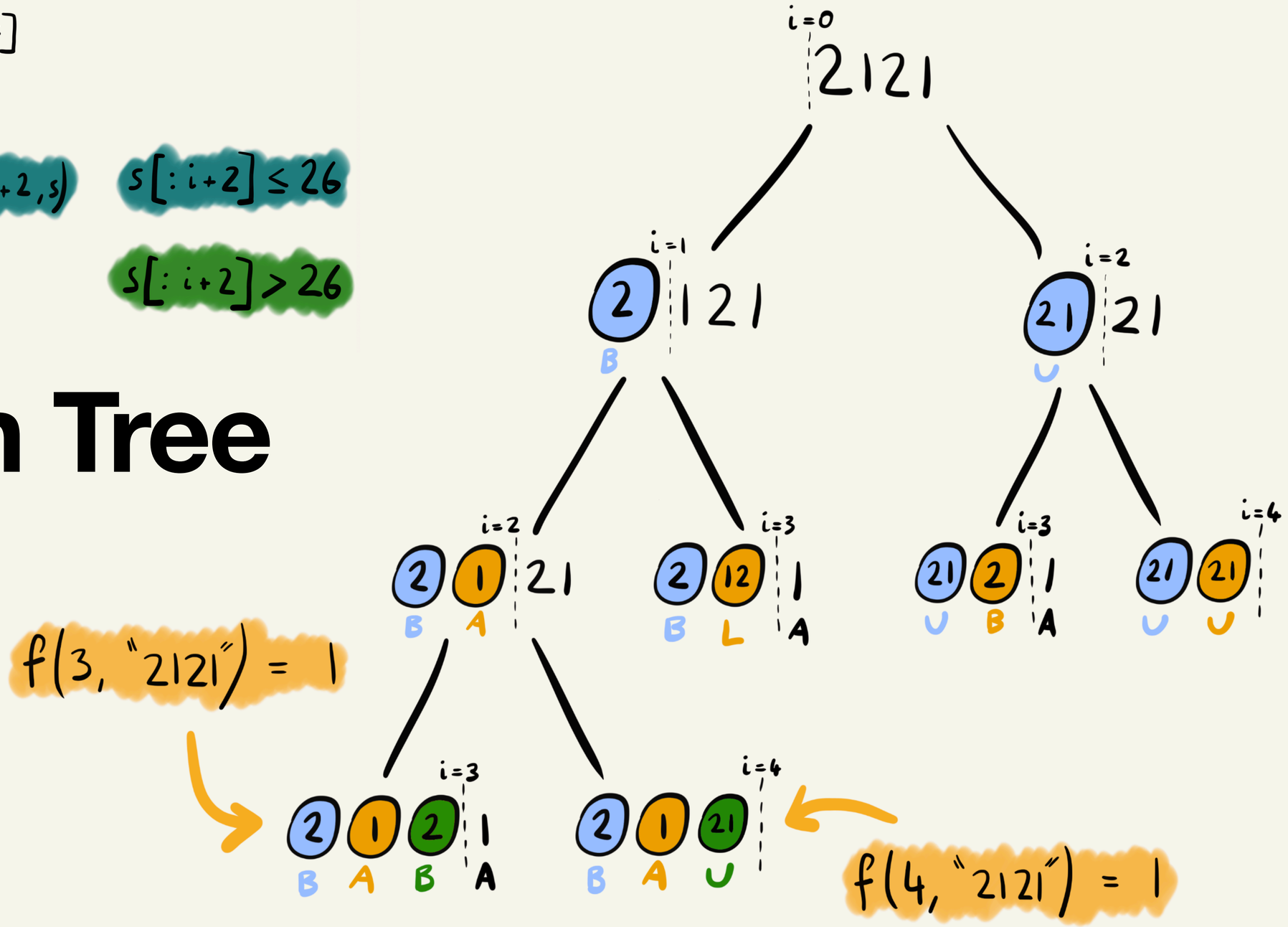
- **3 = C**



number of ways of decoding the string s[i:]

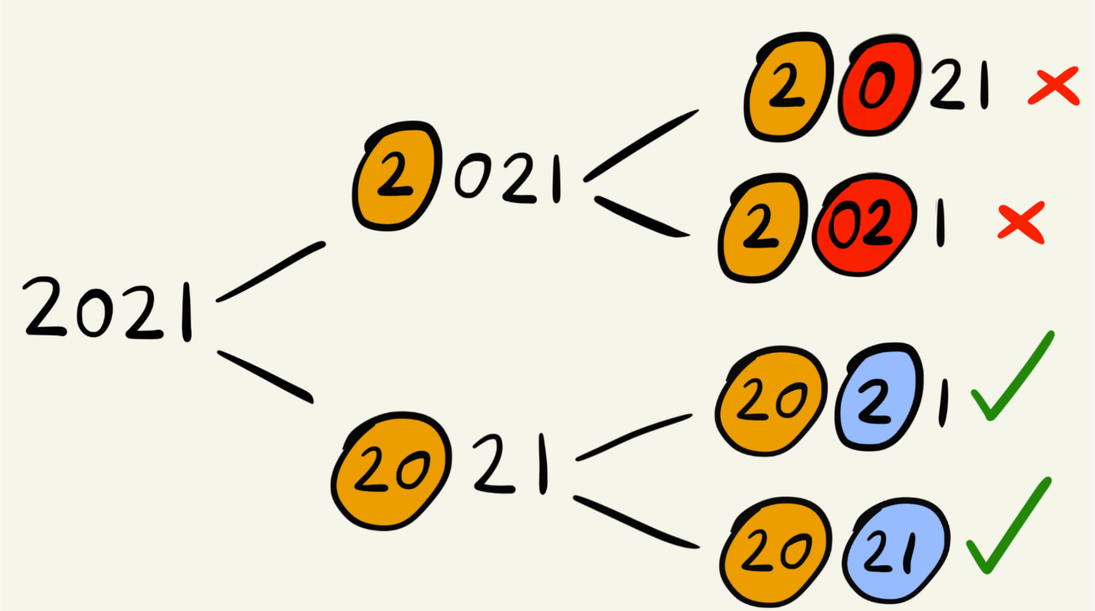
$$f(i,s) = \begin{cases} f(i+1,s) + f(i+2,s) & s[:i+2] \le 26 \\ f(i+1,s) & s[:i+2] > 26 \end{cases}$$

Recursion Tree

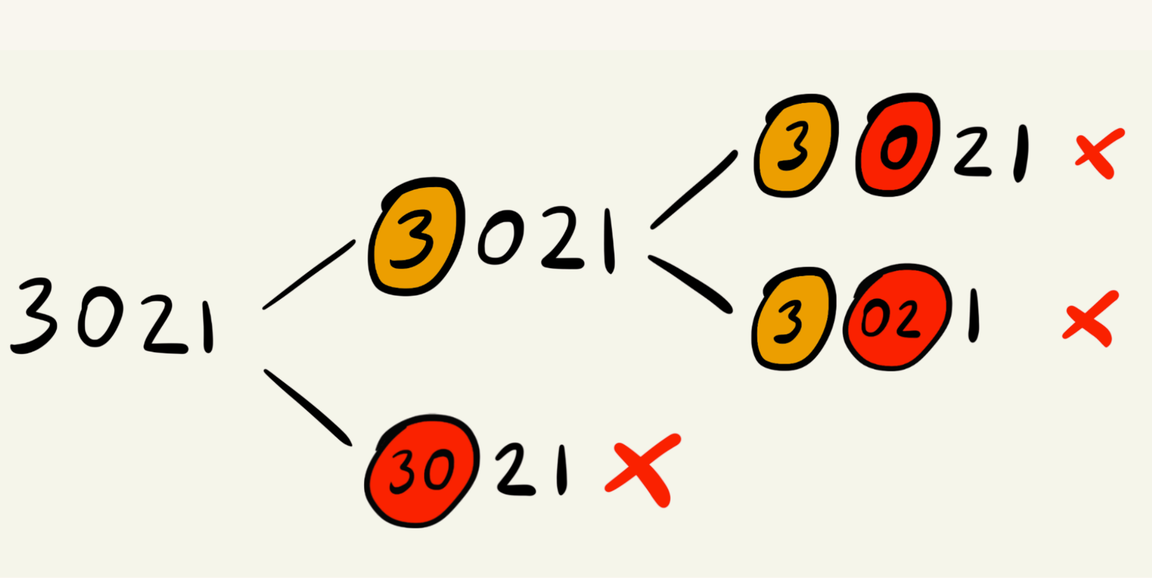


How to Deal with 0s

If a string starts with a 0, there are no ways to decode it, as there are no mappings that start with 0. Mappings start at 1 (A) and finish at 26 (Z).



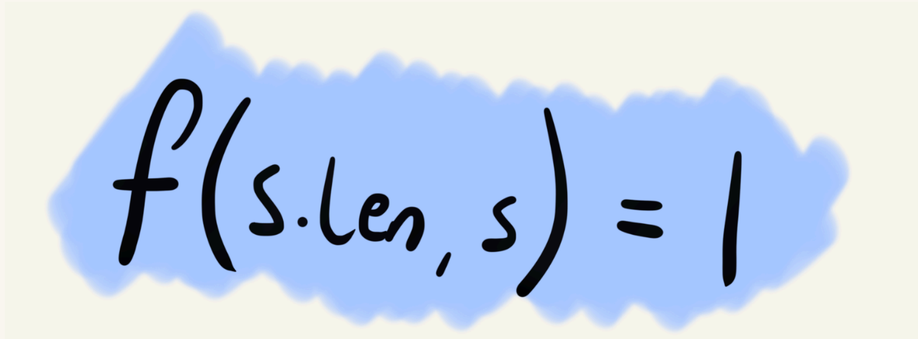
2 solutions



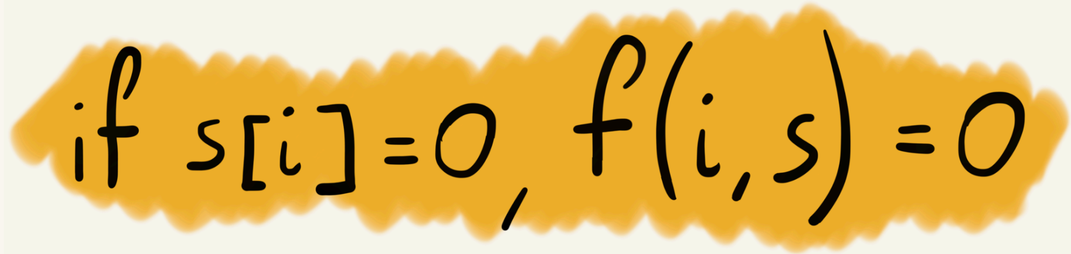
0 solutions

Base Cases

Base Case 1: If we are at the end of the string, we are essentially asking: how many ways are there to decode an empty string? There is one way of decoding “” — it is simply “”.



Base Case 2: If the value of the string at the current index is “0”, then there are going to be no ways of decoding this, as shown previously.



Base Case 3: If we are at the last index of the string, and the value is not “0”, then we know there will be one way of decoding the string as the value is definitely between 1 and 9.