Decode a string to a list of strings

```
class Solution:
    def decode(self, s: str) -> list[str]:
        if not s:
            return []
        sizes = []
        result = []
        i = 0
        while s[i] != '#':
            curr = ""
            while s[i] != ',':
                curr += s[i]
                i += 1
            sizes.append(int(curr))
            i += 1
        i += 1
        for sz in sizes:
            result.append(s[i: i + sz])
            i += sz
        return result
```

print (Solution(). decode("6,4,#georgecode"))

Step 1:

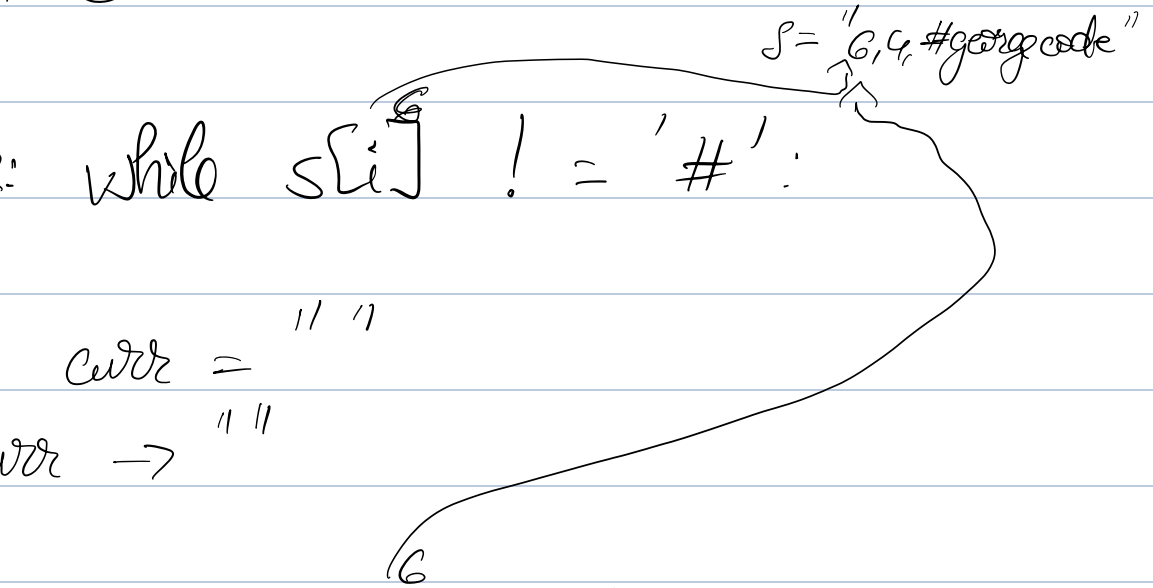
self → solution instance

s → "6,4,#georgecode"

sizes → [] - empty list

result → [] - empty list

i → 0



Step 2: while s[i] != '#':

Step 3: curr = ""

curr → ""

Step 4: while s[i] != ',':

curr += s[i]

i += 1

curr → "6"

i → 1 then s[1] == ",":

Step 5: sizes.append(int(curr)), i += 1

sizes → [6]

i -> 2

s - 6,4,#georgcode

Step 6: while s[i] != '#':

Step 7: curr = ""

curr -> ""

Step 8: while s[i] != ',':

curr += s[i]

i += 1

curr -> "4"

i -> 3 then s[3] == ",":

Step 9: sizes.append(int(curr)), i += 1

sizes -> [6,4]

i -> 4

Step 10: while s[i] != "#":

s[4] == "#"

Step 11: i += 1

i -> 5

Step 12:

```
for s2 in sizes:
    result.append(s[i:i+s2])
    i += s2
```

s = "6,4,#georgecode"

- 1st iteration :

s2 -> 6

s[i:i+s2] => s[5:5+6] => s[6]

s = "6,4,#georgecode"
0 1 2 3 4 5 6 7 8 9 10 11 12 13 14

result -> ["george"]

i -> 11

- 2nd iteration :

s2 -> 4

s[i:i+s2] => s[11:11+4] => s[4]

result -> ["george", "code"] ✓