# 394. Decode String

Given an encoded string, return its decoded string.

The encoding rule is: k[encoded\_string], where the encoded\_string inside the square brackets is being repeated exactly k times. Note that k is guaranteed to be a positive integer.

You may assume that the input string is always valid; there are no extra white spaces, square brackets are well-formed, etc. Furthermore, you may assume that the original data does not contain any digits and that digits are only for those repeat numbers, k. For example, there will not be input like 3a or 2[4].

The test cases are generated so that the length of the output will never exceed 105.

## SOLUTION IN C++

class Solution {

public:

string decodeString(string s) {

stack<pair<string, int>> stack; // (prevStr, repeatCount)

string currStr;

int currNum = 0;

for (const char c : s)

if (isdigit(c)) {

currNum = currNum \* 10 + (c - '0');

} else {

if (c == '[') {

stack.emplace(currStr, currNum);

currStr = "";

currNum = 0;

} else if (c == ']') {

const auto [prevStr, n] = stack.top();

stack.pop();

currStr = prevStr + getRepeatedStr(currStr, n);

} else {

currStr += c;

}

}

return currStr;

}

private:

string getRepeatedStr(const string& s, int n) {

string repeat;

while (n--)

repeat += s;

return repeat;

}

};