Decode Ways (/problems/decode-ways/)

# Submission Detail

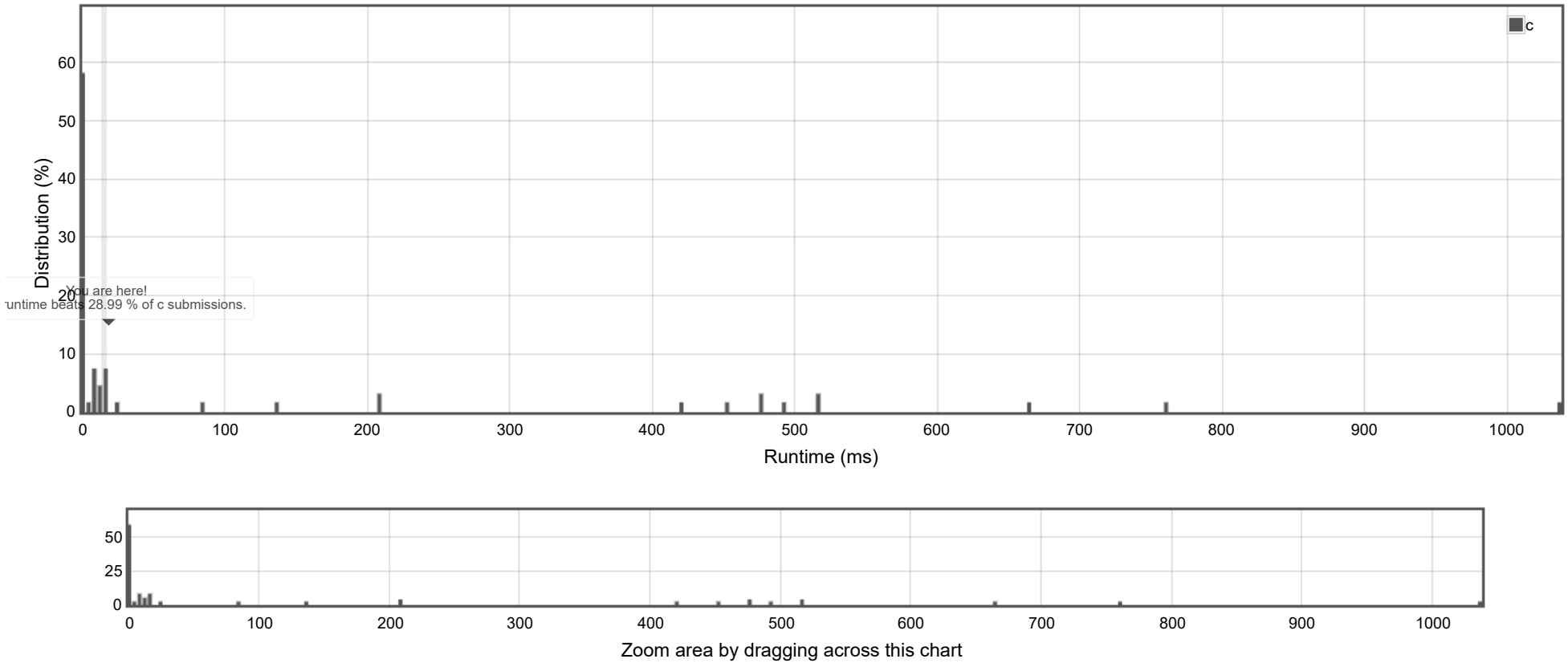
**258 / 258** test cases passed.

Status: Accepted

Runtime: **16 ms**

Submitted: **23 minutes ago**

## Accepted Solutions Runtime Distribution




Invite friends to challenge **Decode Ways**

18

## Submitted Code: 23 minutes ago

Language: c

Edit Code

```c
int numDecodings(char* s) {

    if (s[0] == '0') {
        return 0;
    }
    int *F = (int *)malloc(sizeof(int) * strlen(s));
    int *G = (int *)malloc(sizeof(int) * strlen(s));
    F[0] = 1;
    G[0] = 0;
    for (int i = 1; i < strlen(s); i++) {
        int can_alone = s[i] != '0';
        char tmp[3] = {s[i - 1], s[i], '\0'};
        int num = atoi(tmp);
        int can_combine = 1 <= num && num <= 26 && tmp[0] != 0;
        if (!can_alone && !can_combine) {
            return 0;
        }
        if (can_alone)
            F[i] = F[i - 1] + G[i - 1];
        else
            F[i] = 0;
        if (can_combine) {
            G[i] = F[i - 1];
        } else {
            G[i] = 0;
        }
    }
    int answer = F[strlen(s) - 1] + G[strlen(s) - 1];
    free(F);
    free(G);
    return answer;

}

```

Back to problem (/problems/decode-ways/)