

چالش نردبان

رسیدن به پله آخر مرحله به مرحله است و نیاز به رسیدن به پله‌های قبلش دارد. پس می‌توان با استفاده از نگاه استقرایی و برنامه‌نویسی پویا، مساله را اول برای پله‌ای قبلی و سپس برای پله آخر حل کنیم. برای توصیف زیرمساله، فقط به یک متغیر نیاز داریم که پله است. آرایه $dp[i]$ را راه‌های رسیدن به پله i ام تعریف می‌کنیم.

ایده کلی این است که در هر پله i ام، شارلوت یا از پله قبلی‌اش یا از دو پله پایین‌ترش به این پله رسیده است. پس رابطه بازگشتی برای پر کردن آرایه dp به این صورت می‌شود:

$$dp[i] = dp[i-1] + dp[i-2]$$

برای رسیدن به پله صفر و پله یک هم فقط به راه داریم که حالت‌های پایه ما هستند:

$$dp[0] = dp[1] = 1$$

```
function climb_stairs(n):
```

```
    dp[0] = 1
```

```
    dp[1] = 1
```

```
    for i from 2 to n:
```

```
        dp[i] = dp[i-1] + dp[i-2]
```

```
    return dp[n]
```