

تاس‌ها

این مساله شبیه مساله های کوله‌پشتی صفر و یک و خرد کردن پول است که باید مجموع تعدادی وزنه برابر X شود. با این تفاوت که حالت‌بندی وزنه‌ها متفاوت است.

برای توصیف زیرمساله به i به عنوان ایندکس آخرین تاس و به j به عنوان مجموع تاس‌ها نیاز داریم. زیرا مساله را می‌توان برای آخرین تاس حالت‌بندی کرد و برای اینکار به مجموع‌های مختلفی که تاس‌های قبلی می‌توانستند بسازند نیاز داریم.

پس $dp[i][j]$ را تعداد راه‌های ساخت مجموع j با پرتاب i تاس اول تعریف می‌کنیم. جواب مساله $dp[n][x]$ است.

برای پیدا کردن رابطه بازگشتی روی وجوه تاس i ام حالت‌بندی می‌کنیم:

$$dp[i][j] = dp[i-1][j-1] + dp[i-1][j-2] + \dots + dp[i-1][j-m]$$

البته با شرط مثبت بودن ایندکس دوم آرایه.

برای ساختن مجموع \bullet با \bullet تاس اول یک حالت وجود دارد، پس حالت پایه:

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

و برای ساختن مجموع \bullet با هر تعداد تاس \bullet حالت وجود دارد:

$$dp[i][0] = 0$$

و با \bullet تاس هیچ مجموعی را نمی‌توان ساخت:

$$dp[0][j] = 0$$

```
function toss_sum(n, m, x):
```

```
    dp = 2D array with size(n+1, x+1) initialized to 0
```

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

```
    for i from 1 to n:
```

```
        for j from 1 to x:
```

```
            for k from 1 to m:
```

```
                if( j >= k ):
```

```
                    dp[i][j] = dp[i-1][j-k]
```

```
    return dp[n][x]
```