

REKURZIO

Dice Combinations

Feladat:

hogya megszámolja, hány módon lehet n összeget létrehozni egy kocka egyszeri vagy többszöri dobásával. Minden dobás 1 és 6 közötti eredményt ad.

Például, ha $n=3$, akkor 4 módja van:

1+1+1

1+2

2+1

3

Bemenet

Az egyetlen bemeneti sorban van egy n egész szám.

Kimenet

Nyomtassa ki a modulo 10^9+7 módok számát.

Példa

Bemenet: 3

Kimenet: 4

Programkód:

```
Dice_Combinations_Rekurzio.py x
1 MOD = 10 ** 9 + 7
2 import sys
3 sys.setrecursionlimit(10**6) # Állítsuk a maximális rekurziós mélységet magasabb értékre
4 # Memorization tároló az eredményekhez
5 memo = {}
6
7 def kiszamitas(n): 2 usages
8     # Alapeset: Az összeg 0 elérésének egyetlen módja van (nem dobunk semmit)
9     if n == 0:
10         return 1
11     # Ha a szám kisebb, mint 0, nincs lehetséges megoldás
12     if n < 0:
13         return 0
14     # Ha az eredmény már ki van számolva, térjünk vissza a tárolt értékkel
15     if n in memo:
16         return memo[n]
17
18     # Számítsuk ki az n-hez vezető összes lehetőséget az utolsó legfeljebb 6 lépés alapján
19     memo[n] = 0
20     for i in range(1, 7):
21         memo[n] += kiszamitas(n - i)
22         memo[n] %= MOD # Tartsuk a számokat a MOD tartományon belül
23
24     return memo[n]
25
26 # Bemenet
27 n = int(input().strip())
28 print(kiszamitas(n))
```

Részletesen:

```
MOD = 10 ** 9 + 7
import sys
sys.setrecursionlimit(10**6) # Állítsuk a maximális rekurziós mélységet magasabb értékre
# Memorization tároló az eredményekhez
memo = {}
```

MOD = konstans érték amivel majd az eredmény kiíratásánál használunk fel
n=bevitel a terminálról str-ként majd letisztítva int-é alakítva
a rekurziós mélység miatt a teszt eredmény rosszabb volt.

```
def kiszamitas(n): 2 usages
    # Alapeset: Az összeg 0 elérésének egyetlen módja van (nem dobunk semmit)
    if n == 0:
        return 1
    # Ha a szám kisebb, mint 0, nincs lehetséges megoldás
    if n < 0:
        return 0
    # Ha az eredmény már ki van számolva, térjünk vissza a tárolt értékkel
    if n in memo:
        return memo[n]

    # Számítsuk ki az n-hez vezető összes lehetőséget az utolsó legfeljebb 6 lépés alapján
    memo[n] = 0
    for i in range(1, 7):
        memo[n] += kiszamitas(n - i)
    memo[n] %= MOD # Tartsuk a számokat a MOD tartományon belül
```

kiszamitas függvény n: Az a szám, amelyhez el szeretnénk érni az összes lehetséges kombinációk számát.

n==0 visszatér 1

n<0 visszatér 0

Ha a n-hez vezető kombinációk száma már korábban ki lett számolva, akkor az eredmény a memo szótárban van tárolva.

Ebben az esetben a függvény azonnal visszatér a tárolt értékkel, és nem számol újra, ezáltal időt takarít meg.

for ciklus

Ez a ciklus a dobókocka lehetséges értékeit járja végig (1-től 6-ig).

Az n-hez vezető utak számát úgy számítjuk ki, hogy megnézzük, az előző állapotok (n-1,n-2,...,n-6n-1, n-2, ..., n-6n-1,n-2,...,n-6) hányféleképpen érhetők el.

Rekurzívan meghívjuk a kiszamitas függvényt az n-i értékre, amely a n-hez vezető összes előző állapotot összegzi.

Az eredményt hozzáadjuk a memo[n] értékéhez. MOD-olva visszatér az értéket.