K-Distribution

Submissions Attempted by: **211** | Solved by: **73** | Partially Solved by: **122** |

Combinatorics Dynamic Programming Hard Math Number Theory Edit

In order to get at least some points you can try **all possible distributions** and for every of them check if requirements are satisfied. From *Constraints* section it is obvious that given solution should score at least **28 points**. But to get full score you need a different, faster approach.

In problem *K-distribution* you are asked about number of **valid partitions**. However, as it often happens in such tasks, it is easier to approach it from an opposite side. Let's count **bad partitions** – answer to original problem will be equal to total number of possible partitions minus number of bad partitions.

Counting bad partitions turns out to be a **standard DP problem** – it's one of modifications of **knapsack problem**. State of DP : *dp[sum] – number of ways to pick numbers which sum up to S*. Add numbers one by one and recalculate values of DP (read about knapsack problem first if you are not familiar with it). When calculating final answer, don't forget that two symmetric ways of distribution are counted as different.

Also you have to handle cases when both groups will have sum **less than K**. Depending on implementation, you may count them twice or not count at all. Look at solutions below to see how it can be handled (again, with some sort of **inclusion-exclusion idea**). However, it turns out that things are much simpler. When both groups have sum **less than K** – it means that total sum of all numbers is **less than 2*K**. But in this case answer will be simply **0**, and you don't even need to calculate values of DP table.

Edit Editorial

Author Solution by Pavel Sheftelevich

```cpp
#include <cstdio>
#include <algorithm>
#include <vector>
#include <iostream>
#include <set>
#include <map>
#include <iomanip>
#include <string>
#include <string.h>
#include <cstdlib>
#include <bitset>
#include <cmath>

#define X first
#define Y second
#define mp make_pair
#define pb push_back

typedef long long ll;

using namespace std;

const int MAXN = 110, MOD = 1E9 + 7, MAXK = 100100;
int n, a[MAXN], k;
int dp[MAXK];
ll totSum ;
int main() {
    cin>>n>>k;
    for (int i = 0; i < n; i++) {
        cin>>a[i];
        totSum += a[i];
    }
    dp[0] = 1;
    for (int i = 0; i < n; i++) {
        if (a[i] >= k) {
            continue;
        }
        for (int j = k - a[i]; j >= 0; j--) {
            dp[j + a[i] ] += dp[j];
            if (dp[j + a[i] ] >= MOD) {
                dp[j + a[i] ] -= MOD;
            }
        }
    }
    int sum = 0;
    for (int i = 0; i < k; i++) {
        //cerr<<i<<" : "<<dp[i]<<endl;
        sum += dp[i];
        if (sum >= MOD) {
            sum -= MOD;
        }

        if (i != totSum - i) {
```

```
                //cerr<<"fklfl"<<endl;
                sum += dp[i];
                if (sum >= MOD) {
                     sum -= MOD;
                }
           }
      }
      //cerr<<sum<<endl;
      int tot = 1;
      for (int i = 0; i < n; i++) {
           tot *= 2;
           if (tot >= MOD) {
                tot -= MOD;
           }
      }
      tot -= sum;
      if (tot < 0) {
           tot += MOD;
      }
      cout<<tot<<endl;
      return 0;
}
```

Tester Solution by Bohdan Pryshchenko

```
#include<bits/stdc++.h>
#define bs 1000000007

using namespace std;

int n,a[1<<20],ans,T;
int answ;
int dp[1<<20];
long long S;
long long rem;

void add(int &a,int b)
{
        a+=b;
        if (a>=bs)
                a-=bs;
}

int main(){

cin>>n;
cin>>T;
--T;

for (int i=1;i<=n;i++)
 cin>>a[i];

ans=1;
```

```
29. for (int i=1;i<=n;i++)
30.  ans*=2,
31.  ans%=bs;
32.
33. S=0;
34. for (int i=1;i<=n;i++)
35.         S+=a[i];
36.
37. dp[0]=1;
38. for (int i=1;i<=n;i++)
39.         for (int j=T-a[i];j>=0;--j)
40.                 if (dp[j]>0)
41.                         add(dp[j+a[i]],dp[j]);
42.
43. answ=0;
44. for (int i=0;i<=T;i++)
45. {
46.         rem=S-i;
47.         if (rem<=T)
48.                 add(answ,dp[i]);
49.         else
50.                 add(answ,2*dp[i]%bs);
51. }
52.
53. ans=(ans-answ+bs)%bs;
54. cout<<ans<<endl;
55.
56. return 0;}
```

COMMENTS (4) Refresh

Join Discussion...

Cancel Post

nagasiva sudhakarreddykovvuri 4 months ago

whyad add (answ,2*dp[i]%s);

Reply ● Message ● Permalink

Bohdan Pryshchenko 4 months ago

You need to count every distribution twice, because if you have some way to pick items with cost X from set with total cost S, it gives you 2 distributions: (X,S-X) and (S-X,X) - you may choose to put your items in either first or second part.

Reply ● Message ● Permalink

nagasiva sudhakarreddykovvuri 4 months ago

what abt add(answ,dp[i]);

Reply ● Message ● Permalink

Bohdan Pryshchenko 4 months ago

These are cases which will be counted only once, so you don't need *2 for them.

Reply ● Message ● Permalink

PROFILE IMPACT

Complete Profile

*Excellent profile will increase your profile discoverability and keep you on top among others.

PROBLEMS SUGGESTED FOR YOU

The function problem
Solved by 249

Game Of Strengths
Solved by 497

Putting matrix
Solved by 17

more...

RECENT SUBMISSIONS

User	Result	Time	Lang
Swapnil ...		2.7472	C++
Roman Bi...		1.7096	C
Himadri ...		13.4437	C++
Himadri ...		13.4778	C++
Parag Go...		1.7106	C++
Parag Go...		1.71	C++
Parag Go...		2.2292	C++

View All

TRENDING NOTES

Number Theory - III
written by Boris Sokolov

Exact String Matching Algorithms
written by Alei Reyes

Binary Indexed Tree or Fenwick Tree
written by Chandan Mittal

Small tricks in for loop
written by Rangeesh

Strings And String Functions
written by Vinay Singh

more ...

DEVELOPERS TO FOLLOW


সায়ান চৌধুরী
2160 followers


Hemesh Singh
6 followers


Sachin Gupta
6361 followers

COMPANIES TO FOLLOW

Gigstart
1717 followers

Zoomcar
552 followers

HARMAN
1971 followers

RECOMMENDED CHALLENGES

Horlicks Hack 4 Fun

03 Sep 2015, 09:00 PM IST

Register

CODE-HUNT-2F

21 Oct 2015, 05:00 PM IST

Register

Zoomcar Ruby Challenge

23 Oct 2015, 06:00 PM IST

Register

Zomato Hiring Challenge

23 Oct 2015, 06:00 PM IST

Register

Diona iOS Developer Hiring Challenge

24 Oct 2015, 12:00 PM IST

Register

Tipstat Android Developer Hiring Challenge

24 Oct 2015, 12:00 PM IST

Register

D'code

SUBSCRIBE TO HACKEREARTH NEWS

bhawnesh.dipu@gmail.com

Subscribe

JOIN PROGRAMMING CLUB ON FACEBOOK

Join now

IIIrd Floor, Salarpuria Business Center,
4th B Cross Road, 5th A Block,
Koramangala Industrial Layout,
Bangalore, Karnataka 560095, India.

contact@hackerearth.com
+91-80-4155-4695
+1-650-461-4192