The art of security [4]

Submissions Attempted by: **133** | Solved by: **90** | Partially Solved by: **30** | ★★★★☆

Combinatorics Greedy Medium Edit

Editorialist: Pawel Kacprzak

In this problem you are about to deal with N creatures entering the city. Each creature has its own power and there are 2 entrances to the city.

Creatures enter the city one by one in the following manner:

> While there are creatures outside the city:
>
> > One of creatures from outside the city decides to enter it. It picks one of its two entrances to do that and go there.

You see that these can produce different arrangements of creatures at the entrances. We call an arrangement a good arrangement if and only if **at any time**, the sum of powers of creatures standing in the first entrance is not greater than the sum of powers of creatures standing in the second entrance. Any other arrangement is bad.

Your task here is to count the number of good arrangements. Knowing this and the sum of powers of all creatures is enough to provide a correct output.

First observation that we can make is to notice that N is small enough to iterate over all permutations of creatures. For each such permutation we can consider at most 2^N arrangements of creatures to the entrances while taking into result just the good ones. It important to see that we do not have and should not consider any arrangement which is not good. So when we about to place a creature at the first entrance, we check if the produced arrangement is good and if it is not, we do not extend it any further.

Time complexity

The exact time complexity depends on values of powers of individual creatures. It is definitely $O(N! * 2^N)$ which is about $2 * 10^8$ operations, but for a fixed permutation of creatures, not extending bad arrangements further, allows us to avoid checking many unnecessary arrangements. For example, for a fixed permutation of creatures, placing the first creature in the permutation which has a positive power at the first entrance, will produce a bad arrangement so we will not consider any such arrangement. To sum up, this method will easily pass all testscases in time.

My solution is available here.

Edit Editorial

Author Solution by Arjit Srivastava

```
#include <bits/stdc++.h>
using namespace std;

int arr[13], n, ans = 0, entrance1, entrance2;

void solve(int num) {
        if(num == n){
                ans++;
                return;
        }

        if (entrance1 + arr[num] <= entrance2) {
                entrance1 += arr[num];
                solve(num + 1);
                entrance1 -= arr[num];
        }
        entrance2 += arr[num];
        solve(num + 1);
        entrance2 -= arr[num];
}

int main() {
        scanf("%d",&n);
        int sum = 0;
        for(int i=0; i<n; i++) {
                scanf("%d",&arr[i]);
                sum += arr[i];
        }
        sort(arr, arr+n);
        do{
                entrance1=0; entrance2=0;
                solve(0);
        }while(next_permutation(arr,arr+n));
        printf("%d %d\n",ans, sum);
        if (ans>sum)
                printf("We will win!\n");
        else
                printf("Got no way out!\n");
        return 0;
}
```

Tester Solution by FatalEagle

```
// The art of security
```

```
// Tester solution by FatalEagle
// O(N! * 2^N)

#include <bits/stdc++.h>

void assert_digit() {char c = getchar(); assert('0' <= c && c <=

using namespace std;

int N;
int A[8];
int p[8];

int main()
{
    assert_digit();
    scanf("%d", &N);
    assert(0<=N && N<=8);
    if(N==0)
        printf("1 0\nWe will win!\n");
    else
    {
        assert(getchar()=='\n');
        set<int> uniq;
        int sum=0;
        for(int i=0; i<N; i++)
        {
            assert_digit();
            scanf("%d", A+i);
            assert(0<=A[i] && A[i]<=500);
            if(i!=N-1)
                assert(getchar()==' ');
            sum+=A[i];
            uniq.insert(A[i]);
        }
        assert((int)uniq.size()==N);
        int ans=0;
        for(int i=0; i<N; i++)
            p[i]=i;
        do
        {
            function<int(int, int, int)> rec=[&](int j, int sa, i
                if(sa<sb)
                    return 0;
                if(j==N)
                    return 1;
                return rec(j+1, sa+A[p[j]], sb)+rec(j+1, sa, sb+A
            };
            ans+=rec(0, 0, 0);
        }
        while(next_permutation(p, p+N));
        printf("%d %d\n", ans, sum);
        if(ans<=sum)
            printf("Got no way out!\n");
```

```
56.          else
57.               printf("We will win!\n");
58.     }
59.     assert(getchar()==EOF);
60.     return 0;
61. }
```

PROFILE IMPACT

Complete Profile

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

PROBLEMS SUGGESTED FOR YOU

Flip the words
Solved by 14

Play the Base
Solved by 7

Fun With Sequences
Solved by 336

more...

RECENT SUBMISSIONS

User	Result	Time	Lang
anwar ah...		1.1116	C
Koushik		1.1069	C++
Koushik		1.1064	C++
Koushik		1.1068	C++
Rohil Ra...		1.1068	C++
Rohil Ra...		1.1067	C++
Rohil Ra...		1.1064	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


srajan dongre
1 followers


Lalit Kundu
2212 followers


Pradeep Choudhary
3 followers

COMPANIES TO FOLLOW

AT&T
2713 followers

iRageCapital
1003 followers

McAfee, Part of Intel Security
3879 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