Codeforces Round #713 (Div. 3)

A. Spy Detected!

A. Spy Detected!

time limit per test: 2 seconds
memory limit per test: 256 megabytes
input: standard input
output: standard output

You are given an array a consisting of n ($n \geq 3$) positive integers. It is known that in this array, all the numbers except one are the same (for example, in the array $[4, 11, 4, 4]$ all numbers except one are equal to 4).

Print the index of the element that does not equal others. The numbers in the array are numbered from one.

Input

The first line contains a single integer t ($1 \leq t \leq 100$). Then t test cases follow.

The first line of each test case contains a single integer n ($3 \leq n \leq 100$) — the length of the array a.

The second line of each test case contains n integers $a_1, a_2, \ldots, a_n$ ($1 \leq a_i \leq 100$).

It is guaranteed that all the numbers except one in the a array are the same.

Output

For each test case, output a single integer — the index of the element that is not equal to others.

Example

input Copy

```
4
4
11 13 11 11
5
1 4 4 4 4
10
3 3 3 3 10 3 3 3 3 3
3
20 20 10
```

output Copy

```
2
1
5
3
```

```cpp
#include <bits/stdc++.h>
using namespace std;
#define IOS ios::sync_with_stdio(0)
#define LL long long
#define maxn (int)(2e5 + 10)


int a[maxn];
int b[maxn];
int ans[maxn];
int main ()
{
    IOS;
    int T ; cin >> T;
    for ( int cas = 1 ; cas <= T ; cas++)
    {
        int n ; cin >> n ;
        for ( int i = 1; i <= n ; i++ ) cin >> a[i];
        memcpy( b , a , (n+5)*sizeof(int));
        sort ( b+1 , b+1+n ) ;
        int x ;
        if ( b[1] == b[2] ) x = b[n];
        else x = b[1];
        for ( int i = 1 ; i <= n ; i++ )
        {
            if ( a[i] == x )
            {
                ans[cas] = i ;
                break;
            }
        }
    }
    for ( int i = 1 ; i <= T ; i++ )
        cout << ans[i] << endl;
}
```

B. Almost Rectangle

B. Almost Rectangle

time limit per test: 2 seconds
memory limit per test: 256 megabytes
input: standard input
output: standard output

There is a square field of size $n \times n$ in which two cells are marked. These cells can be in the same row or column.

You are to mark two more cells so that they are the corners of a rectangle with sides parallel to the coordinate axes.

For example, if $n = 4$ and a rectangular field looks like this (there are asterisks in the marked cells):

```
. . * .
. . . .
* . . .
. . . .
```

Then you can mark two more cells as follows

```
* . * .
. . . .
* . * .
. . . .
```

If there are several possible solutions, then print any of them.

Input

The first line contains a single integer t ($1 \le t \le 400$). Then t test cases follow.

The first row of each test case contains a single integer n ($2 \le n \le 400$) — the number of rows and columns in the table.

The following n lines each contain n characters '.' or '*' denoting empty and marked cells, respectively.

It is guaranteed that the sums of n for all test cases do not exceed 400.

It is guaranteed that there are exactly two asterisks on the field. They can be in the same row/column.

It is guaranteed that the solution exists.

Output

For each test case, output n rows of n characters — a field with four asterisks marked corresponding to the statements. If there multiple correct answers, print any of them.

Example

input Copy

```
6
4
..*.
....
*...
....
2
*.
.*
2
.*
.*
3
*.*
...
...
5
.....
..*..
.....
.*...
.....
4
....
....
*...
*...
```

output Copy

```
*.*.
```

```
....
*.*.
....
**
**
**
**
*.*
*.*
...
.....
.**..
.....
.**..
.....
....
....
**..
**..
```



```cpp
#include <bits/stdc++.h>
using namespace std;
#define IOS ios::sync_with_stdio(0)
#define LL long long
#define maxn (int)(2e5 + 10)

int main ()
{
    int T ; cin >> T ;
    while (T--)
    {
        int n ; cin >> n ;
        string s[500];
        for ( int i = 1 ; i <= n ; i++ )
            cin >> s[i] , s[i] = ' ' + s[i];
        struct Point {
            int x ; int y ;
        };
        vector<Point> v;
        for ( int i = 1 ; i <= n ; i++ )
            for ( int j = 1 ; j <= n ; j++ )
                if ( s[i][j] == '*' )
                    v.push_back({i,j});
        Point a , b ;
        a = v.front() , b = v.back();
        if ( a.x != b.x && a.y != b.y )
            s[a.x][b.y] = s[b.x][a.y] = '*';
        else if ( a.x == b.x )
        {
            int d = 1;
```

```
            if ( a.x - d >= 1 )
                s[a.x-d][a.y] = s[a.x-d][b.y] = '*';
            else s[a.x+d][a.y] = s[a.x+d][b.y] = '*';
        }
        else if ( a.y == b.y )
        {
            int d = 1;
            if ( a.y - d >= 1 )
                s[a.x][a.y-d] = s[b.x][b.y-d] = '*';
            else s[a.x][a.y+d] = s[b.x][b.y+d] = '*';
        }
        for ( int i = 1 ; i <= n ; i++ )
        {
            for ( int j = 1 ; j <= n ; j++ )
                cout << s[i][j] ;
            cout << endl;
        }
    }

}
```

C. A-B Palindrome

CODEFORCES
Sponsored by Telegram

DYf3244 | Logout

HOME TOP CONTESTS GYM PROBLEMSET GROUPS RATING EDU API CALENDAR HELP ABOUT STATUS

PROBLEMS SUBMIT CODE MY SUBMISSIONS STATUS HACKS STANDINGS CUSTOM INVOCATION

C. A-B Palindrome

time limit per test: 2 seconds
memory limit per test: 256 megabytes
input: standard input
output: standard output

You are given a string s consisting of the characters '0', '1', and '?'. You need to replace all the characters with '?' in the string s by '0' or '1' so that the string becomes a palindrome and has **exactly** a characters '0' and **exactly** b characters '1'. Note that each of the characters '?' is replaced **independently** from the others.

A string t of length n is called a palindrome if the equality $t[i] = t[n - i + 1]$ is true for all i ($1 \le i \le n$).

For example, if s ="01?????0", $a = 4$ and $b = 4$, then you can replace the characters '?' in the following ways:

- "01011010";
- "01100110".

For the given string s and the numbers a and b, replace all the characters with '?' in the string s by '0' or '1' so that the string becomes a palindrome and has **exactly** a characters '0' and **exactly** b characters '1'.

Input

The first line contains a single integer t ($1 \le t \le 10^4$). Then t test cases follow.

The first line of each test case contains two integers a and b ($0 \le a, b \le 2 \cdot 10^5$, $a + b \ge 1$).

The second line of each test case contains the string s of length $a + b$, consisting of the characters '0', '1', and '?'.

It is guaranteed that the sum of the string lengths of s over all test cases does not exceed $2 \cdot 10^5$.

Codeforces Round #713 (Div. 3)
Finished
Practice

→ Virtual participation

Virtual contest is a way to take part in past contest, as close as possible to participation on time. It is supported only ICPC mode for virtual contests. If you've seen these problems, a virtual contest is not for you - solve these problems in the archive. If you just want to solve some problem from a contest, a virtual contest is not for you - solve this problem in the archive. Never use someone else's code, read the tutorials or communicate with other person during a virtual contest.

Start virtual contest

→ Practice

You are registered for practice. You can solve problems unofficially. Results can be found in the contest status and in the bottom of standings.

→ Clone Contest to Mashup

You can clone this contest to a mashup.

Clone Contest

Output

For each test case, output:

- "-1", if you can't replace all the characters '?' in the string s by '0' or '1' so that the string becomes a palindrome and that it contains **exactly** a characters '0' and **exactly** b characters '1';
- the string that is obtained as a result of the replacement, otherwise.

If there are several suitable ways to replace characters, you can output any.

Example

input Copy

```
9
4 4
01?????0
3 3
??????
1 0
?
2 2
0101
2 2
01?0
0 1
0
0 3
1?1
2 2
?00?
4 3
??010?0
```

output Copy

```
01011010
-1
0
-1
0110
-1
111
1001
0101010
```

→ Submit?

Language: GNU G++11 5.1.0

Choose file: 浏览... 未选择文件。

Submit

→ Last submissions

Submission	Time	Verdict
112642969	Apr/11/2021 13:50	Accepted
112642044	Apr/11/2021 13:39	Wrong answer on test 2

→ Problem tags

constructive algorithms implementation strings

No tag edit access

→ Contest materials

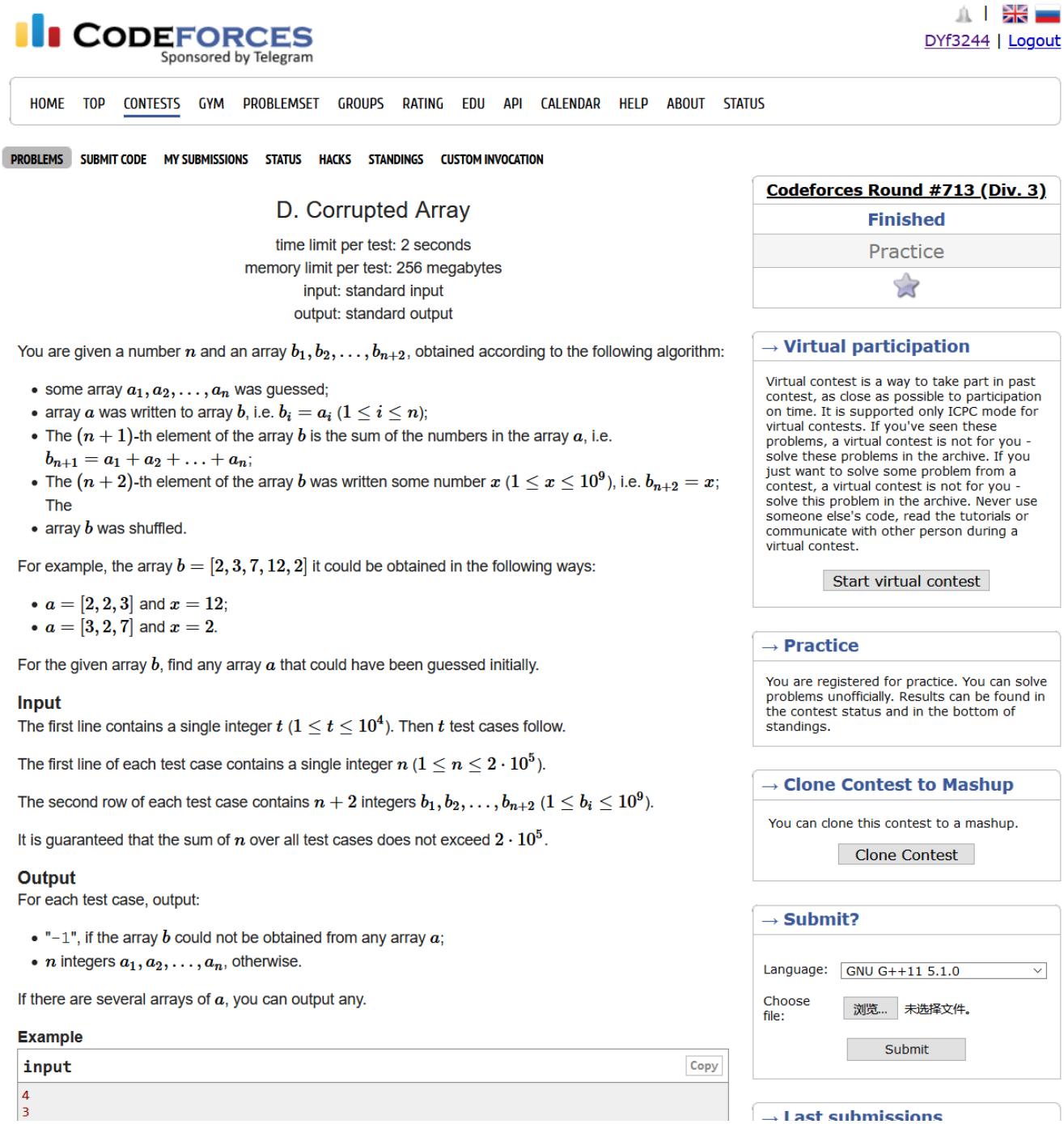
- Announcement
- Tutorial


```cpp
#include <bits/stdc++.h>
using namespace std;
#define IOS ios::sync_with_stdio(0)
#define LL long long
#define maxn (int)(1e6 + 10)

string s;
bool ans_flag[maxn];
string ans[maxn];

int main ()
{
    IOS;
    int T ; cin >> T;
    for ( int cas = 1 ; cas <= T ; cas++)
    {
```

```
        int a , b ; cin >> a >> b ;
        cin >> s; s = ' ' + s ;
        int n = a+b;
        if ( (a & 1) && (b & 1) )
            ans_flag[cas] = 0;
        else
        {
            bool flag = 1;
            int cnt0 = 0 , cnt1 = 0 ;
            for ( int i = 1 ; i <= n ; i++ )
            {
                if ( s[i] == '0' )
                {
                    if ( s[n-i+1] == '1' )
                    {
                        flag = 0 ; break;
                    }
                    cnt0++;
                }
                else if ( s[i] == '1' )
                {
                    if ( s[n-i+1] == '0' )
                    {
                        flag = 0 ; break;
                    }
                    cnt1++;
                }
                else if ( s[i] == '?' )
                {
                    if ( s[n-i+1] == '0' )
                    {
                        s[i] = '0';
                        cnt0++;
                    }
                    else if ( s[n-i+1] == '1' )
                    {
                        s[i] = '1';
                        cnt1++;
                    }
                }
            }
            if ( cnt0 > a || cnt1 > b ) flag = 0 ;
            if ( a & 1 && s[n/2+1] == '1' ) flag = 0 ;
            if ( b & 1 && s[n/2+1] == '0' ) flag = 0 ;
            ans_flag[cas] = flag ;
            if ( flag )
            {
                if ( n & 1 && s[n/2+1] == '?' )
                    if ( a & 1 ) s[n/2+1] = '0' , a--;
                    if ( b & 1 ) s[n/2+1] = '1' , b--;
                a -= cnt0 ; b -= cnt1 ;
                for ( int i = 1 ; i <= n/2 ; i++ )
                {
                    if ( s[i] != '?' ) continue;
```

```
                    if (a > 0) s[i] = s[n-i+1] = '0' , a -= 2;
                    else if (b > 0) s[i] = s[n-i+1] = '1' , b -= 2;
                }
            }
            for ( int i = 1 ; i <= n ; i++ )
                ans[cas].push_back(s[i]);
        }
    }
    for ( int i = 1 ; i <= T ; i++ )
        if ( ans_flag[i] )
            cout << ans[i] << endl;
        else cout << "-1" << endl;
}
```

D. Corrupted Array



CODEFORCES
Sponsored by Telegram

DYf3244 | Logout

HOME TOP CONTESTS GYM PROBLEMSET GROUPS RATING EDU API CALENDAR HELP ABOUT STATUS

PROBLEMS SUBMIT CODE MY SUBMISSIONS STATUS HACKS STANDINGS CUSTOM INVOCATION

D. Corrupted Array

time limit per test: 2 seconds
memory limit per test: 256 megabytes
input: standard input
output: standard output

You are given a number n and an array $b_1, b_2, \ldots, b_{n+2}$, obtained according to the following algorithm:

- some array $a_1, a_2, \ldots, a_n$ was guessed;
- array a was written to array b, i.e. $b_i = a_i$ $(1 \le i \le n)$;
- The $(n+1)$-th element of the array b is the sum of the numbers in the array a, i.e. $b_{n+1} = a_1 + a_2 + \ldots + a_n$;
- The $(n+2)$-th element of the array b was written some number x $(1 \le x \le 10^9)$, i.e. $b_{n+2} = x$; The
- array b was shuffled.

For example, the array $b = [2, 3, 7, 12, 2]$ it could be obtained in the following ways:

- $a = [2, 2, 3]$ and $x = 12$;
- $a = [3, 2, 7]$ and $x = 2$.

For the given array b, find any array a that could have been guessed initially.

Input

The first line contains a single integer t $(1 \le t \le 10^4)$. Then t test cases follow.

The first line of each test case contains a single integer n $(1 \le n \le 2 \cdot 10^5)$.

The second row of each test case contains $n+2$ integers $b_1, b_2, \ldots, b_{n+2}$ $(1 \le b_i \le 10^9)$.

It is guaranteed that the sum of n over all test cases does not exceed $2 \cdot 10^5$.

Output

For each test case, output:

- "-1", if the array b could not be obtained from any array a;
- n integers $a_1, a_2, \ldots, a_n$, otherwise.

If there are several arrays of a, you can output any.

Example

input Copy

```
4
3
```

Codeforces Round #713 (Div. 3)
Finished
Practice

→ Virtual participation

Virtual contest is a way to take part in past contest, as close as possible to participation on time. It is supported only ICPC mode for virtual contests. If you've seen these problems, a virtual contest is not for you - solve these problems in the archive. If you just want to solve some problem from a contest, a virtual contest is not for you - solve this problem in the archive. Never use someone else's code, read the tutorials or communicate with other person during a virtual contest.

Start virtual contest

→ Practice

You are registered for practice. You can solve problems unofficially. Results can be found in the contest status and in the bottom of standings.

→ Clone Contest to Mashup

You can clone this contest to a mashup.

Clone Contest

→ Submit?

Language: GNU G++11 5.1.0

Choose file: 浏览... 未选择文件。

Submit

→ Last submissions

```
2 3 7 12 2
4
9 1 7 1 6 5
5
18 2 2 3 2 9 2
3
2 6 9 2 1
```

output Copy

```
2 3 7
-1
2 2 2 3 9
1 2 6
```

Submission	Time	Verdict
112644509	Apr/11/2021 14:06	Accepted

→ **Problem tags**

constructive algorithms | data structures | greedy

No tag edit access

→ **Contest materials**

- Announcement
- Tutorial


```cpp
#include <bits/stdc++.h>
using namespace std;
#define IOS ios::sync_with_stdio(0)
#define LL long long
#define maxn (int)(2e5 + 10)

vector<int> ans[maxn];
LL a[maxn];
int n;

int is_the_last()
{
    LL sum = 0 ;
    for ( int i = 1 ; i < n ; i++ )
        sum += a[i];
    for ( int i = 1 ; i < n ; i++ )
    {
        if ( sum - a[i] == a[n] )
            return i;
    }
    return -1;
}
int main ()
{
    IOS;
    int T ; cin >> T;
    for ( int cas = 1 ; cas <= T ; cas++)
    {
        cin >> n;
```

```
        n += 2;
        for ( int i = 1; i <= n ; i++ ) cin >> a[i];
        sort ( a+1 , a+1+n );
        int res = is_the_last();
        if ( res != -1 )
        {
            for ( int i = 1 ; i < n ; i++ )
            {
                if ( i == res ) continue;
                ans[cas].push_back(a[i]);
            }
        }
        else
        {
            LL sum =  0;
            for ( int i = 1 ; i <= n-2 ; i++ )
                sum += a[i];
            if ( sum == a[n-1] )
            {
                for ( int i = 1 ; i <= n-2 ; i++ )
                    ans[cas].push_back(a[i]);
            }
            else ans[cas].push_back(-1);
        }
    }
    for ( int i = 1 ; i <= T ; i++ )
    {
        for ( int x : ans[i] )
            cout << x << " ";
        cout << endl;
    }
}
```

E. Permutation by Sum

CODEFORCES
Sponsored by Telegram

DYf3244 | Logout

HOME TOP CONTESTS GYM PROBLEMSET GROUPS RATING EDU API CALENDAR HELP ABOUT STATUS

PROBLEMS SUBMIT CODE MY SUBMISSIONS STATUS HACKS STANDINGS CUSTOM INVOCATION

E. Permutation by Sum

time limit per test: 2 seconds
memory limit per test: 256 megabytes
input: standard input
output: standard output

A permutation is a sequence of n integers from 1 to n, in which all the numbers occur exactly once. For example, $[1]$, $[3, 5, 2, 1, 4]$, $[1, 3, 2]$ are permutations, and $[2, 3, 2]$, $[4, 3, 1]$, $[0]$ are not.

Polycarp was given four integers n, l, r ($1 \le l \le r \le n$) and s ($1 \le s \le \frac{n(n+1)}{2}$) and asked to find a permutation p of numbers from 1 to n that satisfies the following condition:

- $s = p_l + p_{l+1} + \ldots + p_r$.

For example, for $n = 5$, $l = 3$, $r = 5$, and $s = 8$, the following permutations are suitable (not all

Codeforces Round #713 (Div. 3)
Finished
Practice

→ Virtual participation

Virtual contest is a way to take part in past contest, as close as possible to participation on time. It is supported only ICPC mode for virtual contests. If you've seen these problems, a virtual contest is not for you - solve these problems in the archive. If you just want to solve some problem from a contest, a virtual contest is not for you - solve this problem in the archive. Never use someone else's code, read the tutorials or communicate with other person during a

options are listed):

- $p = [3, 4, 5, 2, 1]$;
- $p = [5, 2, 4, 3, 1]$;
- $p = [5, 2, 1, 3, 4]$.

But, for example, there is no permutation suitable for the condition above for $n = 4$, $l = 1$, $r = 1$, and $s = 5$.

Help Polycarp, for the given n, l, r, and s, find a permutation of numbers from 1 to n that fits the condition above. If there are several suitable permutations, print any of them.

Input
The first line contains a single integer t ($1 \le t \le 500$). Then t test cases follow.

Each test case consist of one line with four integers n ($1 \le n \le 500$), l ($1 \le l \le n$), r ($l \le r \le n$), s ($1 \le s \le \frac{n(n+1)}{2}$).

It is guaranteed that the sum of n for all input data sets does not exceed 500.

Output
For each test case, output on a separate line:

- n integers — a permutation of length n that fits the condition above if such a permutation exists;
- -1, otherwise.

If there are several suitable permutations, print any of them.

Example

input Copy

```
5
5 2 3 5
5 3 4 1
3 1 2 4
2 2 2 2
2 1 1 3
```

output Copy

```
1 2 3 4 5
-1
1 3 2
1 2
-1
```

virtual contest.

Start virtual contest

→ Practice
You are registered for practice. You can solve problems unofficially. Results can be found in the contest status and in the bottom of standings.

→ Clone Contest to Mashup
You can clone this contest to a mashup.

Clone Contest

→ Submit?
Language: GNU G++11 5.1.0

Choose file: 浏览... 未选择文件。

Submit

→ Last submissions

Submission	Time	Verdict
112647072	Apr/11/2021 14:34	Accepted

→ Problem tags
greedy math

No tag edit access

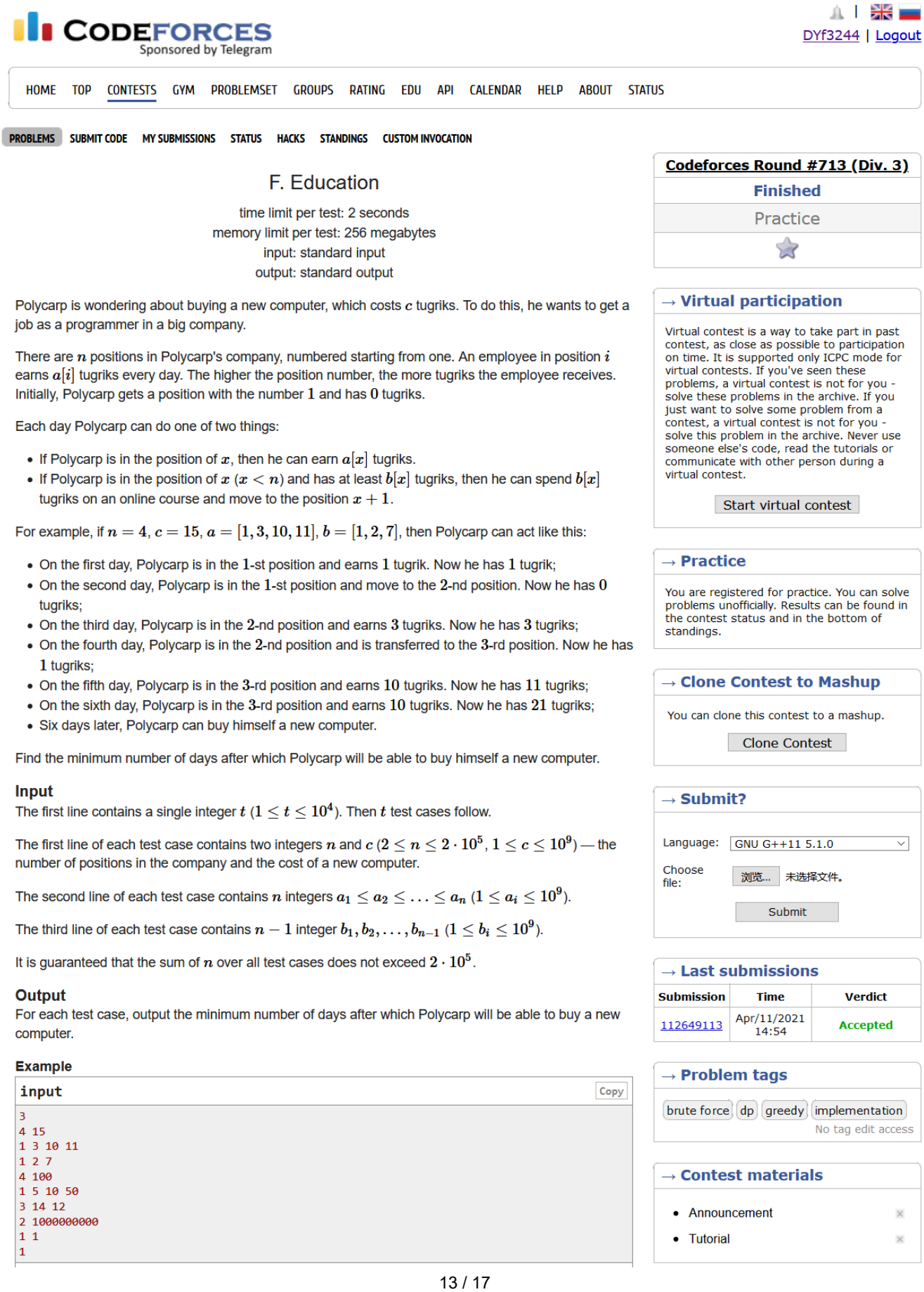
→ Contest materials
- Announcement
- Tutorial


```cpp
#include <bits/stdc++.h>
using namespace std;
#define IOS ios::sync_with_stdio(0)
#define LL long long
#define maxn (int)(2e5 + 10)

vector<int> ans[505];
int n , l , r , s;
int main ()
{
    IOS;
```

```
    int T ; cin >> T;
    for ( int cas = 1 ; cas <= T ; cas++)
    {
        cin >> n >> l >> r >> s;
        int d = r - l + 1 ;
        int minx = d*(d+1)/2 , maxx = ( 2*n-d+1 ) * d / 2;
        if ( s < minx || s > maxx )
        {
            ans[cas].push_back(-1);
            continue;
        }
        int cnt = 0 ;
        while ( minx < s )
        {
            minx = minx + n - cnt - d ;
            d-- ; cnt++ ;
        }
        vector<int> v;
        if ( minx > s )
        {
            int dis = minx - s ;
            v.push_back(n-cnt+1-dis);
            cnt--;
        }
        for ( int i = 1 ; i <= d ; i++ )
            v.push_back(i);
        for ( int i = n ; i >= n-cnt+1 ; i-- )
            v.push_back(i);
        bool vis[505] = {false};
        for ( int x : v ) vis[x] = 1;
        int cur = 1 ;
        for ( int i = 1 ; i < l ; i++ )
        {
            while(vis[cur++]);
            cur--;
            ans[cas].push_back(cur);
            vis[cur] = 1;
        }
        for ( int x : v )
            ans[cas].push_back(x);
        for ( int i = r+1 ; i <= n ; i++ )
        {
            while(vis[cur++]);
            cur--;
            ans[cas].push_back(cur);
            vis[cur] = 1;
        }
    }
    for ( int i = 1 ; i <= T ; i++ )
    {
        for ( int x : ans[i] )
            cout << x << " ";
        cout << endl;
```

```
    }
}
```

F. Education


CODEFORCES
Sponsored by Telegram
DYf3244 | Logout
HOME TOP CONTESTS GYM PROBLEMSET GROUPS RATING EDU API CALENDAR HELP ABOUT STATUS
PROBLEMS SUBMIT CODE MY SUBMISSIONS STATUS HACKS STANDINGS CUSTOM INVOCATION
F. Education
time limit per test: 2 seconds
memory limit per test: 256 megabytes
input: standard input
output: standard output
Polycarp is wondering about buying a new computer, which costs c tugriks. To do this, he wants to get a job as a programmer in a big company.
There are n positions in Polycarp's company, numbered starting from one. An employee in position i earns a[i] tugriks every day. The higher the position number, the more tugriks the employee receives. Initially, Polycarp gets a position with the number 1 and has 0 tugriks.
Each day Polycarp can do one of two things:
If Polycarp is in the position of x, then he can earn a[x] tugriks.
If Polycarp is in the position of x (x < n) and has at least b[x] tugriks, then he can spend b[x] tugriks on an online course and move to the position x + 1.
For example, if n = 4, c = 15, a = [1, 3, 10, 11], b = [1, 2, 7], then Polycarp can act like this:
On the first day, Polycarp is in the 1-st position and earns 1 tugrik. Now he has 1 tugrik;
On the second day, Polycarp is in the 1-st position and move to the 2-nd position. Now he has 0 tugriks;
On the third day, Polycarp is in the 2-nd position and earns 3 tugriks. Now he has 3 tugriks;
On the fourth day, Polycarp is in the 2-nd position and is transferred to the 3-rd position. Now he has 1 tugriks;
On the fifth day, Polycarp is in the 3-rd position and earns 10 tugriks. Now he has 11 tugriks;
On the sixth day, Polycarp is in the 3-rd position and earns 10 tugriks. Now he has 21 tugriks;
Six days later, Polycarp can buy himself a new computer.
Find the minimum number of days after which Polycarp will be able to buy himself a new computer.
Input
The first line contains a single integer t (1 ≤ t ≤ 10^4). Then t test cases follow.
The first line of each test case contains two integers n and c (2 ≤ n ≤ 2 · 10^5, 1 ≤ c ≤ 10^9) — the number of positions in the company and the cost of a new computer.
The second line of each test case contains n integers a_1 ≤ a_2 ≤ ... ≤ a_n (1 ≤ a_i ≤ 10^9).
The third line of each test case contains n − 1 integer b_1, b_2, ..., b_{n−1} (1 ≤ b_i ≤ 10^9).
It is guaranteed that the sum of n over all test cases does not exceed 2 · 10^5.
Output
For each test case, output the minimum number of days after which Polycarp will be able to buy a new computer.
Example
input
Copy
3
4 15
1 3 10 11
1 2 7
4 100
1 5 10 50
3 14 12
2 1000000000
1 1
1
Codeforces Round #713 (Div. 3)
Finished
Practice
→ Virtual participation
Virtual contest is a way to take part in past contest, as close as possible to participation on time. It is supported only ICPC mode for virtual contests. If you've seen these problems, a virtual contest is not for you - solve these problems in the archive. If you just want to solve some problem from a contest, a virtual contest is not for you - solve this problem in the archive. Never use someone else's code, read the tutorials or communicate with other person during a virtual contest.
Start virtual contest
→ Practice
You are registered for practice. You can solve problems unofficially. Results can be found in the contest status and in the bottom of standings.
→ Clone Contest to Mashup
You can clone this contest to a mashup.
Clone Contest
→ Submit?
Language: GNU G++11 5.1.0
Choose file:
浏览...
未选择文件。
Submit
→ Last submissions
Submission
Time
Verdict
112649113
Apr/11/2021 14:54
Accepted
→ Problem tags
brute force
dp
greedy
implementation
No tag edit access
→ Contest materials
Announcement
Tutorial

output Copy

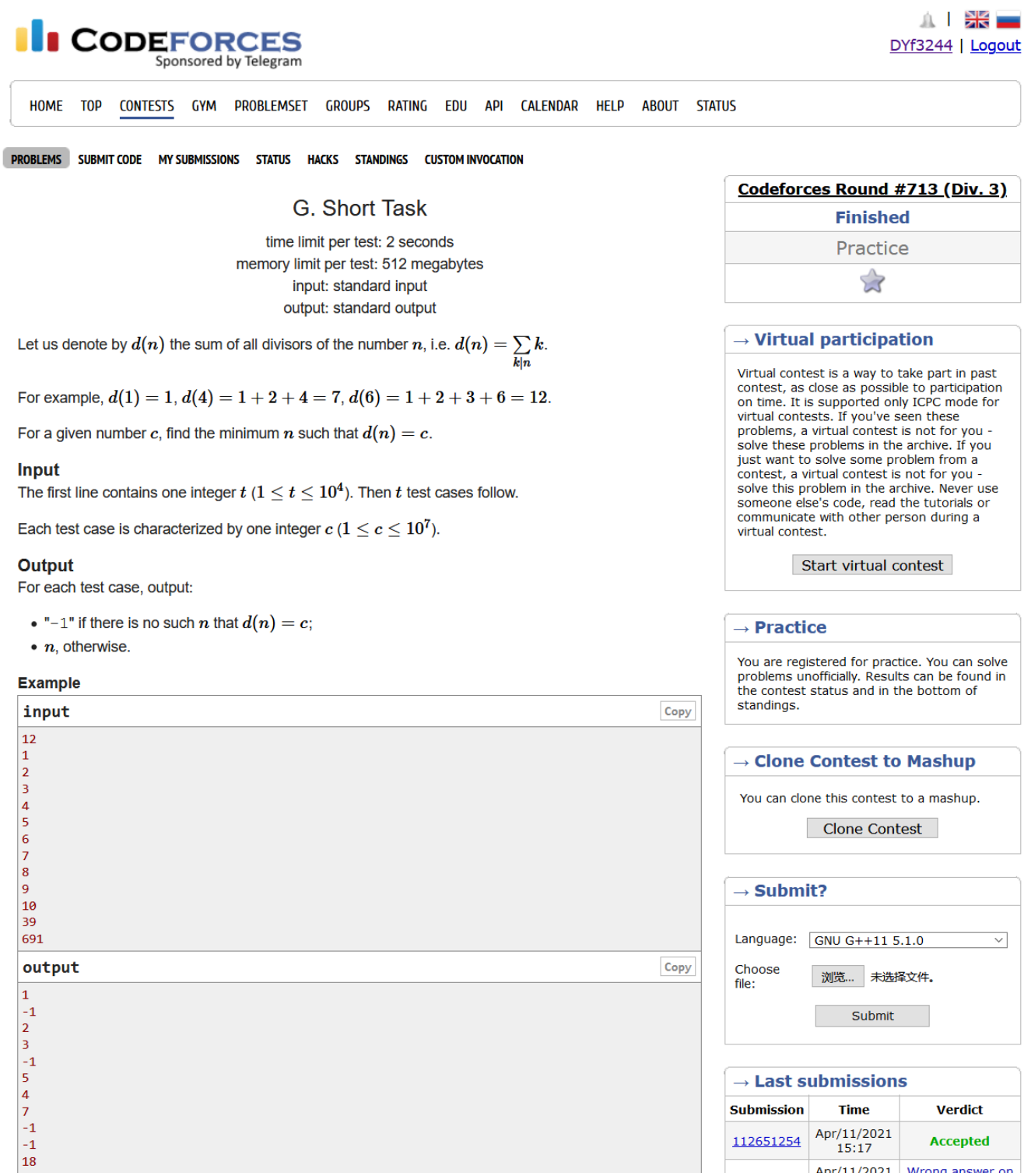
```
6
13
1000000000
```



```cpp
#include <bits/stdc++.h>
using namespace std;
#define IOS ios::sync_with_stdio(0);
#define LL long long
#define maxn (int)(2e5 + 10)

LL ans[maxn];
LL a[maxn] , b[maxn];
LL days[maxn] , leave[maxn];
int n , c ;

LL get_day( LL x , LL y )
{
    if ( y <= 0 ) return 0;
    //if ( y <= x ) return 1;
    LL day = y / x ;
    if ( y % x ) day++;
    return day;
}
int main ()
{
    IOS;
    int T ; cin >> T;
    for ( int cas = 1 ; cas <= T ; cas++)
    {
        cin >> n >> c ;
        for ( int i = 1; i <= n ; i++ ) cin >> a[i];
        for ( int i = 1; i < n ; i++ ) cin >> b[i];
        leave[1] = 0 ;
        days[1] = get_day(a[1],b[1]) + 1;
        leave[2] = a[1]*get_day(a[1],b[1]) - b[1];
        for ( int i = 2 ; i < n ; i++ )
        {
            LL day = get_day(a[i],b[i]-leave[i]);
            days[i] = days[i-1] + day + 1;
            leave[i+1] = leave[i] + a[i]*day - b[i];
        }
```

```
        LL minx = 1ll << 60 ;
        for ( int i = 1 ; i <= n ; i++ )
        {
            minx = min ( minx , days[i-1] + get_day(a[i],c-leave[i]));
        }
        ans[cas] = minx ;
    }
    for ( int i = 1 ; i <= T ; i++ )
        cout << ans[i] << endl;
}
```

G. Short Task



CODEFORCES
Sponsored by Telegram

DYf3244 | Logout

HOME TOP CONTESTS GYM PROBLEMSET GROUPS RATING EDU API CALENDAR HELP ABOUT STATUS

PROBLEMS SUBMIT CODE MY SUBMISSIONS STATUS HACKS STANDINGS CUSTOM INVOCATION

G. Short Task

time limit per test: 2 seconds
memory limit per test: 512 megabytes
input: standard input
output: standard output

Let us denote by $d(n)$ the sum of all divisors of the number n, i.e. $d(n) = \sum_{k|n} k$.

For example, $d(1) = 1$, $d(4) = 1 + 2 + 4 = 7$, $d(6) = 1 + 2 + 3 + 6 = 12$.

For a given number c, find the minimum n such that $d(n) = c$.

Input
The first line contains one integer t ($1 \le t \le 10^4$). Then t test cases follow.

Each test case is characterized by one integer c ($1 \le c \le 10^7$).

Output
For each test case, output:

- "-1" if there is no such n that $d(n) = c$;
- n, otherwise.

Example

input Copy

```
12
1
2
3
4
5
6
7
8
9
10
39
691
```

output Copy

```
1
-1
2
3
-1
5
4
7
-1
-1
18
```

Codeforces Round #713 (Div. 3)
Finished
Practice

→ Virtual participation

Virtual contest is a way to take part in past contest, as close as possible to participation on time. It is supported only ICPC mode for virtual contests. If you've seen these problems, a virtual contest is not for you - solve these problems in the archive. If you just want to solve some problem from a contest, a virtual contest is not for you - solve this problem in the archive. Never use someone else's code, read the tutorials or communicate with other person during a virtual contest.

Start virtual contest

→ Practice

You are registered for practice. You can solve problems unofficially. Results can be found in the contest status and in the bottom of standings.

→ Clone Contest to Mashup

You can clone this contest to a mashup.

Clone Contest

→ Submit?

Language: GNU G++11 5.1.0

Choose file: 浏览... 未选择文件。

Submit

→ Last submissions

Submission	Time	Verdict
112651254	Apr/11/2021 15:17	Accepted
	Apr/11/2021	Wrong answer on

```
-1
```

112649900	Apr/11/2021 15:03	Wrong answer on test 3

→ Problem tags

brute force | dp | math | number theory

No tag edit access

→ Contest materials

- Announcement
- Tutorial


```cpp
#include <bits/stdc++.h>
using namespace std;
#define IOS ios::sync_with_stdio(0);
#define LL long long
#define maxn (int)(1e7 + 10)

int ans[maxn];
int d[maxn];
LL s[maxn];
int res[maxn];
void get_fac()
{
    d[1] = 1;
    for (int i = 2; i * i < maxn; i++)
    {
        if (!d[i])
        {
            d[i] = i;
            for (int j = i * i; j < maxn; j += i)
                if (!d[j])
                    d[j] = i;
        }
    }
    s[1] = 1;
    for ( int i = 2 ; i < maxn ; i++ )
    {
        if (!d[i])
        {
            d[i] = i ;
            s[i] = i+1;
        }
        else
```

```
        {
            int j = i ;
            s[i] = 1;
            while ( j % d[i] == 0 )
            {
                j /= d[i];
                s[i] = s[i] * d[i] + 1;
            }
            s[i] *= s[j];
        }
    }
    for ( int i = maxn-1 ; i > 0 ; i-- )
        if ( s[i] < maxn )
            res[s[i]] = i;
}

int main()
{
    IOS;
    get_fac();
    int T;
    cin >> T;
    for (int cas = 1; cas <= T; cas++)
    {
        int n;
        cin >> n;
        ans[cas] = res[n];
    }
    for (int i = 1; i <= T; i++)
        if (ans[i] == 0)
            cout << "-1" << endl;
        else
            cout << ans[i] << endl;
}
```