

ACM.TUIT.UZ

Contest 1 Review

© Qurbon Raxmanov & Sirojiddin Abdugarimov

C1A. Teskari tartib

n natural soni hamda n ta butun sonlar ketma-ketligi berilgan. Shu sonlarni teskari tartibda chiqaruvchi dastur tuzing.

Kiruvchi ma'lumotlar:

Birinchi satrda n ($1 \leq n < 1000$) natural soni berilgan. Ikkinchi satrda esa n ta butun son probel bilan ajratilgan holda berilgan. Sonlar $(-10^9; 10^9)$ oraliqda berilgan.

Chiquvchi ma'lumotlar:

Bitta satrda berilgan n ta sonni teskari tartibda probel bilan chiqaring.

Yechish g'oyasi

Berilgan masalamiz boshlovchilar uchun mo'ljallangan. Bu masalani yechishning birinchi usulida oddiy bir o'lchovli massivga sonlarni saqlab olib uni teskari tartibda chiqarishdan iborat. Massivni teskari chiqarish uchun siklni teskari tashkil qila olish kifoya.

Teskari sikl:

```
for (i = n-1; i >= 0; i --)
```

2-usul

Bu masalani boshqa usulda ham yechish mumkin. Ya'ni massivdan foydalanmasdan ham yechish mumkin.

Massivsiz qanday yechish mumkin?

Massivdan foydalanmasdan turib rekursiv usuldan foydalanish mumkin. Bu usulda yechimni ko'rsatib o'tamiz.

```
void rec(int n){  
    int a;  
    cin >> a;  
    if(n > 1) rec(n -1);  
    cout << a << " ";  
}
```

C1B. Ko'priklar

TATU talabalari sayohatga chiqishmoqchi. Ular sayohatga avtobusda borishmoqchi. Ularning avtobusning balandligi 437 sm ga teng. Ularning yo'llarida bir nechta ko'priklar mavjud. Afsuski ko'priklarning hammasi ham baland emas. Ular hamma ko'priklarni balandliklarini bilishadi. Talabalar orasidagi Adham ular qaysi ko'prikga yetib borganlarida sayohatni to'xtatishga majbur bo'lishlarini bilmoqchi. U sizlardan yordam so'rayapti. Unga yordam bering.

Kiruvchi ma'lumotlar:

Birinchi satrda ko'priklar soni n ($1 \leq n \leq 1000$), ikkinchi satrda esa ketma-ket ko'priklarning balandliklari (sm hisobida) berilgan.

Chiquvchi ma'lumotlar:

Siz Adhamga yordam bering. Ular qaysi ko'prikga borib o'ta olmasliklarini aniqlang. Masalan k-ko'prikda o'ta olishmasa "Crash" k ni chiqaring. Agar hamma ko'prikda o'ta olishsa "No Crash" chiqaring. Katta va kichik harflarga e'tibor bering.

Yechish g'oyasi

Bu masalani yechish uchun birinchi ko'priklar balandliklarini massivga saqlab olamiz va massivimizni bir chekkadan tekshirib boramiz. Agar i - ko'prik balandligi 437 dan kichik yoki teng bo'lsa `cout << "Crash " << i; ni` pechatga beramiz va dastur ishini yakunlovchi buyruqni beramiz, ya'ni `return 0;` Sikl tugaganidan keyin esa "No Crash" pechatga beriladi.

C1C. Musbat, nol va manfiy

Bitta satrda bir-biri bilan probel bilan ajratilgan sonlar berilgan (nechtaligi noma'lum). Ulardan nechitasi musbat, nechitasi nol va nechitasi manfiy ga tengligini aniqlovchi dastur tuzing.

Kiruvchi ma'lumotlar:

Bitta satrda probel bilan ajratilgan butun sonlar berilgan. Sonlar modul jihatdan 1000000000 dan katta emas. Sonlar soni 10000 dan kam ekanligi kafolatlangan.

Chiquvchi ma'lumotlar:

Bitta satrda 3 ta butun son, Musbat sonlar soni, nollar soni, manfiy sonlar soni.

Yechish g'oyasi

Birinchi navbatda nechtaligi noma'lum bo'lgan sonlar ketma-ketligini qanday o'qib olish mumkin, degan savol tug'iladi. Buning uchun

```
while (cin >> a)
```

operatoridan foydalangan holda o'qib olish zarur.

So'ngra masalani yechish uchun 3 ta o'zgaruvchi olamiz. Bu o'zgaruvchilar bizga hisoblagich vazifasini bajaradi. Biz har bir kiritilgan sonni tekshirib uning musbat, nol va manfiy qay biriga kirishiga qarab o'sha hisoblagichni o'zgartirib boramiz.

```
if (a > 0) musbat ++; else
```

```
if (a == 0) nol ++; else manfiy ++;
```

Shu tariqa hamma sonlarni kiritilgan paytdayoq tekshirsak bizga sonlarni saqlashga massiv ham kerak bo'lmaydi.

C1D. Girls and Boys

TATUda n ta guruh bor. Har bir i -guruhda $q[i]$ ta qiz bola va $b[i]$ ta o'gil bola o'qiydi. Sizga vazifa qilib o'gil bolalar ko'pmi yoki qiz bolalar ko'pligini aniqlash vazifasi qo'yilgan.

Kiruvchi ma'lumotlar:

Birinchi qatorda guruhlar soni n ($0 < n \leq 1000$) berilgan. Qolgan n ta satrda ikkitadan son bilan i -guruhdagi qiz bolalar $q[i]$ va o'gil bolalar $b[i]$ soni ($0 \leq a[i], b[i] \leq 1000$) berilgan.

Chiquvchi ma'lumotlar:

Agar universitetda qiz bolalar ko'p bo'lsa "Girls" va nechta ko'p ekanligini probel bilan, agar universitetda o'g'il bolalar ko'p bo'lsa "Boys" va nechta ko'p ekanligini probel bilan, agarda ular soni teng bo'lsa "Equal" so'zini chiqaring.

Yechish g'oyasi

Bu masalada bizga n ta juftlik sonlar beriladi. Biz ularni kiritilgan paytning o'zida ikkita hisoblagichga qo'shib borsak bo'ldi.

```
cin >> x >> y;
```

```
Girls += x;
```

```
Boys += y;
```

Hamma juftliklar tugaganidan so'ng ularning qay biri ko'pligiga qarab javobni chiqaramiz.

C1E. Kun va oy

Yil ($1 \leq m \leq 9999$) va kun ($1 \leq n \leq 365$) raqamlari berilgan. Siz m-yildagi n-kunning qaysi oyga to'g'ri kelishini topishingiz kk. Masalan, 2013 yilning 10 dekabri 2013 yilning 344 kuniga to'g'ri keladi. Sizga 2013 344 sonlari berilsa, 10 December chiqarishingiz kerak. Oy kunlari: January, February, March, April, May, June, July, August, September, October, November va December. Kabisa yili deb 400 ga bo'linadigan yoki 4 ga bo'linib 100 bo'linmaydigan yillar kiradi.

Kiruvchi ma'lumotlar:

Birinchi satrda yil m ($1 \leq m \leq 9999$) va kun n ($1 \leq n \leq 365$) sonlari probel bilan berilgan.

Chiquvchi ma'lumotlar:

Berilgan sonlar uchun kun va oyni probel bilan chqaring. Katta va kichik harflarga e'tibor bering.

Yechish g'oyasi

Biz birinchi navbatda quyidagi massivlarni hosil qilib olishimiz kerak.

```
int a[]={0,31,28,31,30,31,30,31,31,30,31,30,31};  
string s[]={ "", "January", "February", "March",  
"April", "May", "June", "July", "August",  
"September", "October", "November",  
"December"};
```

Bu yerda a massivimiz mos oylardagi kunlar sonini, s massiv esa mos oylarning nomlari saqlaydi. Agar berilgan yilimiz kabisa yili bo'lsa $a[2] = 29$ ga tenglashtirib qo'yamiz.

Bu masalani yechishda Yanvar oyidan boshlab tekshirishni boshlaymiz. Agar yanvar oyidagi kunlar berilgan kundan kam bo'lsa biz berilgan kundan yanvar oyidagi kunlar sonini ayirib tashlab oyni keyingi oyga ya'ni fevralga o'tkazib yana tekshiramiz. Bu tekshirish Oydagi kunlar soni bizga berilgan kun sonida katta yoki teng bo'lgunicha davom etadi. Biz qaysi oyda ekanligini shu usul bilan aniqlab olamiz va shu oyni va berilgan kunni chiqarib yuborsak bo'ldi.

Masalan, 204-yildagi 60-kun qaysi sanaga to'g'ri kelishini aniqlash uchun yanvar oyidagi kunlar bilan solishtiramiz. Yanvar oyida 31 kun mavjud demak bizga berilgan kun yanvar oyida emas, undan yanvardagi kunlar sonini ayirib tashlaymiz, ya'ni berilgan kunni $60 - 31 = 29$ ga aylantirib olamiz. Endi Fevral oyidagi kunlar soni 204-yilda 29 kunga teng. Biz solishtirib ko'rsak bizda qolgan kun va oydagi kunlar soni teng, demak, bundan ko'rinib turibdiki 204-yilning 60-kuni 29-fevralga to'g'ri kelar ekan.

C1F. Kublar yig'indisi

N natural soni berilgan. Shu sonni 3 ta nomanfiy sonning kublari yig'indisi shaklida ifodalash mumkin yoki mumkin emasligini aniqlovchi dastur tuzing.

Kiruvchi ma'lumotlar:

N ($0 \leq N < 10000000$) natural soni berilgan.

Chiquvchi ma'lumotlar:

Agar berilgan sonni 3 ta nomanfiy sonning kublari yig'indisi shaklida ifodalash mumkin bo'lsa "YES" aks holda "NO" chiqaring.

1-test 277684 = 11*11*11 + 12*12*12 + 65*65*65;

2-test ifodalab bo'lmaydi shuning uchun NO;

3-test $8 = 2 * 2 * 2 + 0 * 0 * 0 + 0 * 0 * 0$

Yechish g'oyasi

Bu masalani yechish uchun biz [perebordan](#) foydalanamiz.

Perebor o'zi nima?

Perebor bu barcha holatlarni ko'rib chiqishdan iborat. Bu masalamiz uchun perebor:

$$0^3 + 0^3 + 0^3$$

$$0^3 + 0^3 + 1^3$$

$$0^3 + 0^3 + 2^3$$

$$0^3 + 0^3 + k^3$$

$$0^3 + 1^3 + 1^3$$

$$0^3 + 1^3 + 2^3$$

$$0^3 + 1^3 + k^3$$

$$0^3 + 2^3 + 2^3$$

$$0^3 + 2^3 + 3^3$$

$$0^3 + k^3 + k^3$$

$$1^3 + 1^3 + 1^3$$

$$1^3 + 1^3 + 2^3$$

$$k^3 + k^3 + k^3$$

Bu yerda k sonining kubi berilgan sonimizdan kichik va eng yaqin bo'lgan sonidir.

```
for(i = 0; i <= k; i ++){
    for(j = i; j <= k; j ++){
        for(q = j; q <= k; q ++){
            if(i*i*i + j*j*j + q*q*q == n)
                ***
        }
    }
}
```