



Департамент Информатика
Школа „Състезателно програмиране“
СЪСТЕЗАНИЕ, 21 февруари 2015 г.

А. Повтарящи се числа

Дадени са N ($1 \leq N \leq 1000$) цели числа, между 1 и 1000 включително. Интересуваме се от това, което се повтаря най-много пъти, измежду тях. Ако тези числа са няколко, интересуваме се от най-малкото, измежду всички повтарящи се.

На първия ред на входа е зададен броя на тестовите примери (не повече от 100). Всеки от тях започва с числото N . Следват N реда, на всеки от които е дадено по едно число.

За всеки пример извеждайте на отделен ред търсеното число.

Вход	Изход
3	42
3	7
42	11
42	
19	
4	
7	
99	
99	
7	
5	
11	
12	
13	
14	
15	



Департамент Информатика
Школа „Състезателно програмиране“
СЪСТЕЗАНИЕ, 21 февруари 2015 г.

В. Растящи числа

Растящо число наричаме, такова на което всяко следваща цифра в десетичния му запис е по-голяма или равна на предишната. Такова число е 123, докато 101 и 111100001111 не са.

На първия ред на стандартния вход ще бъде зададен броя на тестовите примери. За всеки тест на отделен ред ще е зададено едно цяло, положително число, с не повече от 80 цифри. Ако това число е растящо, то интересуваме се от броя на растящите числа по-малки от него, ако не е извеждайте -1. Гарантирано е, че изхода винаги се побира в 64 битово цяло число, без знак.

Вход	Изход
5	10
11	65
123	-1
101	220
1111	2001
99999	



Департамент Информатика
Школа „Състезателно програмиране”
СЪСТЕЗАНИЕ, 21 февруари 2015 г.

С. Израз

Даден ви е следния математически израз:

[number][op][number]=[number]

Всеки **[number]** съдържа една или повече цифри (0–9), евентуално знак минус от пред и нула или няколко въпросителни знака. Никое от тези **[number]** няма да съдържа водещи нули, освен ако самото то не е 0 и ще бъде в интервала – 999999 и 999999. Няма да има -0 и всеки **[number]** ще съдържа най-много 7 символа. **[op]** ще разделя два **[number]**-а и може да бъде знак за събиране (+), изваждане (-) или умножение (*). Между втория и третия **[number]** винаги ще има знак за равенство (=). В дадения израз няма да има никакви разстояния, табулации или каквито и де е други символи, освен описаните. Във всеки израз ще има няколко (поне едно) '?', на чиито места би трябвало да седи *една и съща* цифра, такава че да направи равенството вярно. Ако има повече от една възможна цифра, то интересуваме се от *най-малката такава*.

Броя на тестовите примери ще бъде зададен на първия ред на стандартния вход. Всеки тест ще бъде зададен на отделен ред.

За всеки тест извеждайте на отделен ред на стандартния изход едно число – възможно най-малкото такава, че замествайки го с въпросителните знаци равенството да бъде вярно. Ако такава число не съществува, извеждайте -1.

Вход	Изход
5	2
1+1=?	6
123*45?=5?088	0
-5?* -1=5?	-1
19--45=5?	5
??*??=302?	



Департамент Информатика
Школа „Състезателно програмиране”
СЪСТЕЗАНИЕ, 21 февруари 2015 г.

D. Планети

Зададени са ви p ($1 \leq p \leq 60$) планети, с имената и целочислени си координати. Имената им могат да съдържат само латински букви и цифри, винаги започват с буква и са не по-дълги от 50 символа. Имайте в предвид, че **E**arth и **e**arth са различни планети. Според данните, с които разполагате между w ($0 \leq w \leq 40$) от тях разстоянието е нула. Разстоянието между останалите планети е равно на Евклидовото разстояние. По зададени q ($1 \leq q \leq 60$) на брой заявки, всяка съдържаща по две планети, вие трябва да отговорите колко е най-късото разстояние между тях.

На първия ред на входа ще бъде зададен броя на тестовите примери. Всеки от тях започва с p . Следват p реда съдържащи името на поредната планета, както и нейните x y z координати ($0 \leq x, y, z \leq 2 \cdot 10^6$). На следващия ред е дадено w . На следващите w реда ще има по два низа a и b – указващи **еднопосочен** път от a към b , с дължина нула между тях. Следва ред с q , а следващите q реда съдържат по два низа p_1 и p_2 . За всяка заявка извеждайте най-късото разстояние между двете планети, закръглено към най-близкото цяло число. Спазвайте стриктно изходния формат от примера:

Примерен вход:	<i>... продължение на входа ...</i>
3	z2 z1
4	z1 z2
Earth 0 0 0	z1 z3
Proxima 5 0 0	2
Barnards 5 5 0	Mars 12345 98765 87654
Sirius 0 5 0	Jupiter 45678 65432 11111
2	0
Earth Barnards	1
Barnards Sirius	Mars Jupiter
6	
Earth Proxima	Примерен изход:
Earth Barnards	Case 1:
Earth Sirius	The distance from Earth to Proxima is 5 parsecs.
Proxima Earth	The distance from Earth to Barnards is 0 parsecs.
Barnards Earth	The distance from Earth to Sirius is 0 parsecs.
Sirius Earth	The distance from Proxima to Earth is 5 parsecs.
3	The distance from Barnards to Earth is 5 parsecs.
z1 0 0 0	The distance from Sirius to Earth is 5 parsecs.
z2 10 10 10	Case 2:
z3 10 0 0	The distance from z2 to z1 is 17 parsecs.
1	The distance from z1 to z2 is 0 parsecs.
z1 z2	The distance from z1 to z3 is 10 parsecs.
3	Case 3:
<i>входа продължава...</i>	The distance from Mars to Jupiter is 89894 parsecs.



Департамент Информатика
Школа „Състезателно програмиране”
СЪСТЕЗАНИЕ, 21 февруари 2015 г.

Е. Лицеви опори

Панко е голям футболен маниак, но в днешния мач е наказан от треньора да не участва в мача, т.е. ще гледа от скамейката. Ще гледа ли? Не само че няма да участва, но няма и да гледа, а ще трябва прави лицеви опори, когато неговия отбор вкара гол! Треньора явно доста му е набрал, тъй като му е замислил следната схема: всеки път когато отбора вкара гол, Панко ще трябва да прави лицеви опори, чиито брой е равен на разстоянието от което е вкаран поредния гол, и това с натрупване. Пример: ако първия вкаран гол е от 7 метра, то Панко прави 7 лицеви опори. Ако втория гол е вкаран от 3 метра, то Панко прави $7 + 3 = 10$ лицеви опори. При вкаран трети гол от 2 метра, той ще трябва да направи 12 лицеви опори. И така в крайна сметка Панко е направил $7 + 10 + 12 = 29$ лицеви опори за целия мач.

След мача той има среща с приятелката си, но от тези лицеви опори е тотално изтрецял и единственото, което помни е, че по време на мача е направил n лицеви опори. Приятелката му пък се опитва да познае сумата на разстоянията, от които са вкарвани головете. Например ако Панко е направил 29 лицеви опори, то би могло отборът да е вкарвал голове първо от 3 метра, после от 2, пак от 2 и от 7 метра, при което търсената сума $3 + 2 + 2 + 7 = 14$. Разбира се, такива суми може да има повече от една. При това положение се интересуваме от най-голямата възможна.

Входа започва с броя на тестовите примери. Всеки от тях съдържа N и M ($1 \leq N \leq 5000$, $1 \leq M \leq 10$). N е броя на лицевите опори, които е направил Панко. На следващия ред са зададени M различни, цели положителни числа S ($1 \leq S_i \leq 20$) – дължините (в метри), от които са вкарвани голове по време на мача.

За всеки тест извеждайте на отделен ред търсената сума. Ако със зададените дължини не е възможно Панко да е направил точно N лицеви опори, то извеждайте -1.

Вход	Изход
5	14
29 3	5
7 3 2	-1
15 1	3
1	14
16 1	
1	
6 2	
3 1	
29 3	
3 7 2	



Департамент Информатика
Школа „Състезателно програмиране”
СЪСТЕЗАНИЕ, 21 февруари 2015 г.

Г. Пътуване до луната

Страните-членки на ООН планират да изпратят 2-ма души до Луната, но има един проблем. В съответствие със своите принципи на глобалното единство, те искат двойката астронавти да са от две различни страни.

Отговорните за избора на екип разполагат с N ($1 \leq N \leq 10^5$) обучени астронавти, номерирани от 0 до $N - 1$. Проблема е, че не разполагат с информация за гражданството на всеки един астронавт. Единствената информация която имат те е, че някои конкретни двойки астронавти принадлежат към една и съща страна.

Вашата задача е да изчислите по колко различни начина могат да се изберат двойка астронавти, които не принадлежат към една и съща страна. Като информация ще получите достатъчно двойки астронавти, за да идентифицирате принадлежността на всеки един от тях. Например, ако астронавти 1, 2 и 3 са от една и съща страна, то достатъчно е да се спомене, че (1, 2) и (2, 3) са двойки астронавти от една и съща страна, и не е нужно да се споменава информация за трета двойка (1, 3).

Всеки тестов пример започва с N и P ($1 \leq P \leq 10^4$). Следват P реда, с двойки числа A и B ($0 \leq A, B \leq N - 1$), показващи че A и B са от една страна. Края на входа е маркиран с две нули.

За всеки тест извеждайте търсения брой на отделен ред.

Вход	Изход
4 2	4
0 1	23
2 3	
10 7	
0 2	
1 8	
1 4	
2 8	
2 6	
3 5	
6 9	
0 0	

Пояснение към първия пример: Астронавти 0 и 1 са от една страна, 2 и 3 също, но от друга страна. Следователно възможните двойки, които могат да бъдат сформирани са (0, 2), (0, 3), (1, 2) и (1, 3).



Департамент Информатика
Школа „Състезателно програмиране”
СЪСТЕЗАНИЕ, 21 февруари 2015 г.

Г. Точки

Върху права са дадени отсечки и точки, определени с техните координати.

Напишете програма, която за всяка точка намира в колко от дадените отсечки се съдържа. Ако левият и десният край на i -тата отсечката са съответно L_i и R_i , то точката с координата X се съдържа в тази отсечка когато $L_i \leq X \leq R_i$.

На първия ред на стандартния вход е записан броят на тестовите примери. На първия ред на всеки от тях са зададени числата N и M – съответно броя на отсечките и точките. Следват N реда с по две числа – координатите на краищата на всяка отсечка. На последния ред са зададени M числа – координатите на дадените точки. Възможно е двата края на една отсечка да съвпадат.

За всеки тест на отделен ред на стандартния изход програмата трябва да извежда за всяка точка (по реда от входа) в колко отсечки се съдържа.

Ограничения: $1 \leq N, M \leq 10^5$, координатите са цели числа в интервала $[-10^9; 10^9]$.

Вход	Изход
1	2 0 3 1
3 4	
2 5	
7 3	
5 10	
4 11 5 8	



Департамент Информатика
Школа „Състезателно програмиране“
СЪСТЕЗАНИЕ, 21 февруари 2015 г.

Н. Отсечки

Върху числовата ос последователно боядисваме n отсечки, всяка с крайни точки целите числа a_i и b_i , $i = 1, 2, \dots, n$. Някои от поредно боядисаните отсечки може да припокриват частично или изцяло предишно боядисани отсечки. На края на процеса на боядисването се оформят определен брой непресичащи се отсечки. Напишете програма, която намира този брой и извежда дължината на най-дългата от получените отсечки.

Вход

Всеки тестов пример започва с n . Следващите n реда съдържат двойки координати на отсечките, които боядисваме: $a_1, b_1, a_2, b_2, \dots, a_n, b_n$. Края на входа е маркиран с -1.

Изход

Две цели числа (на отделен ред за всеки тест, разделени с един интервал), които са съответно равни на броя на получените отсечки и на дължината на най-голямата от тях.

Ограничения: $0 < n < 1\,000$; $0 < a_i < b_i < 10\,000$ за $i = 1, \dots, n$.

Вход	Изход
6 11 12 1 3 6 7 2 6 12 13 11 12 -1	2 6



Департамент Информатика
Школа „Състезателно програмиране”
СЪСТЕЗАНИЕ, 21 февруари 2015 г.

I. Число

Една сутрин, съвсем случайно, докато Панко се шляел из улиците на София, попаднал на цялото положително число N . Тъй като Панко обожава числото 30, той иска да знае най-голямото кратно на 30, което може да бъде получено чрез разместване на цифрите на N . Подритвал числото, побутвал го, и какво ли още не, но нещо все му убягвало, отново...

За всеки тестов пример N ще бъде зададено на отделен ред и броят на цифрите му няма да надвишава 10^5 .

За всеки тест извеждайте на отделен ред търсеното число или -1, ако такова не съществува.

Вход	Изход
30	30
102	210
2931	-1



Департамент Информатика
Школа „Състезателно програмиране”
СЪСТЕЗАНИЕ, 21 февруари 2015 г.

Ж. Инвертиране

Напишете програма, която въвежда цялото положително число n ($0 \leq n < 2^{32}$) и извежда n' , което се получава от n , като всички 0-ли в двоичния запис биват заменени с 1-ци и обратното – всички 1-ци в 0-ли, т.е. n се инвертира.

Вход

От стандартния вход се въвежда броя на тестовите примери. Всеки от тях е зададен на отделен ред и съдържа n .

Изход

За всеки тестов пример на стандартния изход да се изведе n' .

Вход	Изход
3	2147483648
2147483647	4294967294
1	4294967295
0	