

Нов български университет
Департамент Информатика
Школа „Състезателно програмиране“
СЪСТЕЗАНИЕ, 19 ноември 2011 г.

Задача А. Прости числа

Простите числа представляват интерес за много хора, от много дълго време. Освен това се появяват сравнително често на състезания по програмиране. Именно за това в настоящата задача ще трябва да напишете програма, която да отговаря на серия от различни типове заявки върху простите числа в интервала $[2, 10^8]$. Тези, които не са запознати с термините просто число и прост делител, бих посъветвал да преминат към друга задача, а останалите да разгледат следващата таблица, описваща типовете заявки и очаквания отговор за всяка от тях:

Тип заявка	Очакван отговор
A a b	Броя на простите числа в интервала $[a, b]$.
B k	Най-малкият прост делител на k.
C k	1 - ако k е просто число и 0 в противен случай.
D k	1 - ако k обърнато на обратно (след премахването на водещите нули, ако има такива) е просто число и 0 - в противен случай.
E k	Най-близкото просто число до k. Ако две числа са еднакво близо до k, да се изведат и двете в нарастващ ред.
F k	Броят на простите числа, по-малки от k.
quit	Прекратяване изпълнението на програмата.

Първият символ за всяка от командите може да е малка или главна латинска буква.

Пример:

Вход	Изход
A 3 7	3
B 176	2
C 95	0
D 11	1
E 5	5
F 20	8
E 6	5 7
quit	

Нов български университет
Департамент Информатика
Школа „Състезателно програмиране“
СЪСТЕЗАНИЕ, 19 ноември 2011 г.

Задача В. Най-близките две точки

В равнината са дадени N ($2 < N \leq 25\,000$) точки с целочислени координати в интервала $[-10\,000; 10\,000]$. Да се намери разстоянието между най-близките две измежду тях.

Входът се състои от множество тестови примери, като всеки нов ред от поредния тестов пример съдържа по две числа – координатите на поредната точка (x, y) . За край на поредния тестов пример ще считаме ред, в който има само едно число - числото -1. Това, разбира се означава, че измежду зададените ви точки няма да има нито една с x координата равна на -1.

За всеки тестов пример, трябва да изведете търсеното минимално разстояние с точност 5 знака след десетичната точка. Броят на редовете в изхода трябва да бъде равен на броя на тестовите примери.

Пример:

Вход	Изход
0 1	3.60555
2 4	6.40312
6 8	
7 0	
-1	
7 9	
12 5	
90 3	
-1	

Нов български университет
Департамент Информатика
Школа „Състезателно програмиране“
СЪСТЕЗАНИЕ, 19 ноември 2011 г.

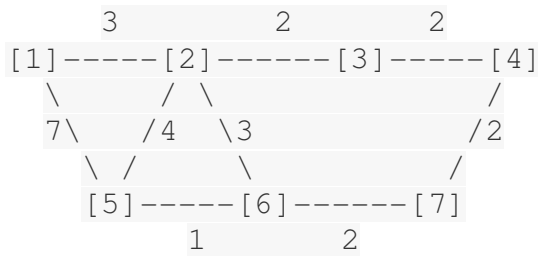
Задача С. Опознай Родината си!

В България има много прекрасни места, които си заслужава да бъдат посетени. Двама приятели програмисти много обичали да пътуват из България. Те били видни участници в движението „Опознай Родината за да я обикнеш“, обикаляли къде ли не – то не са пещери, планини, ждрела, язовири, манастири, музеи, морето... В началото (преди около 3-4 години) помежду другото събирали и печати от 100-те Национални туристически обекта. По едно време, това обаче се превърнало в нещо като мания. За ден-два те минавали по 400-500 километра и събирали по 10-тина печата. Единият от тях бил особено вманиачен и правел какви ли не планове за това от къде да мине, че да съберат колкото се може повече печати, за времето с което разполагат, а цялото това начинание получило подобаващото си име „военен туризъм“. Понякога участвали на олимпиади по програмиране (които се провеждали далеч от София) и без значение от представянето им на олимпиадата те обирали всички печати в околността. Докато за пореден път пътували за Горно Нанадолнище, единият разглеждал текущия им план за действие и му хрумнала гениална идея и казал на другия: „Като си толкова печен програмист, защо всеки път смяташ разстоянията между обектите с bgmaps, а не си напишеш програма която да прави това?“. Другият отвърнал, че това е тривиална задача и не му се занимава с глупости и задал контра-въпрос: „А ти можеш ли да напишеш програма, която от зададена стартова позиция да ти каже кое е най-късото разстояние, което трябва да изминеш, за да посетиш други два обекта?“. „Естествено, че мога, ама и на мен не ми се занимава, дай да я дадем на тези от школата по състезателно програмиране да видим дали те могат ;-)" – отвърнал другия. И така:

Да приемем, че в България има P ($1 \leq P \leq 100000$) населени места. Директните двупосочни пътищата, които ги свързват са C ($1 \leq C \leq 200000$), като разстоянието между две различни населени места P_{1_i} ($1 \leq P_{1_i} \leq P$) и P_{2_i} ($1 \leq P_{2_i} \leq P$) е D_i , зададено в бройна система K_i ($2 \leq K_i \leq 36$). Сумата от всички пътища не надвишава 2000000000. Пътната мрежа е така устроена, че няма директен път, който да започва и да свършва в едно и също място. Известно е още че от всяко населено място може да се стигне до кое да е друго.

По зададена стартова позиция P_B ($1 \leq P_B \leq P$) трябва да се определи минималното разстояние, за да посетим останалите два обекта - PA_1 ($1 \leq PA_1 \leq P$) и PA_2 ($1 \leq PA_2 \leq P$). Разбира се тези три места са различни.

Да разгледаме следния пример:



PB = [5], PA1 = [1], PA2 = 4. Тогава искания отговор е 12, а пътя е 5 -> 6-> 7 -> 4* -> 3 -> 2 -> 1*. За простота разстоянията, както и резултата са в десетична бройна система.

На първия ред на **входа** е дадено T – броя на тестовите примери. Всеки тест започва с числата C, P, PB, PA1, PA2 и B. Първите пет вече ги разяснихме, а числото B, както може би се досещате е бройната система, в която трябва да отпечатате отговора за поредния тест. Редовете 2..C+1 съдържат по четири числа - P1_i, P2_i, K_i и D_i, със значения описани по-горе.

Изходът трябва да представлява T реда, всеки съдържащ търсеното разстояние в бройна система B_i.

Пример:

Вход	Изход
2	110
9 7 5 1 4 3	6b3
5 1 2 111	
6 7 3 2	
4 7 4 2	
5 6 5 1	
5 2 6 4	
4 3 7 2	
1 2 8 3	
3 2 9 2	
2 6 10 3	
15 10 1 9 5 13	
1 3 2 1100101100	
1 4 3 202022	
2 4 4 31301	
1 2 5 2123	
8 7 6 4435	
3 2 7 1055	
5 10 8 1463	
8 6 9 1568	
4 8 10 599	
2 6 11 5a9	
8 1 12 345	
4 3 13 5b2	
5 7 14 c	
3 5 15 85	
7 9 16 160	

Нов български университет
Департамент Информатика
Школа „Състезателно програмиране“
СЪСТЕЗАНИЕ, 19 ноември 2011 г.

Задача D. MODEX

Напишете програма, която по зададени цели положителни числа x , y и n , изчислява по ефективен начин $x^y \bmod n$.

Първият ред на **входа** започва с числото C , следват C реда съдържащи числата x , y и n , разделени с интервал. Ред $C + 1$ съдържа числото 0.

Може да разчитате, че $1 < x$, $n < 2^{15} = 32768$, и $0 < y < 2^{31} = 2147483648$.

Изходът трябва да съдържа C реда, като i -тия ред се намира положителното число z , такова че

$$z = x^y \bmod n$$

за зададените в i -тия тест x , y и n .

Пример:

Вход	Изход
2	3
2 3 5	11
2 2147483647 13	
0	

Нов български университет
Департамент Информатика
 Школа „Състезателно програмиране“
СЪСТЕЗАНИЕ, 19 ноември 2011 г.

Задача Е. Охлюв

Охлюв се намира на земята и иска да се изкачи до върха на дървен прът с височина V . За един ден той може да изкачи A метра нагоре, през нощта обаче, когато спи, се спуска B метра надолу.

Да се определи колко дни са необходими на охлюва, за да се изкачи от земята до върха на пръчката.

Входът се състои от няколко тестови примера, всеки от тях съдържащ три цели числа – A , B и V ($1 \leq B < A \leq V \leq 1\,000\,000\,000$). Числата от всеки тестов пример, както и самите примери може да бъдат разделени с повече от един интервал и/или нови редове.

За всеки тестов пример трябва да изведете по едно число – търсеният брой дни, на нов ред.

Пример:

Вход	Изход
2 5 5 1 6	4 2 999999901
100 99 1000000000	

Нов български университет
Департамент Информатика
Школа „Състезателно програмиране“
СЪСТЕЗАНИЕ, 19 ноември 2011 г.

Задача G. Ограничени суми

За дадени цяли положителни числа p и s , напишете програма, която пресмята броя на всички различни редици от n неотрицателни цели числа, в които всеки елемент е по-малък от p и сумата на всички елементи е по-малка от s .

Първият ред на входа съдържа броя на тестовете. Данните за всеки тест са дадени на отделен ред в следната последователност: p , n и s .

Ограничения: $0 < p < s < 30$ и $0 < n < 20$.

Програмата трябва да изведе намерените стойности на съответни редове в изхода.

Пример:

Вход	Изход
1	7
2 3 3	190
5 4 7	

Пояснение за първия тест – редиците са:

0+0+0

0+0+1

0+1+0

1+0+0

0+1+1

1+1+0

1+0+1

Нов български университет
Департамент Информатика
Школа „Състезателно програмиране“
СЪСТЕЗАНИЕ, 19 ноември 2011 г.

Задача Н. Редица

Напишете програма, която въвежда цялото положително число n ($n < 50$) и извежда броя на редиците с дължина n , съставени от 0 и 1, в които няма три единици една до друга.

На първия ред във входа е записан броят на тестовете. Данните за всеки тест са записани на един ред. Съответните отговори трябва да се изведат на отделни редове.

Пример:

Вход	Изход
2	7
3	13
4	

Нов български университет
Департамент Информатика
Школа „Състезателно програмиране“
СЪСТЕЗАНИЕ, 19 ноември 2011 г.

Задача I. Цифри

Напишете програма, която въвежда низ S , съставен от ненулеви десетични цифри, т.е знаци от множеството {'1', '2', ... , '9'}. Дължината на низа не надминава 10^6 знака и е по-голяма от 2. Програмата трябва да изведе сумата от всички цифри във всички низове, които се получават от S чрез премахване на точно един знак.

На първия ред във входа е записан броят на тестовете. Данните за всеки тест са записани на един ред. Съответните отговорите трябва да се изведат на отделни редове.

Пример:

Вход	Изход
2	33
2234	30
1234	

Нов български университет
Департамент Информатика
 Школа „Състезателно програмиране“
СЪСТЕЗАНИЕ, 19 ноември 2011 г.

Задача J. Последния - първи

Дефинираме функцията $f(s)$ с аргумент - низ $s = c_1c_2...c_n$ по следния начин:

$$f(s) = c_n c_1 c_2 \dots c_{n-1}.$$

Дадено е множество S от m елемента - низове с дължина n , съдържащи само малки букви от латинската азбука. Прилагайки функцията f върху елементите на множеството $S_0 = S$, получаваме ново множество S_1 . Отново пресмятаме функцията f за елементите на S_1 и получаваме S_2 , и т.н. Задачата е да се намери k -тия по големина елемент ($0 < k \leq m$) в множеството S_i ($0 \leq i < n$), според лексикографската наредба за низове.

На стандартния вход се задава число - броят на елементите на множеството, след което на отделни редове са самите елементи на множеството. На следващия ред се намира число N - брой на множествата от редицата S_0, S_1, \dots, S_{m-1} , за които ще търсим k -ия по големина елемент. Следват $2N$ реда: за всяка двойка редове на първия ред са две числа - индекс на множество в редицата и броя на числата на втория ред. Там са дадени кои k -ти елементи от това множество да бъдат изведени на стандартния изход.

Ограничениата са: $n < 100, m < 10^5, N < 10^3$.

На стандартния изход се извеждат k -тите по големина елементи от съответните множества на отделни редове, като се оставя по един празен ред между отделните множества от редицата.

Пример:

Вход	Изход
3	aabb
abcd	bbad
aabb	
bbad	bbaa
2	
0 2	
1 3	
2 1	
2	

Сортираните множества, получени след прилагане на функцията върху елементите им са: $S_0 = \{\mathbf{aabb}, abcd, \mathbf{bbad}\}$, $S_1 = \{baab, dabc, dbba\}$, $S_2 = \{adbb, \mathbf{bbaa}, cdab\}$, $S_3 = \{abba, badb, bcda\}$. За 2 множества от редицата $\{S_0, S_1, S_2, S_3\}$ ще търсим k -ия по големина елемент. Първото множество е S_0 и извеждаме 2 елемента от него - първия и третия елемент. Второто множество е S_2 и от него трябва да изведем един елемент - втория.