

НОВ БЪЛГАРСКИ УНИВЕРСИТЕТ

Департамент *Информатика*

Школа *Състезателно програмиране*

СЪСТЕЗАНИЕ, 16 декември 2017 г.

Задача А. 1000001

Да се намери броят на всички естествени числа, които имат n нули и m единици в двоичния си запис.

Вход. За всеки тестов пример от стандартния вход се четат две цели числа – m и n , $1 < n \leq m < 32$

Изход. За всеки тестов пример на стандартния изход на нов ред се отпечатва резултата.

Пример:

Вход.

3 2
10 1

Изход.

10
11

НОВ БЪЛГАРСКИ УНИВЕРСИТЕТ

Департамент *Информатика*

Школа *Състезателно програмиране*

СЪСТЕЗАНИЕ, 16 декември 2017 г.

Задача В. Квадратна честотна функция

За даден низ, състоящ се от n малки латински букви, дефинираме квадратна честотна функция като сума от квадратите на броя на срещанията на различните букви в низа. Дефинираме и операция смяна на буква – заместване на всички срещания на една буква с друга буква.

Да се намери максималната стойност на квадратната честотна функция за низа, получен при прилагане на най-много k пъти на тази операция.

Вход. За всеки тестов пример на стандартния вход на един ред се задава низа и числото k .

Ограничения. $n, k < 100$

Изход. За всеки тестов пример на стандартния изход на отделен ред се отпечатва отговора.

Пример:

Вход.

alabala 1

Изход.

37

НОВ БЪЛГАРСКИ УНИВЕРСИТЕТ

Департамент *Информатика*

Школа *Състезателно програмиране*

СЪСТЕЗАНИЕ, 16 декември 2017 г.

Задача С. Прости дроби

Напишете програма, за събиране и изваждане на прости дроби, като представите резултата във вид на несъкратима дроб.

Стандартен вход: Всеки пример се задава със сума или разлика на няколко (най-малко 2 и не повече от 20) дроби и цели числа на отделен ред, като дробите са представени във вида m/n , където m и n са естествени числа. Всички числителни, знаменатели и цели числа са по-малки от 10001. Входът съдържа няколко примери.

Стандартен изход: За всеки пример на изхода се записва резултата като несъкратимата дроб по същия начин, както зададените на входа дроби. Когато решението е цяло число, то се записва по нормалния начин.

Пример:

Вход.

$$1/2 + 1/3$$

$$1/3 - 1/2 + 1$$

$$1/3 - 1/2 - 1/6$$

Изход.

$$5/6$$

$$5/6$$

$$-1/3$$

НОВ БЪЛГАРСКИ УНИВЕРСИТЕТ

Департамент *Информатика*
Школа *Състезателно програмиране*
СЪСТЕЗАНИЕ, 16 декември 2017 г.

Задача D. Последователни гласни

Намерете броя на низовете, съставени от малки латински букви с N символа, такива, че не съдържат повече от K последователни гласни. Гласни са латинските букви 'a', 'e', 'o', 'u', 'i', 'y'.

Вход. На първия ред на стандартния вход е зададен броят тестове T . Всеки от следващите T реда съдържа по две цели числа N и K – съответно дължината на низа, който трябва да се състави, и колко най-много последователни гласни може да има в него.

Изход. За всеки тест на отделен ред изведете по едно цяло число – броя възможни низове. Тъй като това число може да е много голямо, го изведете по модул 1000000009.

Ограничения. $1 \leq N, K \leq 1000$

Пример:

Вход.

```
3
3 1
5 2
666 42
```

Изход.

```
15920
11510720
88651987
```

Обяснение: В първия тест отговорът е броят на всички стрингове с дължина 3 (26^3) минус броят на стринговете, които имат поне две последователни гласни ($2 * (62 * 26) - 63$).

НОВ БЪЛГАРСКИ УНИВЕРСИТЕТ

Департамент *Информатика*
Школа *Състезателно програмиране*
СЪСТЕЗАНИЕ, 16 декември 2017 г.

Задача Е. Просто сортиране

Напишете програма, която сортира редица от цели положителни числа в нарастващ ред по броя на единиците в двоичното им представяне. Ако този брой е равен, то числата се сортират в намаляващ ред по тяхната стойност.

Вход. На всеки ред на стандартния вход е зададена редица от не повече от 10000 цели положителни числа, разделени с един или няколко интервала.

Изход. За всеки тест на отделен ред да се изведе редицата от числа, сортирана по искания начин. Числата трябва да са разделени с точно един интервал.

Пример:

Вход.

```
1 2 3 4 5 6 7
4 1 2 3 4 5 6 7
2 3 5 6 8
2 4 5
3 6 8 2
1 2
100
```

Изход.

```
4 2 1 6 5 3 7
4 4 2 1 6 5 3 7
8 2 6 5 3
4 2 5
8 2 6 3
2 1
100
```

Обяснение. Например 1 2 3 4 5 6 7 трябва да бъде сортирана по следния начин (в скоби е броят на единиците в двоичното представяне на числото): 4(1) 2(1) 1(1) 6(2) 5(2) 3(2) 7(3)

НОВ БЪЛГАРСКИ УНИВЕРСИТЕТ

Департамент *Информатика*
Школа *Състезателно програмиране*
СЪСТЕЗАНИЕ, 16 декември 2017 г.

Задача F. Остатъци

Дадени са четири естествени числа N , A , B и P . Можете да прилагате следното действие произволен брой пъти: вземате числото N и го заменят с едно от числата: $(N+A)\%P$ или $(N+B)\%P$. Колко операции най-малко ще са ви необходими, за да получите цялото неотрицателно число R ?

Вход. За всеки тест от първия ред на стандартния вход се въвеждат числата N и P . От следващия ред се задават числата A , B и R .

Ограничения. $0 < N \leq 10^9$ $0 < A, B \leq P \leq 10^6$ $0 \leq R < P$

Изход. За всеки тест на един ред на стандартния изход трябва да изведете числото S – минималния брой операции от гореописания вид, с които от N може да се получи R . Ако от N не може да се получи R чрез прилагане на описаните операции, програмата да изведе 1.

Пример:

Вход.

20 10

5 5 5

20 10

2 4 1

Изход.

1

-1

НОВ БЪЛГАРСКИ УНИВЕРСИТЕТ

Департамент *Информатика*
Школа *Състезателно програмиране*
СЪСТЕЗАНИЕ, 16 декември 2017 г.

Задача G. Игра

Двама играчи – Бял и Черен – играят игра на квадратна дъска с N реда и N стълба, като се редуват да правят ходове. Във всеки ред на дъската има по един бял и един черен пул. Белият играч има право да мести само белите пулове, а Черният – само черните. Пул може да се мести само по реда, в които се намира, в посока към пула на противника. Играчът, който е на ход, може да премести само един от своите пулове на всяко едно празно квадратче в реда, но няма право да застава върху другият пул или да го прескочи. Губи този играч, който е на ход и не може да премести пул. Напишете програма, която определя победителя в играта, ако се знае, че Белият играч е винаги първи на ход и двамата играчи играят оптимално.

Вход. На първия ред на стандартния вход ще бъде зададен броят на тестовите примери. Всеки тестов пример започва с ред, на който е зададен размерът N на игралната дъска ($2 \leq N \leq 1000$). Следват N реда, описващи началната ситуация на дъската. На i -тия от тези редове са зададени две числа – стълба, в който се намира белият и стълба, в който се намира черният пул на i -тия ред от дъската. Номерацията на редовете и стълбовете започва от 1.

Изход. За всеки тестов пример, в който Белият играч може да спечели, програмата трябва да изведе на един ред на стандартния изход думата WHITE, последвана от едно число – номера на реда от дъската, в който той трябва да направи своя печеливш ход. Думата и числото трябва да са разделени с един интервал. Ако съществуват няколко реда, в които Белият играч може да направи печеливш ход, програмата трябва да изведе този с минимален номер. Ако играта се печели от Черния играч, независимо какво играе Белият, програмата трябва да изведе за този тестов пример думата BLACK.

Пример:

Вход.

```
2
8
2 7
8 5
3 7
8 2
6 2
1 8
3 6
3 1
4
1 2
4 3
3 2
2 1
```

Изход.

```
WHITE 1
BLACK
```

НОВ БЪЛГАРСКИ УНИВЕРСИТЕТ

Департамент *Информатика*

Школа *Състезателно програмиране*

СЪСТЕЗАНИЕ, 16 декември 2017 г.

Задача Н. Брой на задрасквания

Дадени са два низа от малки латински букви. Задраскваме някои букви в низовете, но без да разместваме останалите така, че двата низа да станат еднакви. Напишете програма, която определя колко най-малко задрасквания трябва да направим.

Вход. На стандартния вход се въвеждат двойки низове на отделни редове. Дължината на дадените низове не надвишава 1000.

Изход. За всяка двойка на стандартния изход да се изведе броя на задраскванията.

Пример:

Вход.

abbcdddefef

bccdefababcabc

Изход.

14

НОВ БЪЛГАРСКИ УНИВЕРСИТЕТ

Департамент *Информатика*
Школа *Състезателно програмиране*
СЪСТЕЗАНИЕ, 16 декември 2017 г.

Задача I. Стоянчо овчарчето

Стоянчо овчарчето живеело на село и си гледало овчици. То имало N колиби и K овце. Всички колиби били построени на една линия и Стоянчо им знаел координатите, при това с много голяма точност. Колибите не били големи и затова най-много една овца можела да се побере в една колиба. По неясни причини овцете започнали да се карат една с друга. В следствие на това млекодобивът намалял. За да се справи с възникналата ситуация, Стоянчо решил когато прибира овцете в колибите им да ги подреди по такъв начин, че минималното разстояние между две овце да е максимално.

Вашата задача е да напишете програма, която ще помогне на Стоянчо да нареди овцете и те да се успокоят, та да започнат отново да дават мляко.

Вход. На първия ред на стандартния вход е зададен броят тестове във входа. За всеки тест са зададени две числа – N и K . N е броят на колибите, а K е броят на овцете. На втория ред във всеки тест има N числа, описващи координатите на всяка една от колибите. Тези стойности са цели числа в интервала $[1, 100000000]$. Известно е, че $1 \leq N \leq 100000$, $2 \leq K \leq N$.

Изход. Изходните данни се извеждат на стандартния изход. За всеки тест се извежда по един ред с едно число P – търсеното разстояние.

Пример:

Вход.

```
1
5 3
1
2
8
4
9
```

Изход.

```
3
```

НОВ БЪЛГАРСКИ УНИВЕРСИТЕТ

Департамент *Информатика*
Школа *Състезателно програмиране*
СЪСТЕЗАНИЕ, 16 декември 2017 г.

Задача J. Обиколка

Стоянка решила да си направи обиколка на България. Искала да посети определен брой градове, естествено всеки по веднъж, и накрая да се върне вкъщи. Тя преценила различните маршрути, по които може да направи това, и, разбира се, избрала най-късия. Вашата задача е по зададена карта на България с отбелязани на нея градове, които Стоянка иска да посети, да кажете каква е дължината на измисления от Стоянка път.

Вход. Данните се четат от стандартния вход. На първия ред е зададен броят на тестовете T . За всеки тест:

На първия ред стоят три числа: броят на градовете $2 \leq N \leq 20$, броят на пътищата между съседните градове $1 \leq M$ и номерът на града, в който се намира Митко $1 \leq S \leq N$ (приемаме, че градовете са номерирани с числата от 1 до N).

Следват M реда, като всеки от тях описва по една улица с три цели числа: $1 \leq X \leq N$, $1 \leq Y \leq N$ и $1 \leq Z \leq 1000$. Това значи, че има двупосочен път между X и Y , който е с дължина Z . Възможно е между някои двойки градове да има повече от една улица.

Изход. Резултатът се извежда на стандартния изход. Трябва да изведете T реда като на всеки стои по едно число – дължината на минималния път, който започва от S , минава през всички градове точно по веднъж и приключва в S . Ако няма път, отговарящ на това условие, изведете -1 .

Пример:

Вход.

```
3
2 2 2
1 2 1
1 2 5
4 6 1
1 2 1
2 3 2
2 4 2
1 3 1
1 4 4
4 3 3
3 2 1
1 2 4
2 3 7
```

Изход.

```
2
7
-1
```