Орієнтовні завдання І туру Всеукраїнської учнівської олімпіади з інформатики 2017-2018 навчального року

Завдання 1. «Сміх»

Степан любить багато сміятися. Сміх – це послідовність букв «a» і «h», які чергуються. Так наприклад, «ahahaha», «hah» і «a» є сміхом, а «abacaba» і «hh» - ні.

Степан розмовляє дуже швидко, тому всі його слова зливаються в одне велике. Для дослідження вам потрібно з'ясувати, як довго він може сміятися. У вас є рядок – запис розмови Степана. Визначте найбільшу довжину сміху в цій розмові.

Вхідні дані:

Перший рядок вхідного файлу містить одне натуральне число N ($1 \le N \le 10^5$) - довжина рядка з розмовою Степана. У другому рядку міститься рядок з маленьких латинських букв довжиною N - запис розмови Степана.

Вихідні дані:

У вихідний файл виведіть одне число - найбільшу довжину сміху в розмові Степана.

Система оцінювання:

В даній задачі кожен тест оцінюється окремо.

Приклади вхідних та вихідних даних:

laugh.in	**laugh.out**
5 ahaha	5
24 ahahrunawayahahsofasthah	4
10 ahahaahaha	5

Завдання 2. «Операції з дробами»

Дроби, як відомо, давня слабкість Степана. Ось і зараз він бере правильний нескоротний дріб a/b і виконує з ним наступні операції: до чисельника і знаменника дробу додає 1, а потім дріб скорочує до нескоротного.

Степана зацікавило питання, чи можна за допомогою таких операцій з дробу a/b отримати правильний дріб c/d?

Допоможіть Степану.

Вхідні дані:

Вхідний файл містить чотири числа a, b, c, d ($0 < a < b \le 10^5$, $0 < c < d \le 10^5$), числа a і b взаємно прості, c і d взаємно прості, $a/b \ne c/d$.

Вихідні дані:

Виведіть одне натуральне число – скільки описаних операцій потрібно зробити, щоб з дробу a/b отримати правильний дріб c/d. Якщо цього зробити не можливо, то виведіть 0.

Система оцінювання:

В даній задачі кожен тест оцінюється окремо.

Приклади вхідних та вихідних даних:

fractions.in	**fractions.out**

1 3 2 3	2
2 3 1 3	0

Завдання 3. «Максимальний добуток»

Степан останнім часом приділяв мало уваги програмуванню і як, результат, не здав залік. Тепер йому потрібно терміново вирішити наступну задачу:

Дано масив цілих чисел $A_1, A_2, ..., A_N$, абсолютна величина елементів якого не перевищує 2. Потрібно знайти такий непорожній підвідрізок $A_l, A_{l+1}, ..., A_r$ цього масиву $(1 \le l \le r \le N)$, що добуток чисел $A_l * A_{l+1} * ... * A_r$ є максимально можливим.

Звісно, Степан просить у вас допомоги у вирішенні даної задачі.

Вхідні дані:

В першому рядку вхідного файлу знаходиться число N $(1 \le N \le 200\ 000)$ — кількість елементів масиву. В другому рядку знаходиться N цілих чисел A_i $(-2 \le A_i \le 2)$ - елементи масиву.

Вихідні дані:

Єдиний рядок вихідного файлу має містити два числа l і r - знайдені границі оптимального відрізка $(1 \le l \le r \le N)$. Якщо існує декілька відповідей, виведіть будь-яку з них.

Система оцінювання:

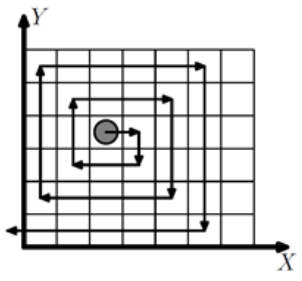
В даній задачі кожен тест оцінюється окремо.

Приклади вхідних та вихідних даних:

maximum.in	**maximum.out**
5 1 -1 2 2 1	3 5
3 -1 0 -2	2 2
7 -1 -2 -1 -2 1 2 -2	2 7

Завдання 4. «Дивний сон»

Степану сниться дивний сон. У ньому Степан знаходиться на полі в клітиночку розміром $N \times M$ в клітинці з координатами (x, y).



Спочатку Степан дивиться уздовж додатного напрямку осі X. Потім він починає йти по полю з наступною закономірністю:

- Пройти на одну клітинку вперед. Повернути на 90° вправо.
- Пройти на одну клітинку вперед. Повернути на 90° вправо.
- Пройти на дві клітинки вперед. Повернути на 90° вправо.
- Пройти на дві клітинки вперед. Повернути на 90° вправо.
- Пройти на три клітинки вперед. Повернути на 90° вправо.

- Пройти на три клітини вперед. Повернути на 90°вправо.

- Пройти на чотири клітини вперед. Повернути на 90°вправо.

- І так далі...

Рух триває до тих пір, поки Степан не вийде за межі поля. Після цього він прокидається. Вранці Степан вирішив проаналізувати свій сон. Він здогадався, що в кожній клітинці він був максимум один раз, але ніяк не може згадати, скільки клітинок він відвідав. Степан просить вас написати програму, яка порахує кількість відвіданих ним клітинок.

Вхідні дані: У першому рядку вхідного файлу знаходяться два натуральних числа $N, M (1 \le N, M \le 10^9)$ - розміри дошки уздовж осі X і осі Y відповідно. У другому рядку знаходяться два натуральних числа $x, y (1 \le x \le N; 1 \le y \le M)$ - координати стартової позиції Степана.

Вихідні дані:

У вихідний файл виведіть одне число - кількість клітинок, відвіданих Степаном уві сні.

Система оцінювання:

В даній задачі кожен тест оцінюється окремо.

Приклади вхідних та вихідних даних:

dream.in	**dream.out**
7 6 3 4	36
2 2 1 1	2
2 2 1 2	4

Завдання 5. «Вірус»

Степан дуже ретельно слідкує за своїм здоров'ям. Кожного дня він читає книги з медицини і шукає інформацію про нові хвороби, тому вже давно знає, що різні хвороби можуть збуджуватись вірусами. Степану давно відомі деякі види вірусів і він перевірив, що жодного з них у нього немає.

Але одного не найвдалішого для Степана дня він дізнався, що віруси можуть мутувати після того, як потрапили в організм людини. Мутація – це зміна ДНК таким чином, що в ньому змінюються рівно 2 символи, відстань між якими дорівнює k. ДНК віруса до потрапляння в організм людини представлено у вигляді рядка t, який складається з n маленьких літер латинського алфавіту.

Дізнавшись таку страшну новину, Степан одразу кинувся перевіряти, чи не хворіє він мутованим вірусом. ДНК Степана також є рядком з маленьких літер латинського алфавіту. Допоможіть йому дізнатись, чи є в його ДНК підрядки довжиною n, які підходять під опис мутованого вірусу, наведеного вище.

Вхідні дані:

В першому рядку вхідного файлу дано представлення ДНК Степана s - рядок з маленьких літер латинського алфавіту. У другому рядку задано представлення

ДНК віруса ***t*** - рядок з ***n*** маленьких літер латинського алфавіту. В третьому рядку задано число ***k (1 ≤ k ≤ n-1)***.

Вихідні дані:

У перший рядок вихідного файлу виведіть скільки разів мутивований вірус зустрічається в ДНК Степана. У другому рядку виведіть через пробіл в зростаючому порядку індекси початку входження мутивованого вірусу.

Система оцінювання:

В даній задачі три підзадачі. Бали за кожну підзадачу нараховуються тільки якщо усі тести підзадачі пройдені.

Підзадача 1 (30 балів):

ДНК Степана і віруса складаються не більш чим із 100 символів.

Підзадача 2 (30 балів):

ДНК Степана і віруса складаються не більш чим із 10 000 символів.

Підзадача 3 (40 балів):

ДНК Степана і віруса складаються не більш чим із 200 000 символів.

Приклади вхідних та вихідних даних:

virus.in	**virus.out**
abaaaaa baab 3	2 3 4
aaaaaaa aaaa 3	0