**І тур ІІІ етапу олімпіади з інформатики**

**2016-2017 н.р.**

**А. 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 ≤ N ≤ 105)*** - 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** |
| 5ahaha | 5 |
| 24ahahrunawayahahsofasthah | 4 |
| 10ahahaahaha | 5 |

**В. 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 ≤ 105, 0 < c < d ≤ 105)***, числа ***a*** і ***b*** взаємно прості, ***c*** і ***d*** взаємно прості, ***a/b ≠ 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 |

**C. 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 ***A1, A2, ..., AN***, абсолютна величина елементів якого не перевищує 2. Потрібно знайти такий непорожній підвідрізок ***Al, Al+1, ..., Ar*** цього масиву ***(1 ≤ l ≤ r ≤ N)***, що добуток чисел ***Al \* Al+1 \* ... \* Ar*** є максимально можливим.

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

**Вхідні дані:**

В першому рядку вхідного файлу знаходиться число ***N (1 ≤ N ≤ 200 000)*** — кiлькiсть елементів масиву. В другому рядку знаходиться ***N*** цiлих чисел ***Ai (-2 ≤ Ai ≤ 2)*** - елементи масиву.

**Вихідні дані:**

Єдиний рядок вихідного файлу має містити два числа ***l*** і ***r*** - знайдені границі оптимального відрізка ***(1 ≤ l ≤ r ≤ N)***. Якщо iснує декiлька вiдповiдей, виведiть будь-яку з них.

**Система оцінювання:**

В даній задачі кожен тест оцінюється окремо.

**Приклади вхідних та вихідних даних:**

|  |  |
| --- | --- |
| **maximum.in** | **maximum.out** |
| 51 -1 2 2 1 | 3 5 |
| 3-1 0 -2 | 2 2 |
| 7-1 -2 -1 -2 1 2 -2 | 2 7 |

**D. Дивний сон**

Степану сниться дивний сон. У ньому Степан знаходиться на полі в клітиночку розміром ***N х 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 ≤ N, M ≤ 109)*** - розміри дошки уздовж осі ***X*** і осі ***Y*** відповідно. У другому рядку знаходяться два натуральних числа ***x, y (1 ≤ x ≤ N; 1 ≤ y ≤ 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 63 4 | 36 |
| 2 21 1 | 2 |
| 2 21 2 | 4 |

**E. 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** |
| abaaaaabaab3 | 23 4 |
| aaaaaaaaaaa3 | 0 |

### ІІ тур

### А. Останнє число

Степан вирішив сьогодні поекспериментувати з послідовністю натуральних чисел від 1 до ***N***. Він спочатку викреслив усі непарні числа. Потім з тих, що залишились викреслив числа, які стоять на не парних місцях. Цю процедуру він повторював до тих пір, поки не залишилось тільки одно число.

Допоможіть Степану знати число яке залишилось.

**Вхідні дані:**

Єдиний рядок вхідного файлу містить одне число ***N (1 ≤ N ≤ 1018)***.

**Вихідні дані:**

Виведіть одне натуральне число - відповідь на задачу.

**Система оцінювання:**

В даній задачі кожен тест оцінюється окремо.

**Приклади вхідних та вихідних даних:**

|  |  |
| --- | --- |
| **number.in** | **number.out** |
| 2 | 2 |

**Степінь подібності**

Степан називає степенем подібності двох рядків з великих латинських літер кількість сусідніх пар елементів першого рядка, які зустрічаються в другому рядку.

Наприклад, нехай у нас є два рядки ABBACAB і BCABB. Перший рядок має наступні пари: AB, BB, BA, AC, CA, AB. Відповідно, з указаних пар першого рядка в другому зустрічаються чотири пари AB, BB, CA, AB.

Допоможіть Степану знайти степінь подібності двох заданих рядків.

**Вхідні дані:**

Вхідний файл містить два рядки з великих латинських літер, кожен рядок не порожній і його довжина не перевищує 105.

**Вихідні дані:**

Виведіть одне натуральне число - степінь подібності.

**Система оцінювання:**

В даній задачі кожен тест оцінюється окремо.

**Приклади вхідних та вихідних даних:**

|  |  |
| --- | --- |
| **degree.in** | **degree.out** |
| ABBACABBCABB | 4 |

### C. Цукерки

Степан дуже любить цукерки. Сьогодні він йде на побачення і хоче пригостити дівчину цукерками. Степан виклав в ряд ***N*** цукерок. У кожної цукерки є тип ***pi***. Степан планує вибрати послідовність цукерок, що йдуть підряд за однієї умови - в цій послідовності повинно бути рівно два різних типи цукерок. Степан просить вас дізнатися, яку максимальну кількість цукерок він може взяти, враховуючи умову.

**Вхідні дані:**

У першому рядку вхідного файлу знаходиться одне натуральне число ***N (1 ≤ N ≤ 106)*** - кількість цукерок.

У другому рядку знаходиться ***N*** цілих чисел ***pi (1 ≤ pi ≤ 109)*** - де ***рi*** - тип ***і***-ї цукерки.

**Вихідні дані:**

У першому рядку вихідного файлу виведіть ціле число - максимальну кількість цукерок, яку Степан зможе взяти на побачення.

**Система оцінювання:**

В даній задачі кожен тест оцінюється окремо.

**Пояснення до прикладів:**

У першому прикладі на столі лежить три типи цукерок 1, 2 і 3. Степан може взяти перші три цукерки з типами 3, 3, 1, а може взяти останні чотири цукерки 1, 2, 2, 1. Значить, максимальна кількість цукерок, яку він може взяти дорівнює чотирьом.

У другому прикладі існує всього один тип цукерок, значить Степан не зможе взяти жодної цукерки.

**Приклади вхідних та вихідних даних:**

|  |  |
| --- | --- |
| **candies.in** | **candies.out** |
| 63 3 1 1 2 2 1 | 4 |
| 21 1 | 0 |

### D. Нове захоплення Степана

Нове захоплення Степана - малювання. Він вирішив купити ***К*** наборів олівців. Кожен набір складається з одного або декількох олівців. Кожен олівець має додатну довжину, яка виражається цілим числом міліметрів.

У магазині продаються ***N*** наборів олівців. Після того, як Степан купить рівно ***К*** наборів, він прийде додому і складе всі олівці в одну коробку. Степан дуже зрадіє, якщо різниця в довжині між найбільшим і найменшим олівцями в цій коробці буде мінімальна.

Тому він просить вас допомогти йому: виберіть з ***N*** наборів олівців рівно ***К*** так, щоб різниця між максимальним і мінімальним серед всіх куплених олівців була якомога менша.

**Вхідні дані:** У першому рядку вхідного файлу знаходяться два натуральних числа ***N, K (1 ≤ N ≤ 105, 1 ≤ K ≤ N)*** - кількість наборів олівців, наявних в магазині, і кількість наборів, необхідних Степану.

У кожному з наступних ***N*** рядків знаходиться ***Ci (1 ≤ Ci ≤ 2\*105)*** - кількість олівців в наборі. Далі, в цьому ж рядку, слідують ***Ci*** натуральних чисел ***Aij (1 ≤ Aij ≤ 109)*** - довжини олівців в ***і***-му наборі.

Гарантується, що сума всіх ***Ci*** не перевищує 2\*105.

**Вихідні дані:**

У єдиному рядку вихідного файлу виведіть найменшу різницю між максимальним і мінімальним купленими олівцями, яку можна досягти.

**Система оцінювання:**

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

**Підзадача 1 (31 бал): *n ≤ 20, k > 1***

**Підзадача 2 (31 бал): *n ≤ 2000, p = 0***

**Підзадача 3 (38 балів):** Без додаткових обмежень.

**Приклади вхідних та вихідних даних:**

|  |  |
| --- | --- |
| **pencil.in** | **pencil.out** |
| 3 23 1 3 43 5 1 21 4 | 3 |
| 5 33 2 1 32 4 13 4 2 44 3 2 3 32 5 6 | 3 |

### E. Підводний човен

Підводний човен сів на мілину. Для його виявлення використовують дані супутника, який з високою точністю вимірює відхилення висоти поверхні води від середнього рівня моря. Знімок, отриманий із супутника, представляє собою масив з ***h*** рядків по ***w*** елементів у кожному рядку.



Введемо на знімку систему координат з віссю абсцис, яка напрямлена вздовж рядків знімку зліва направо, і віссю ординат, напрямленій вздовж стовпців знизу догори. Потенційне зображення підводного човна представляє собою множину елементів масиву, що складається з наступних частин:

- "корпус" - полоса з елементів з координатами від ***(x1, y1)*** до ***(x2, y1)***, де ***x1 < x2***;

- "рубка" - полоса з елементів з координатами від ***(x3, y1)*** до ***(x3, y2)***, де ***x1 ≤ x3 < x2, y1 ≤ y2***;

- "хвіст" - полоса з елементів з координатами від ***(x4, y3)*** до ***(x4, y4)***, де ***x3 < x4 ≤ x2, y3 ≤ y1 ≤ y4***.

Оскільки підводний човен знаходиться поблизу поверхні в районі з сильною течією, рівень води над ним трохи підвищується. Тому зображенням підводного човна на знімку будемо вважати зображення з максимально можливою сумою елементів масиву, що входять в нього.

Напишіть програму, яка знаходить на знімку зображення підводного човна і виводе суму його елементів.

**Вхідні дані:**

Для стиснення даних, що передаються з супутника, кожен елемент знімка кодується маленькою буквою англійського алфавіту. Перший рядок вхідних даних містить число ***k (k ≤ 26)*** - кількість використаних для кодування букв. Другий рядок вхідних даних містить ***k*** цілих чисел ***Ci*** - значення відхилень, які відповідають кожному кодовому символу по порядку букв в англійському алфавіті від 1 до ***k***-ї.

Третій рядок вхідних даних містить числа ***h*** i ***w*** - розміри знімка. Наступні ***h*** рядків містять по ***w*** символів - кодові значення елементів знімка.

**Вихідні дані:**

У вихідний файл виведіть одне число - суму елементів масиву, що відповідають зображенню підводного човна.

**Система оцінювання:**

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

**Підзадача 1 (32 бала): *5 ≤ h, w ≤ 10, |Ci| ≤ 10***

**Підзадача 2 (22 бала):** ***5 ≤ h, w ≤ 100, |Ci| ≤ 100***

**Підзадача 3 (23 бала): *5 ≤ h, w ≤ 500, |Ci| ≤ 500***

**Підзадача 3 (23 балів):** ***5 ≤ h, w ≤ 2000, |Ci| ≤ 2000***

***Пояснення:***

Для прикладу нижче наведено кілька потенційних зображень підводного човна.



Нижче наведено кілька множин елементів знімка, які не є потенційними зображеннями підводного човна:



**Приклади вхідних та вихідних даних:**

|  |  |  |
| --- | --- | --- |
| **submarine.in** | **submarine.out** | **Пояснення** |
| 2-10 16 11aaaaaaaaaaaaaabaaaaaaaaaabaaaabaaabbbbbbbbbaaaaaaaaabaaaaaaaaaaaaa | 13 | ..............b..........b....b...bbbbbbbbb.........b............. |
| 3-4 -3 45 5bbabcccaacaccbabaccbbaaaa | 16 | ......c....cc....c....... |
| 3-2 4 05 5abccbcccaccbcbacccbbaccba | 24 | .b....c....b.b.cccbb...b. |
| 4-1 -5 -3 05 5bbabcccaacacdbabaccbbaaaa | -2 | .......aa................ |