

## Числа Сміта (*Smith*)

Поняття числа Сміта було введено Альбертом Віланскі з Університету Лехай у 1982 році. Переглядаючи свою телефонну книжку, математик звернув увагу на те, що телефонний номер його зятя Гарольда Сміта (493-7775) має таку цікаву властивість, що сума його цифр дорівнювала сумі цифр усіх його простих співмножників.

Число 4937775 розкладається на прості співмножники таким чином:  $4937775=3 \times 5 \times 5 \times 65837$ . Сума цифр телефонного номера дорівнює  $4+9+3+7+7+7+5=42$  і сума цифр його розкладання на прості множники також дорівнює  $3+5+5+6+5+8+3+7=42$ . Віланскі назвав такий тип чисел на ім'я свого зятя. Оскільки таку властивість мають всі прості числа, Віланскі не включив їх до означення.

Допоможіть Альберту знайти ще числа Сміта. Для кожного числа із заданого набору знайдіть мінімальне число Сміта, що його перевищує.

### Формат введення-виведення:

Програма *Smith* зчитує з клавіатури (стандартного пристрою введення) натуральне число  $N$  ( $1 \leq N \leq 10$ ) – кількість чисел у наборі, а також  $N$  натуральних чисел  $a_i$  ( $1 \leq a_i \leq 2^{31}$ ).

Програма *Smith* виводить на екран (стандартний пристрій виведення)  $N$  знайдених чисел через пробіл.

### Приклад вхідних та вихідних даних

Введення	Виведення
2 4937774 234	4937775 265

## Числа Сміта (*Smith*)

Понятие числа Сміта было введено Альбертом Вилански из Университета Лехай в 1982 году. Просматривая свою телефонную книжку, математик обратил внимание на то, что телефонный номер его зятя Гарольда Сміта (493-7775) обладал тем интересным свойством, что сумма его цифр равнялась сумме цифр всех его простых сомножителей.

Число 4937775 раскладывается на простые сомножители следующим образом:  $4937775=3 \times 5 \times 5 \times 65837$ . Сумма цифр телефонного номера равна  $4+9+3+7+7+7+5=42$  и сумма цифр его разложения на простые множители также равна  $3+5+5+6+5+8+3+7=42$ . Вилански назвал такой тип чисел по имени своего зятя. Так как этим свойством обладают все простые числа, Вилански не включил их в определение.

Помогите Альберту найти еще числа Сміта. Для каждого числа из заданного набора найдите минимальное превышающее его число Сміта.

### Формат ввода-вывода:

Программа **Smith** читает с клавиатуры (стандартного устройства ввода) натуральное число  $N$  ( $1 \leq N \leq 10$ ) – количество чисел в наборе, а также  $N$  натуральных чисел  $a_i$  ( $1 \leq a_i \leq 2^{31}$ ).

Программа **Smith** выводит на экран (стандартное устройство вывода)  $N$  найденных чисел через пробел.

### Пример входных и выходных данных

Ввод	Вывод
2 4937774 234	4937775 265

### Обнулення масиву (Zeroing)

Є масив, що складається з  $N$  чисел. За один крок дозволяється зменшити на 1 кілька (можливо один) підряд рівних елементів. Мета – зробити всі елементи рівними нулю. За яку мінімальну кількість кроків це може бути зроблено?

### Формат введення-виведення:

Програма **Zeroing** зчитує з клавіатури (стандартного пристрою введення) натуральне число  $N$  ( $1 \leq N \leq 2 \cdot 10^5$ ) – кількість чисел у масиві, а з наступного рядка  $N$  невід'ємних цілих чисел, елементів масиву, кожне з яких не перевищує  $2 \cdot 10^9$ .

Програма **Zeroing** виводить на екран (стандартний пристрій виведення) єдине число – шукану кількість кроків.

### Приклад вхідних та вихідних даних

Введення 1	Виведення 1
3	4
3 4 1	
Введення 2	Виведення 2
3	6
3 1 4	

### Обнуление массива (Zeroing)

Имеется массив, состоящий из  $N$  чисел. За один шаг разрешается уменьшить на 1 несколько (возможно один) подряд идущих равных элементов. Цель – сделать все элементы равными нулю. За какое минимальное количество шагов это может быть сделано?

### Формат ввода-вывода:

Программа **Zeroing** читает с клавиатуры (стандартного устройства ввода) натуральное число  $N$  ( $1 \leq N \leq 2 \cdot 10^5$ ) – количество чисел в массиве, а со следующей

строки  $N$  неотрицательных целых чисел, элементов массива, каждое из которых не превышает  $2 \cdot 10^9$ .

Программа **Zeroing** выводит на экран (стандартное устройства вывода) единственное число – искомое количество шагов.

### Пример входных и выходных данных

**Ввод 1**                      **Вывод 1**

3                                4

3 4 1

**Ввод 2**                      **Вывод 2**

3                                6

3 1 4

### Кооперація (Coop)

Гноми-велетні знамениті своїми гігантськими розмірами та любов'ю до золота. В одному з поселень гномів існує  $N$  пар гномів, причому в  $i$ -ій ( $1 \leq i \leq N$ ) парі як чоловік-гном, так і жінка-гном мають зріст рівно  $i$ . Крім того, кремезні чоловіки-гноми зазвичай виступають у якості опори, у той час як тендітні жінки-гноми вправно орудують киркою.

Геологічна розвідка виявила неподалік селища гномів вертикальну нескінченно високу скелю. Гноми вирішили, що там з великою вірогідністю буде золото, тож домовилися вранці вирушити на полювання на нього. Щоправда, дійшовши до скелі, гноми виявили проблему:  $K$  сімей не прийшли на зустріч.

Гноми збираються довбати скелю в наступний спосіб: нехай якийсь чоловік-гном зросту  $A$  виступає опорою для якоїсь жінки-гнома зросту  $B$ . Разом цей тандем здатний довбати скелю **лише** на висоті  $A + B$ . Для кожної висоті обирається підходящий тандем, причому будь-який присутній гном може виступати (в різні моменти часу) у складі різних тандемів.

Тепер гномів цікавить, скільки ж існує досяжних висот. Іншими словами, для скількох висот знайдеться підходящий тандем?

### Формат введення-виведення:

Програма **Coop** спочатку зчитує з клавіатури (стандартного пристрою введення) два цілих числа  $N$  ( $1 \leq N \leq 2 \cdot 10^9$ ) та  $K$  ( $1 \leq K \leq \min\{N, 10000\}$ ) – кількість пар гномів у поселенні та кількість відсутніх на видобутку сімей гномів.

Потім зчитується рівно  $K$  різних невід'ємних цілих чисел – номери сімей, у довільному порядку, що не прийшли на видобуток. Кожне число не перевищує  $N$ .

Програма **Coop** виводить на клавіатуру (стандартний пристрій виведення) єдине ціле число – кількість досяжних висот.

## Система оцінки

30% балів припадає на тести, в яких  $N, K \leq \min\{N, 1000\}$ .

Ще 40% балів припадає на тести, в яких  $N \leq 10^9, K \leq \min\{N, 500\}$ .

### Приклад вхідних та вихідних даних:

#### Введення 1

6 2  
3 6

#### Виведення 1

9

#### Введення 2

5 3  
2 3 4

#### Виведення 2

3

#### Введення 3

10 4  
2 3 4 5

#### Виведення 3

15

## Кооперация (*Coop*)

Гномы-великаны знамениты своими гигантскими размерами и любовью к золоту. В одном из поселений гномов живут целых  $N$  пар гномов, причём в  $i$ -й ( $1 \leq i \leq N$ ) паре рост как мужа-гнома, так и жены-гнома равен  $i$ . Кроме того, крепкие мужчины-гномы обычно выступают в качестве опоры, в то время как хрупкие женщины-гномы ловко орудуют киркой.

Геологическая разведка обнаружила недалеко от поселения гномов вертикальную бесконечно высокую скалу. Гномы решили, что там с большой вероятностью будет золото, так что договорились утром отправиться на охоту за ним. Правда, дойдя до скалы, гномы обнаружили проблему:  $K$  семей не явились на встречу.

Гномы собираются долбить скалу таким образом: пусть некоторый мужчина-гном ростом  $A$  выступает опорой для некоторой женщины-гнома роста  $B$ . Вместе этот тандем способен долбить скалу **только** на высоте  $A + B$ . Для каждой высоты выбирается подходящий тандем, причём любой присутствующий гном может выступать (в разные моменты времени) в составе разных танDEMов.

Теперь гномов интересует, сколько же существует достижимых высот. Иными словами, для скольких высот найдётся подходящий тандем?

## Формат ввода-вывода

Программа *Coop* сначала считывает с клавиатуры (стандартного устройства ввода) два целых числа  $N$  ( $1 \leq N \leq 2 \cdot 10^9$ ) и  $K$  ( $1 \leq K \leq \min\{N, 10000\}$ ) – количество пар гномов в поселении и количество отсутствующих на добыче семей гномов.

Затем со следующей строки считывается ровно  $K$  разных неотрицательных целых чисел – номера семей в произвольном порядке, не пришедших на добычу. Каждое число не превышает  $N$ .

Программа **Соор** выводит на экран (стандартное устройство вывода) единственное целое число – количество достижимых высот.

### Система оценки

30% баллов приходится на тесты, в которых  $N, K \leq \min\{N, 1000\}$ .

Ещё 40% баллов приходится на тесты, в которых  $N \leq 10^9, K \leq \min\{N, 500\}$ .

### Пример входных и выходных данных

**Ввод 1**

6 2  
3 6

**Вывод 1**

9

**Ввод 2**

5 3  
2 3 4

**Вывод 2**

3

**Ввод 3**

10 4  
2 3 4 5

**Вывод 3**

15

### Багатокутники (*Numconv*)

Задана клітчаста сітка. Скільки різних опуклих багатокутників може бути на ній намальовано, якщо всі вершини повинні лежати у вузлах сітки, а сторони бути або горизонтальними, або вертикальними, або діагональними (під кутом 45 градусів)?

Шириною сітки назвемо кількість вузлів у кожному її ряду, а висотою – кількість вузлів у кожному її стовпчику.

Багатокутники, які потрібно знайти, повинні мати такі властивості:

- їх вершини повинні лежати у вузлах решітки;
- всі сторони горизонтальні, вертикальні або діагональні (45 градусів);
- багатокутник опуклий.

Два багатокутники вважаються різними, якщо їх сторони не співпадають, тобто два однакових за формою багатокутника, що знаходяться у різних позиціях, слід вважати за два. Однак додавання вершини у середину ребра не змінює його форми і не утворює новий (для підрахунку) багатокутник.

За заданими шириною і висотою поля знайдіть кількість багатокутників.

**Формат введення-виведення:**

Програма **Numconv** зчитує з клавіатури (стандартного пристрою введення) два натуральні числа  $a$  та  $b$  ( $2 \leq a, b \leq 40$ ) – ширину та висоту сітки.

Програма **Numconv** виводить на екран (стандартний пристрій виведення) єдине число – кількість можливих багатокутників.

**Приклад вхідних та вихідних даних**

Введення 1	Виведення 1
------------	-------------

3 2	19
-----	----

Введення 2	Виведення 2
------------	-------------

2 2	5
-----	---

**Многоугольники (Numconv)**

Задана клетчатая сетка. Сколько разных выпуклых многоугольников может быть на ней нарисовано, если все вершины должны лежать в узлах сетки, а стороны быть или горизонтальными, или вертикальными, или диагональными (под углом 45 градусов)?

Шириной сетки назовем количество узлов в каждом ее ряду, а высотой – количество узлов в каждом ее столбике.

Многоугольники, которые надо найти, должны иметь такие свойства:

- их вершины должны лежать в узлах решетки;
- все стороны горизонтальные, вертикальные или диагональные (45 градусов);
- многоугольник выпуклый.

Два многоугольника считаются разными, если их стороны не совпадают, т.е. два одинаковых по форме многоугольника, которые находятся в разных позициях, надо считать за два. Однако добавление вершины в середину ребра не изменяет его формы и не создает новый (для подсчета) многоугольник.

За заданными шириною и высотой поля найдите количество многоугольников.

**Формат ввода-вывода:**

Программа **Numconv** читает с клавиатуры (стандартного устройства ввода) два натуральных числа  $a$  та  $b$  ( $2 \leq a, b \leq 40$ ) – ширину и высоту сетки.

Программа **Numconv** выводит на экран (стандартное устройство вывода) единственное число – количество возможных многоугольников.

## Пример входных и выходных данных

<b>Ввод 1</b>	<b>Вывод 1</b>
3 2	19
<b>Ввод 2</b>	<b>Вывод 2</b>
2 2	5

## Вікторина (Quiz)

Гноми-велетні вирішили взяти участь у чемпіонаті гри «Що? Де? Коли?». Чемпіонат складається з  $t$  турів. Кожен тур містить деяку кількість питань. Серія питань – це підпоследовательність питань одного туру від  $l$ -ого до  $r$ -ого включно, причому  $l \leq r$  і підпоследовательність неперервна (беруться до уваги всі підряд питання від  $l$ -ого до  $r$ -ого). Гноми будуть вважати серію питань вдалою, якщо вони дадуть відповіді не менш, ніж на  $q$  відсотків питань з цієї серії. Для підняття настрою команди, знайдіть для кожного туру кількість вдалих серій.

### Формат введення-виведення:

Програма *Quiz* зчитує з клавіатури (стандартного пристрою введення) число  $t$  – кількість турів у чемпіонаті,  $1 \leq t \leq 10$ . У наступних  $t$  рядках знаходиться інформація про кожен з турів у такому вигляді:  $n q a_1 \dots a_n$ ,  $0 \leq q \leq 100$ .  $a_i =$  може бути 0, якщо команда не відповіла на  $i$  питання, або ж 1 в іншому випадку. Крім того, усього на чемпіонаті було задано не більш, ніж 500000 питань. Усі числа вхідних даних цілі.

Програма *Quiz* виводить на екран (стандартний пристрій виведення) рівно  $t$  чисел в один рядок через пробіл:  $i$ -те число дорівнює кількості вдалих серій питань у  $i$ -ому турі.

### Система оцінки

70% балів припадатиме на тести, в яких  $q_i = 50$ .

### Приклад вхідних та вихідних даних

Введення	Виведення
2	
10 50 1 0 1 0 1 1 0 0 0 1	32 4
5 50 1 0 0 0 1	

## Викторина (Quiz)

Гномы-великаны решили принять участие в чемпионате игры «Что? Где? Когда?». Чемпионат состоит из  $t$  туров. Каждый тур содержит некоторое количество вопросов. Серия вопросов – это подпоследовательность вопросов одного тура от  $l$ -го по  $r$ -й включительно, причём  $l \leq r$  и подпоследовательность непрерывна (берутся во внимание все подряд вопросы от  $l$ -го по  $r$ -й). Гномы будут считать серию вопросов удачной, если они дадут

ответы не менее, чем на  $q$  процентов вопросов из этой серии. Для поднятия настроения команды, найдите для каждого тура количество удачных серий.

### Формат ввода-вывода:

Программа *Quiz* читает с клавиатуры (стандартного устройства ввода) число  $t$  – количество туров в чемпионате,  $1 \leq t \leq 10$ . В следующих  $t$  строках находится информация о каждом из туров в следующем виде:  $n \ q \ a_1 \dots a_n$ ,  $0 \leq q \leq 100$ .  $a_i =$  может быть 0, если команда не ответила на  $i$ -й вопрос, или же 1 в ином случае. Кроме того, всего на чемпионате было задано не более, чем 500000 вопросов. Все числа входных данных целые.

Программа *Quiz* выводит на экран (стандартное устройство вывода) ровно  $t$  чисел в одну строку через пробелы:  $i$ -е число равно количеству удачных серий вопросов  $i$ -го тура.

### Система оценки

70% баллов приходится на тесты, в которых  $q_i = 50$ .

### Пример входных и выходных данных

**Ввод**

2

```
10 50 1 0 1 0 1 1 0 0 0 1
5 50 1 0 0 0 1
```

**Вывод**

32 4