

# Задача А. Картахена

Назва вхідного файлу:	стандартний ввід
Назва вихідного файлу:	стандартний вивід
Обмеження за часом:	1 секунда
Обмеження за пам'яттю:	64 мегабайт

Женя любить грати у настільну гру «Картахена». Ціль гри полягає у тому, щоб раніше усіх провести своїх піратів з в'язниці-фортеці через підземний хід у порт, де знаходиться човен.

У цій задачі вам доведеться зіграти у спрощену версію гри. Уявімо підземний хід у вигляді ланцюга відсіків. Кожен відсік характеризується цілим числом від 1 до 6. На початку гри усі ваші пірати, а їх у вас рівно 2, знаходяться перед входом у підземний хід – перед першим відсіком. Човен знаходиться за останнім відсіком. Кожен свій хід ви обираєте будь-яке число від 1 до 6 та переміщуєте одного з двох піратів у бік човна у найближчий з відсіків з цим числом. Якщо даний відсік вже зайнятий другим піратом, то перший пірат автоматично переміщується у наступний за цим найближчий відсік з цим числом. У разі відсутності такого відсіку на шляху пірата, що залишився, пірат автоматично переміщується у човен. Ваша задача полягає у тому, щоб за найменшу кількість кроків перемістити обох піратів у човен.

## Формат вхідного файлу

У першому рядку вхідного файлу задано  $n$  ( $1 \leq n \leq 10^5$ ), у другому рядку задані числа  $a_i$  через пробіл, ( $1 \leq i \leq n, 1 \leq a_i \leq 6$ ).

## Формат вихідного файлу

У єдиному рядку вихідного файлу виведіть єдине число – мінімальну кількість кроків, за які можна перемістити обох піратів у човен.

## Приклад

стандартний ввід	стандартний вивід
12 1 2 3 4 5 6 1 2 3 4 5 6	4
11 6 5 4 3 2 1 6 5 4 3 2	3
1 4	2

# Задача В. Музична скринька

Назва вхідного файлу:	стандартний ввід
Назва вихідного файлу:	стандартний вивід
Обмеження за часом:	1 секунда
Обмеження за пам'яттю:	64 мегабайт

Одного разу археологи знайшли стару музичну скриньку. Вона була настільки стара, що її вирішили не запускати, бо боялись пошкодити механізм усередині. Натомість провели низку аналізів і виявили, що усередині скриньки знаходяться  $N$  музичних барабанів. Причому кожен з барабанів відіграє свою окрему музичну доріжку. Механізм влаштований так, що усі музичні доріжки програватимуться одночасно. Барабани, як виявилось, різних діаметрів, тому кожен з барабанів програватиме фрагмент певної власної довжини, що складає  $x_i$  секунд. Усі фрагменти програватимуться закріплено, тобто як тільки фрагмент завершується, він одразу починає програватись з самого початку. На початку пісні усі барабани починають програвати свої фрагменти з початків. Як було зазначено - пісня завершується тоді, коли настає її наступний початок, тобто у наступний момент усі фрагменти пісні будуть знову грати зі своїх початків одночасно. Науковці зацікавилися - а яка тривалість даної пісні?

## Формат вхідного файлу

У вхідному файлі у першому рядку вказане число  $N$  (ціле число,  $1 \leq N \leq 5$ ). У другому рядку вказано  $N$  чисел  $x_i$  (цілі числа,  $1 \leq x_i \leq 100$ )

## Формат вихідного файлу

У вихідному файлі необхідно вивести єдине число - тривалість пісні.

## Приклад

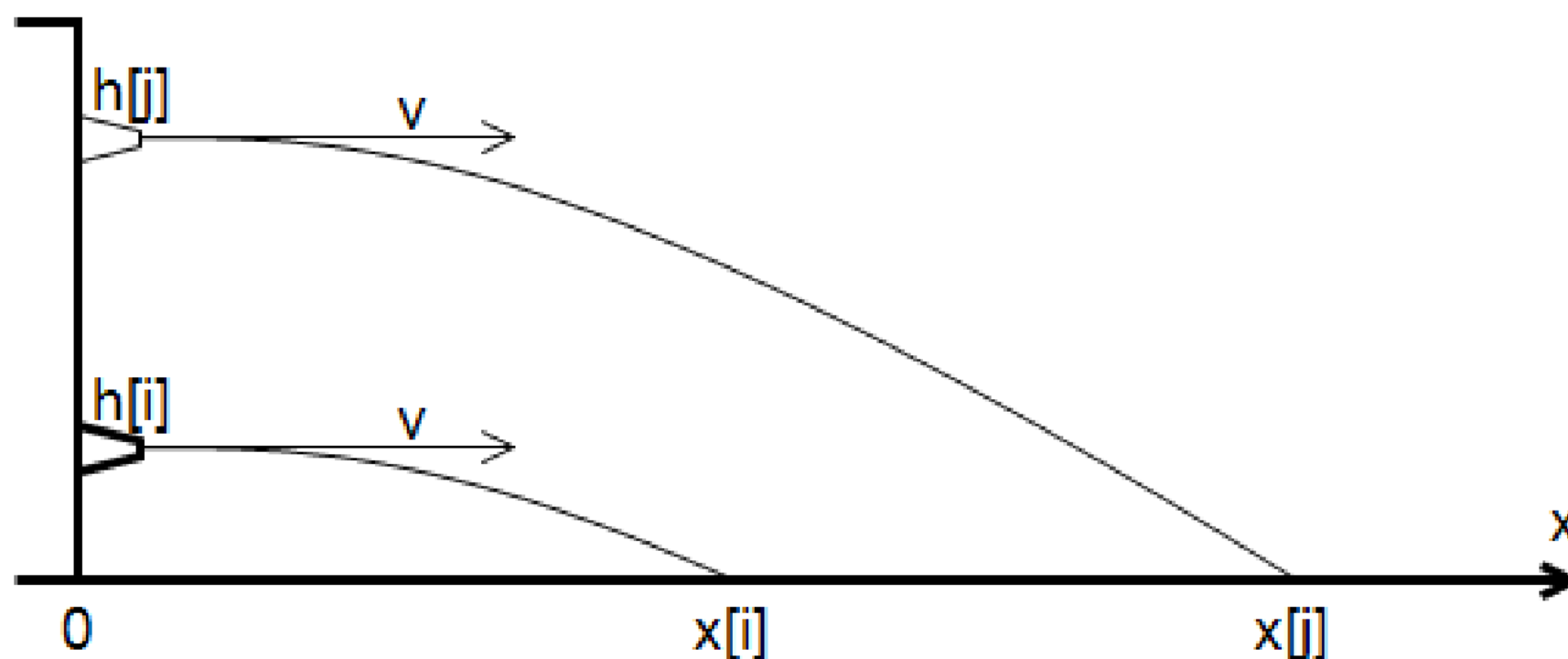
стандартний ввід	стандартний вивід
3 6 9 11	198

## Задача С. Облога замку

Назва вхідного файлу:	стандартний ввід
Назва вихідного файлу:	стандартний вивід
Обмеження за часом:	1 секунда
Обмеження за пам'яттю:	64 мегабайт

Одного разу в середньовіччі невелика група людей вирішила захопити замок. Для цього посеред ночі вони знайшли  $N$  гармат і з ними поїхали на поле, що знаходиться одразу перед замком. Коли ж було вирішено почати атаку, загарбники помітили, що забули взяти з собою гарматні ядра. Тому вони вирішили швидко повернутись, знайти необхідну кількість ядер, і вже потім повернутись і розпочати обстріл.

Однак, як тільки загарбники зникли з поля, прокинувся король і вирішив оглянути поле. Те, що він побачив, йому не сподобалось, і він вирішив позбутись ворожих гармат. Для цього він вирішив використати систему оборони замку, яку розробив його двоюрідний брат. Система оборони являє собою одну гармату, яку можна зарядити одразу декількома ядрами. Як помітив король, у таку гармату можна зарядити одразу  $N$  ядер, ще й залишиться вільне місце. Конструкція даної гармати є доволі складною - вона може стріляти лише горизонтально, однак вона може підійматись на будь-яку висоту. Також вона може стріляти лише коли підіймається угору, а якщо вона починає спуск униз, то завжди опускається до рівня землі, і у проміжку зупинитись не може. Починає свій рух вона також від рівня землі (висота над землею складає 0 метрів).



Поки король пояснював як влаштована гармата, він побачив як на обрії загарбники вже несуть ядра до своїх гармат. Тому було прийнято рішення про початок обстрілу, причому, як підказали королю, обстріл потрібно провести за один підйом гармати, і за один раз уразити всі цілі. Однак, на яких висотах необхідно виконувати постріли - у замку ніхто не знає. Саме тому й вирішили звернутись до вас. Як підказав двоюрідний брат короля - початкова горизонтальна складова швидкості польоту ядра дорівнює  $V$  м/с, і на весь час польоту ядра вона (горизонтальна складова) не змінюється, поки не влучить у ціль. Придворні науковці підказали, що кожна з  $N$  ворожих гармат знаходиться на відстані  $x_i$

метрів від замку, а також що у цьому місці  $g = 9.81 \text{ м/с}^2$ , і що опором повітря можна нехтувати.

## Формат вхідного файлу

У вхідному файлі у першому рядку дано два числа  $V$  (дійсне число,  $1 \leq V \leq 100$ ) та  $N$  (ціле число,  $1 \leq N \leq 100$ ). У наступному рядку дано  $N$  чисел -  $x_i$  - відстані від замку до ворожих гармат у довільному порядку (дійсні числа,  $1 \leq x_i \leq 1000$ )

## Формат вихідного файлу

У вихідному файлі необхідно вивести  $N$  чисел у порядку зростання - висоти, з яких необхідно робити постріли, щоб влучати у всі цілі. Усі числа - дійсні з точністю до двох цифр після коми.

## Приклад

стандартний ввід	стандартний вивід
1 3	4.91
2 1 1.41	9.75
	19.62



# Задача D. Лавруша і висота шеренги

Назва вхідного файлу:	стандартний ввід
Назва вихідного файлу:	стандартний вивід
Обмеження за часом:	1 секунда
Обмеження за пам'яттю:	64 мегабайт

Лавруша - хлопець мрійливий. Цього разу на уроках математики він думає про ... фізкультуру.

Нехай в класі Лавруші  $N$  учнів, які на уроці фізкультури стоять в шерензі. На жаль, дисципліна в колективі нашого мрійника кульгає, тому учні не обов'язково стоять в порядку неспадання по зросту. Назвемо підшеренгою шеренги якусь послідовність учнів, які стоять підряд. Назвемо висотою підшеренги, в якій знаходяться  $X$  учнів, висоту  $[(X + 1)/2]$  по зросту учня серед тих, хто знаходиться в даній підшерензі.

Лавруша має зріст  $M$  одиниць (які то одиниці історія замовчує). І тут наш герой задумався: а скільки підшеренг даної шеренги мають висоту не меншу, ніж зріст Лавруші.

## Формат вхідного файлу

В першому рядку знаходяться два числа:  $N$  і  $M$  ( $1 \leq N \leq 100000$ ;  $0 \leq M \leq 10^9$ ), які розділені одиничним пробілом - кількість учнів в шерензі і зріст Лавруші.

В наступному рядку знаходяться  $N$  чисел  $a_i$  ( $0 \leq a_i \leq 10^9$ ;  $1 \leq i \leq N$ ) - зріст  $i$ -го учня в шерензі. Всі числа - цілі невід'ємні.

## Формат вихідного файлу

Виведіть одне число - кількість підшеренг даної шеренги, висота яких не менша за зріст Лавруші.

## Приклад

стандартний ввід	Стандартний вивід
5 3 1 2 3 4 5	9

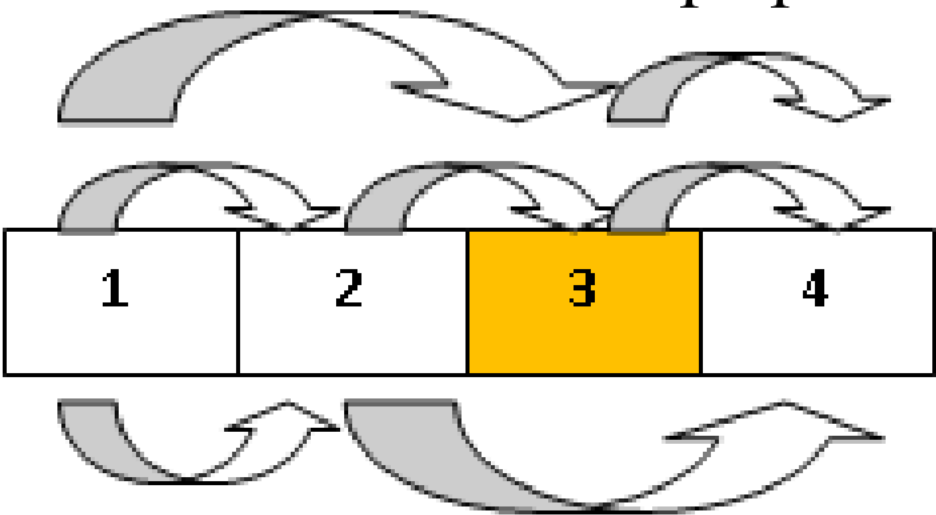
# Задача Е. Кількість шляхів

Назва вхідного файлу:	стандартний ввід
Назва вихідного файлу:	стандартний вивід
Обмеження за часом:	1 секунда
Обмеження за пам'яттю:	64 мегабайт

Є стрічка довжини  $N$ , одна із клітинок якої є особливою. Ми можемо робити кроки двох видів - на наступну клітинку та через одну клітинку. Потрібно знайти загальну кількість шляхів, які:

- 1) обов'язково проходять через особливу клітинку;
- 2) обов'язково не проходять через особливу клітинку.

Наприклад:  $N = 4$  і особлива клітинка номер три.



Отже, ми маємо два шляхи, які обов'язково проходять через особливу клітинку 3 ( $1 - 2 - 3 - 4$ ,  $1 - 3 - 4$ ) та один шлях, який не проходить через особливу клітинку 3 ( $1 - 2 - 4$ ).

## Формат вхідного файлу

Єдиний рядок містить два числа  $N$  та  $M$  ( $3 \leq N \leq 50$ ,  $1 < M < N$ ) – кількість клітинок та номер особливої клітинки.

## Формат вихідного файлу

Єдиний рядок має містити одне число – кількість знайдених шляхів.

## Приклад

стандартний ввід	Стандартний вивід
4 3	3

# Задача F. Прогулянка містом

Назва вхідного файлу:	стандартний ввід
Назва вихідного файлу:	стандартний вивід
Обмеження за часом:	1 секунда
Обмеження за пам'яттю:	64 мегабайт

У Києві, неподалік Дніпра стоїть Козак: кам'яний, з великими вусами, величезною шаблею, до неба гордо піднесеною головою. Київ – улюблене місто туристів, які з'їжджаються сюди звідусіль. Василько – один з них. Карта Києва являє собою таблицю, розміром  $N \times M$ , клітинками якої є квартали, деякі з яких доступні для руху, а деякі – ні. Також на карті відмічені цікаві місця, причому це також квартали, по яких можливий рух. Василько вибрав маршрут для ознайомлення з містом. Він почне з клітинки (1; 1), прийде в клітинку ( $N$ ;  $M$ ), а потім знову повернеться туди звідки прийшов (в клітинку (1; 1)). Звичайно, Василько хоче переглянути якнайбільше різних цікавих місць. Оскільки Василько знає Київ дуже погано, та й, взагалі, заблукати тут дуже небажано, він вирішив, що на шляху до клітинки ( $N$ ;  $M$ ) буде завжди йти вправо, або вниз (відносно карти), а на зворотному шляху – вліво, або вгору. Допоможіть Василькові знайти максимальну кількість різних цікавих місць, в яких він може побувати.

## Формат вхідного файлу

У першому рядку вхідного файлу містяться два числа –  $N$  та  $M$  ( $1 \leq N; M \leq 100$ ). У наступних  $N$  рядках міститься опис карти – рядок з  $M$  літер, причому “#” означає квартал недоступний для руху, “\*” – місце для перегляду, а “.” (крапка) – квартал доступний для руху. Гарантується, що клітинки (1; 1) й ( $N$ ;  $M$ ) не будуть позначені символом “#”.

## Формат вихідного файлу

У єдиному рядку вихідного файлу має міститись одне число – максимальна кількість різних місць для перегляду, в яких може побувати Василько протягом свого маршруту, або -1, якщо маршрут здійснити не вдасться.

## Приклад

стандартний ввід	стандартний вивід
4 5 **.*. ..##. *#*.. *.*.*	7

# Задача G. Цікаве число

Назва вхідного файлу:	стандартний ввід
Назва вихідного файлу:	стандартний вивід
Обмеження за часом:	2 секунди
Обмеження за пам'яттю:	256 мегабайт

Для заданого числа N знайдіть найменше додатне ціле число з сумою цифр N, яке ділиться на N.

## Формат вхідного файлу

У першому рядку знаходиться число N ( $1 \leq N \leq 1000$ ).

## Формат вихідного файлу

У першому рядку виведіть знайдене число.

## Приклад

стандартний ввід	стандартний вивід
1	1
10	190



# Задача Н. Кількість трійок

Назва вхідного файлу:	стандартний ввід
Назва вихідного файлу:	стандартний вивід
Обмеження за часом:	3 секунди
Обмеження за пам'яттю:	64 мегабайт

Підрахуйте кількість різних трійок чисел  $A, B, C$  (числа – цілі, більші за нуль), для яких виконується умова

$$A + B^2 + C^3 \leq N.$$

## Формат вхідного файлу

Єдиний рядок містить число  $N$  ( $3 \leq N \leq 10^{18}$ ).

## Формат вихідного файлу

Єдиний рядок має містити одне число – відповідь на задачу.

## Приклад

стандартний ввід	стандартний вивід
4	2

## Пояснення до прикладу

є дві трійки чисел, які задовольняють умову:

(1, 1, 1)     $1 + 1^2 + 1^3 = 3 \leq 4$

(2, 1, 1)     $2 + 1^2 + 1^3 = 4 \leq 4$

# Задача І. Цікава задача

Назва вхідного файлу:	стандартний ввід
Назва вихідного файлу:	стандартний вивід
Обмеження за часом:	1 секунда
Обмеження за пам'яттю:	64 мегабайт

Роману, як найкращому фахівцю компанії Loo..., було запропоновано вирішити досить нелегку задачу. Спочатку в пам'яті комп'ютера є єдине число  $k$ . Потрібно за найменшу кількість операцій отримати в пам'яті комп'ютера такі числа:  $A_1k$ ,  $A_2k$ , ...,  $A_nk$ . При цьому комп'ютер може виконувати такі операції:

- додавання двох чисел,
- віднімання двох чисел,
- побітовий зсув вліво (зсув на  $t$  бітів еквівалентний множенню на  $2^t$ ).

Усі отримані проміжні значення зберігаються в пам'яті, тому їх можна використовувати при обчисленні інших значень. Замовники хочуть, щоб при обчисленні ніколи не виходило число більше за  $42k$ . Гарантується, що при дотриманні цього обмеження, в комп'ютері не відбувається переповнення. Також комп'ютер не може працювати з від'ємними числами, тому віднімати від меншого числа більше також не можна.

Порядок, в якому в пам'яті з'являться числа  $A_1k$ ,  $A_2k$ , ...,  $A_nk$  не має значення.

У Романа побачення, тому він просить вашої допомоги.

## Формат вхідного файлу

Перший рядок містить число  $N$  ( $1 \leq N \leq 41$ ) – кількість необхідних значень. У другому рядку міститься  $N$  чисел  $A_i$  ( $2 \leq A_i \leq 42$ ). Усі числа  $A_i$  різні. Саме число  $k$  не дано, тому ваша послідовність операцій має бути правильною для будь-якого значення  $k$ .

## Формат вихідного файлу

У першому рядку виведіть одне число – мінімальну кількість операцій. Далі виведіть необхідні операції у наступному форматі:

- зсув вліво  $Ak$  на  $t$  біт: " $A<<t$ ",
- додавання  $Ak$  і  $Bk$ : " $A+B$ ",
- віднімання  $Ak$  від  $Bk$ : " $B-A$ ".

Приклад

стандартний ввід	стандартний вивід
1 29	4 1<<1 1<<5 1+2 32-3
3 3 5 18	5 1<<1 1<<4 1+2 2+3 2+16