

## А. Период последовательности

Ограничение по времени: 1 секунда

Ограничение по памяти: 64 мегабайта

Входные данные: с клавиатуры

Выходные данные: на экран

Совсем недавно Вася познакомился с числами Фибоначчи. Первым делом он начал решать задачи, связанные с этими числами. Ему очень понравилось решать такие задачи, поэтому он надеется, что и Вам они тоже доставят удовольствие. Для разминки Вася предложил решить следующую задачу...

Назовем числами Фибоначчи последовательность чисел, удовлетворяющую следующим трем условиям:

1.  $f_0 = 0$ ;
2.  $f_1 = 1$ ;
3.  $f_n = f_{n-1} + f_{n-2}$ , для всех  $n \geq 2$ .

Запишем первые пятнадцать чисел Фибоначчи начиная с нулевого индекса:

0, 1, 1, 2, 3, 5, 8, 13, 21, 34, 55, 89, 144, 233, 377...

Можно заметить, что числа растут очень быстро, поэтому вместо самих чисел Фибоначчи в последовательность будем заносить их остатки от деления на число  $2^N$ . Например, для  $N = 1$  последовательность будет выглядеть следующим образом:

0, 1, 1, 0, 1, 1, 0, 1, 1, 0, 1, 1, 0, 1, 1, 0...

Периодом последовательности  $A_i$  назовем минимальное натуральное число  $K$  такое, что для любых целых неотрицательных  $i$  выполняется условие  $A_i = A_{i+K}$ .

Вася просит найти период последовательности остатков от деления чисел Фибоначчи на число  $2^N$ .

### Входные данные

На вход программе подается единственное целое число  $N$  ( $1 \leq N \leq 30$ ).

### Выходные данные

В результате работы программа должна вывести единственное целое число  $K$  – ответ на задачу.

### Примеры тестов

Входные данные	Выходные данные
1	3
3	12
6	96

## В. Надежный корабль

Ограничение по времени: 1 секунда

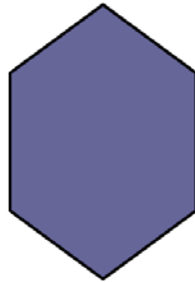
Ограничение по памяти: 64 мегабайта

Ввод: с клавиатуры

Вывод: на экран

Вася всегда интересовался безграничным космосом. На этот раз он решил спроектировать надежный космический корабль, предназначенный для межзвездных перелетов. У него уже есть одна идея для корабля, но ему интересно, будет ли надежен такой корабль при больших перегрузках.

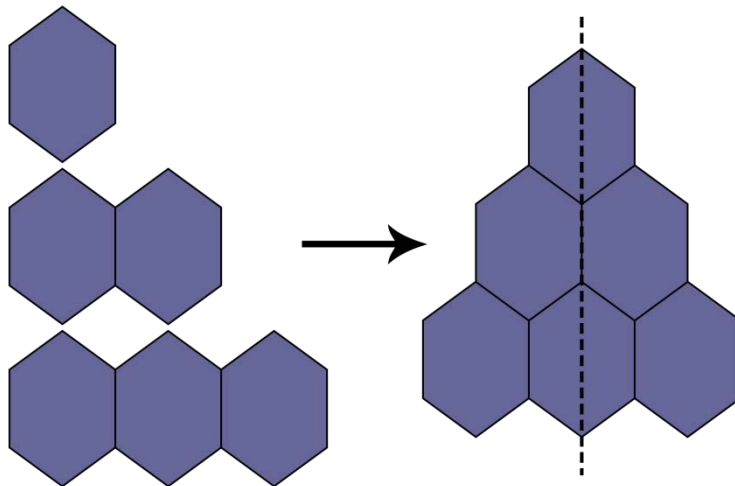
Корабль Васи представляет из себя плоскую фигуру, состоящую из  $N$  слоев. Каждый слой составлен из некоторого количества частей, представляющих собой такие шестиугольники:



Вася выписал количество частей в каждом слое корабля начиная с верхнего слоя. Например, если он выписал такие числа:

1, 2, 3

то корабль можно получить следующим образом:



Вася считает, что корабль будет достаточно надежным, если он будет симметричным относительно вертикальной оси. На примере выше корабль надежен, штриховой линией показана ось симметрии корабля.

Вася хочет знать, можно ли расположить слои корабля так, чтобы получить симметричную фигуру. Менять порядок слоев нельзя, как и вращать их. Части корабля не могут накладываться друг на друга и могут соединяться с другими частями только по сторонам, как на примере выше.

### Входные данные

В начале на вход программе подается одно целое число  $N$  ( $1 \leq N \leq 20$ ) – количество слоев корабля.

Затем на вход программе подается  $N$  целых чисел  $A_i$  ( $1 \leq A_i \leq 100$ ) – количество частей в каждом слое корабля.

## Выходные данные

В результате работы программа должна вывести "Yes", если можно получить надежный корабль и "No" – в противном случае.

## Примеры тестов

Входные данные	Выходные данные
3 1 2 3	Yes
3 1 2 4	No
5 1 2 3 2 5	Yes

## С. Сложность маршрута

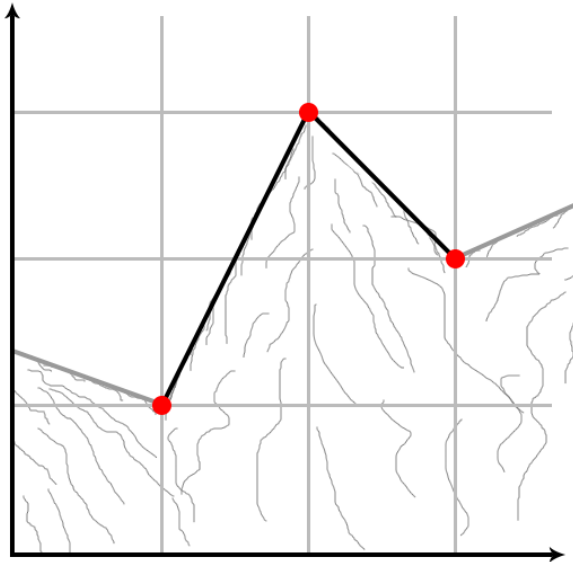
Ограничение по времени: 1 секунда

Ограничение по памяти: 64 мегабайта

Ввод: с клавиатуры

Вывод: на экран

Вася решил отправиться в поход. Он уже составил маршрут и начал собирать все необходимые вещи. В последний момент Вася заподозрил, что маршрут может оказаться слишком сложным, что может плохо сказаться на его здоровье. Поэтому он решил определить сложность выбранного им маршрута.



Сложностью маршрута Вася считает минимальное количество энергии  $E$  необходимое для того, чтобы преодолеть весь маршрут. Он выделил на маршруте  $N$  точек и записал их высоты. Для того, чтобы пройти путь от точки  $A$  до точки  $B$  ему понадобится  $H(B) - H(A)$  энергии, где  $H(x)$  обозначает высоту точки  $x$ . В случае, если  $H(A) > H(B)$  он восстановит  $H(A) - H(B)$  энергии. Вася не сможет пройти путь от точки  $A$  до точки  $B$ , если его текущая энергия  $E'$  окажется меньше необходимой на это энергии.

Помогите Васе определить сложность  $E$  выбранного маршрута.

### Входные данные

В начале на вход программе подается одно целое число  $N$  ( $2 \leq N \leq 20$ ) – количество точек на маршруте.

Затем на вход программе подается  $N$  целых чисел  $H_i$  ( $0 \leq H_i \leq 1\,000$ ) – высоты всех точек на маршруте.

### Выходные данные

В результате работы программа должна вывести единственное число  $E$  – сложность маршрута.

### Примеры тестов

Входные данные	Выходные данные
3 1 3 2	2
3 2 3 1	1
5 1 1 1 1 1	0

## D. Простая задача

Ограничение по времени: 2 секунды

Ограничение по памяти: 256 мегабайт

Ввод: с клавиатуры

Вывод: на экран

Простые задачи нужно решать в последнюю очередь...

Скорее всего Вася, но это не точно

У любой простой задачи условие должно быть простым и понятным. Вася считает очевидным, что и решение такой задачи тоже должно быть простым и понятным. Конечно, Вася не хочет решать простые задачи, поэтому он просит Вас сделать это.

В простой задаче обязательно должны присутствовать простые числа. Эта задача, конечно, не исключение. Простым называется натуральное число  $P > 1$  имеющее ровно два делителя: 1 и  $P$ . Ваша задача заключается в том, чтобы найти все натуральные числа, не делящиеся ни на один из квадратов простых чисел и не превышающие число  $N$ .

Вася хочет знать для заданного числа  $N$  ответ на такую простую задачу.

### Входные данные

На вход программе подается одно целое число  $N$  ( $1 \leq N \leq 10^9$ ) – верхняя граница чисел.

### Выходные данные

В результате работы программа должна вывести единственное число – ответ на задачу.

### Примеры тестов

Входные данные	Выходные данные
3	3
4	3
6	5

## Е. Дождливая погода

Ограничение по времени: 1 секунда  
Ограничение по памяти: 64 мегабайта  
Входные данные: с клавиатуры  
Выходные данные: на экран

Кошка Ханна решила отдохнуть на своем дачном участке. К сожалению, погода в этот день сильно испортилась. Небо нахмурилось, пошел дождь. Кошке не сильно хочется промокнуть под дождем, поэтому она хочет укрыться от него под навесом.

Дачный участок представляет из себя полосу длиной в  $N$  метров и шириной в 1 метр. Навес также имеет форму полосы шириной в 1 метр и может закрыть  $M$  метров дачного участка от дождя. Сам же дождь шел не на всем участке, а лишь на каких-то его частях. Ханне пришлось обойти все  $N$  метров дачного участка, чтобы определить, где шел дождь, а где дождя не было. Так она обозначила каждый метр участка нулем, если дождя там не было, и единицей – в случае, если он там был. В итоге у нее получилась строка длины  $N$ , состоящая только из нулей и единиц, где  $i$ -й символ соответствовал  $i$ -му метру дачного участка.

Помогите Ханне определить, какое максимальное число метров дачного участка не будет под дождем, если она установит навес где-то на своем участке.

### Входные данные

На вход программе подается два целых числа  $N$  и  $M$  ( $1 \leq M \leq N \leq 30$ ) – размеры дачного участка и навеса соответственно.

Далее на вход программе подается строка длины  $N$ , состоящая из символов '0' и '1'.

### Выходные данные

В результате работы программа должна вывести единственное целое число – ответ на задачу.

### Примеры тестов

Входные данные	Выходные данные
9 6 010001001	8
5 2 11111	2

## Ф. Игра в снежки

Имя входного файла: стандартный ввод  
Имя выходного файла: стандартный вывод  
Ограничение по времени: 1 секунда  
Ограничение по памяти: 64 мегабайта

Ребята установили во дворе деревянный щит, нарисовали на нем три окружности разного диаметра с общим центром и устроили соревнование — кто наберет больше очков за попадание снежком в эту мишень. По придуманным ими правилам, за попадание в самую маленькую окружность дается 10 очков, если снежок попал в промежуток между малой и средней окружностью — 7 очков, между средней и большой — 3 очка, за попадание в щит за пределами окружностей — одно очко.

Митя играл с ребятами в эту игру и набрал  $n$  очков. К сожалению, он забыл, сколько бросков он сделал. Помогите Мите, определите, какое **минимальное число** бросков он мог сделать, чтобы набрать  $n$  очков.

### Формат входных данных

В единственной строке задано единственное целое число  $n$  — количество набранных очков.

$$1 \leq n \leq 10^{18}$$

### Формат выходных данных

Выведите единственное целое число — минимальное количество брошенных снежков.

### Примеры

стандартный ввод	стандартный вывод
2	2
3	1
11	2

### Замечание

В третьем тестовом примере Митя мог сделать два броска следующим образом: одним попасть в самую маленькую окружность, другим — в щит за пределами окружностей.

## Г. Снежные шары

Ах, какая же прекрасная пора — зима. Мягкий, пушистый снег засыпал улицы. И как с такими сугробами не начать лепить снеговиков?

Вообще, Митя не очень то и любит лепить снеговиков, но увидев высоченные сугробы он понял, что сейчас самое время начать. Он решил слепить такое количество снеговиков, на которое хватит сил. А как известно, снеговик состоит из снежных шаров и прочих украшений, поэтому Митя начнет именно с лепки снежных шаров.

За один час Митя успевает слепить  $n$  снежных шаров. Он решил, что проведет  $t$  часов на улице и хочет определить, сколько снежных шаров он успеет подготовить за это время. Так как у Мити есть еще и другие дела, он просит вас посчитать это.

### Формат входных данных

В единственной строке задано два целых числа  $n$  и  $t$  — количество снежных шаров за час и количество часов проведенных на улице.

$$1 \leq n \leq 500$$

$$1 \leq t \leq 6$$

### Формат выходных данных

Выведите единственное целое число — общее количество снежных шаров.

### Примеры

стандартный ввод	стандартный вывод
5 1	5
404 5	2020

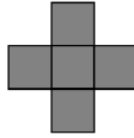
### Замечание

В первом тестовом примере Митя слепит 5 снежных шаров за один час.



## Н. Снежная картина

Митя решил нарисовать картину. Для этого у него есть специальная кисточка, которая одним нажатием позволяет нарисовать снежинку. Для простоты будем считать, что картина представляет из себя клетчатую бумагу, состоящую из  $n$  строк и  $m$  столбцов, где каждая клетка либо закрашена, либо нет. При нажатии кисточкой, клеточка, на которую нажали, а также четыре соседние по стороне клеточки закрашиваются, как на рисунке ниже.



Пример снежинки

Митя уже решил, какую картину он хочет нарисовать, но не уверен, получится ли у него это сделать, используя специальную кисть. Помогите Мите, определите, можно ли нарисовать заданную картину.

Обратите внимание, что Митя не хочет, чтобы стол, на котором лежит бумага испачкался, поэтому снежинка **не может попадать за границы** клетчатой бумаги.

### Формат входных данных

В первой строке задано два целых числа  $n$  и  $m$  — количество строк и столбцов на клетчатой бумаге.

$$1 \leq n, m \leq 1000$$

В следующих  $n$  строках содержится по  $m$  символов — описание желаемой картины. Если клетка должна быть закрашена, то в этой ячейке содержится символ «#», в противном случае символ «.».

### Формат входных данных

В первой строке задано два целых числа  $n$  и  $m$  — количество строк и столбцов на клетчатой бумаге.

$$1 \leq n, m \leq 1000$$

В следующих  $n$  строках содержится по  $m$  символов — описание желаемой картины. Если клетка должна быть закрашена, то в этой ячейке содержится символ «#», в противном случае символ «.».

### Формат выходных данных

В первой строке выведите «Yes», если можно нарисовать картину используя снежинки, в противном случае выведите «No».

Если картину нарисовать можно, то в следующих  $n$  строках выведите по  $m$  символов «.» или «\*», где «\*» означает место, где нужно нажать кисточкой (центр снежинки).

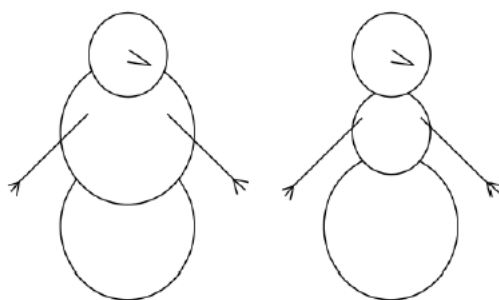
## Примеры

стандартный ввод	стандартный вывод
4 4 .#.. ###. .### ..#.	Yes .... .*.. ..*. ....
3 3 #.. ##. #..	No
3 3 .#. #.# .#.	No

## I. Снеговики

Пришло время строить снеговиков. Традиционно, снеговик состоит из трех шаров, причем верхний шар — самый маленький, а нижний — самый большой. К сожалению, у Мити получились снежные шары только двух размеров: маленькие и большие, поэтому он решил, что его снеговики должны будут удовлетворять только следующим критериям:

1. Снеговик должен состоять из трех снежных шаров;
2. Верхний шар должен быть маленьким;
3. Нижний шар должен быть большим.



Возможные варианты снеговиков

Митя подсчитал, что у него есть  $n$  маленьких и  $m$  больших шаров. Он хочет узнать, а какое максимальное количество снеговиков он сможет составить, используя только эти шары. Помогите Мите, определите это число.

### Формат входных данных

В единственной строке задано два целых числа  $n$  и  $m$  — количество маленьких и больших снежных шаров.

$$1 \leq n, m \leq 15000$$

### Формат выходных данных

Выведите единственное целое число — максимальное количество снеговиков, которое удастся получить.

### Примеры

стандартный ввод	стандартный вывод
1 2	1
3 3	2

### Пояснение к задаче

В первом тестовом примере можно составить только одного снеговика из одного маленького и двух больших шаров.

Во втором тестовом примере можно составить два снеговика: одного снеговика из двух маленьких шаров и одного большого, второго снеговика из двух больших шаров и одного маленького.

## Ж. Точка

На плоскости отмечены все точки, обе координаты которых - натуральные числа. Будет ли видна точка с координатами  $(n, m)$  из начала координат - точки  $(0, 0)$ ? Считаем, что точка видна, если внутри отрезка, соединяющего точки  $(0,0)$  и  $(n, m)$ , нет отмеченных точек. Например, точка  $(2, 2)$  не видна из начала координат, так как ее не дает увидеть точка  $(1, 1)$ . Числа  $n$  и  $m$  являются натуральными и  $n \leq 2000000000$ ,  $m \leq 2000000000$ .

**Ввод:** натуральные числа  $n, m$

**Вывод:** слово «No», если точка  $(n, m)$  не видна из начала координат, и, на следующей строке, - координаты  $(x, y)$  точки, которая не дает увидеть точку  $(n, m)$  из начала координат и расстояние от которой до начала координат минимальное.

Слово «Yes», если точка  $(n, m)$  видна из начала координат.

Пример:

<i>Ввод</i>	<i>Вывод</i>
100	No
200	1 2

## К. Воробьи

*Прилетит воробей, схватит зернышко и улетит...*

Есть амбар, в который помещается  $N$  зерен и в него каждый день привозят еще  $M$  зерен. Изначально амбар полон. Каждый день происходит следующее:

- В амбар привозят еще  $M$  зерен. Если  $M$  зерен в амбар не поместится, то добавляется максимально возможное число зерен.
- Прилетают воробьи и клюют зерно. В первый день прилетает один воробей, во второй день - два воробья, в третий - три и так далее. Каждый воробей съедает по одному зернышку. Если амбар пуст, воробей ничего не съедает.

Женя любит наблюдать за тем, как воробьи клюют зерно. Однажды ему стало интересно, в какой день амбар впервые станет пуст. Помогите Жене и напишите программу, которая найдет это число. Дни нумеруются, начиная с единицы.

**Входные данные:** Два числа  $N$  и  $M$  ( $1 \leq N, M \leq 100000$ )

**Выходные данные:** Одно число - день, когда амбар впервые станет пуст.

**Примеры:**

Ввод	Вывод
5 2	4
8 1	5

## Л. Конфеты

Однажды Петя, решая задачи, проголодался и решил пойти попить чай с конфетами. У него дома было много разных сортов конфет. Он выложил их всех ряд и решил есть не по порядку, а особым образом.

Сначала он находит в ряду две рядом лежащие конфеты одного сорта, съедает их, потом сдвигает все конфеты и опять ищет две рядом лежащие одного сорта, съедает их, и т.д.

У Пети много конфет, он хочет узнать сможет ли он съесть их все, таким способом.

### Входные данные

В первой строке находится целое число  $N$  ( $1 \leq N \leq 15000$ ).

Во второй строке находятся  $N$  целых чисел, разделенные пробелом, каждое означает сорт конфеты. Числа не превышают 1000 и больше 0.

### Выходные данные

Вывести 'YES' (без кавычек) если Петя сможет съесть все конфеты, а если не сможет, то вывести 'NO' (без кавычек).

### Пример

Входные данные	Выходные данные
2 1 1	YES
3 1 1 1	NO
4 2 3 3 2	YES

В этой задаче потестовая оценка.

*Примечание:*

*4*

*2 3 3 2 -> 2 2 -> нет конфет*

*Сначала Петя съест две конфеты 3 сорта. Потом две конфеты 2 сорта. Значит он может съесть все - выведем YES.*

## М. Волшебник

Известный фокусник давно наскучил своим зрителям, показывая одни и те же трюки. Поэтому он решил привлечь публику интеллектуальными математическими фокусами.

Суть нового фокуса заключается в том, что зритель загадывает некоторое натуральное число  $N$ . А на основе этого числа фокусник называет количество пар натуральных чисел  $A$  и  $B$  таких, что квадрат меньшего из них на  $N$  меньше квадрата большего из них.

Например, если  $N = 15$ , то таких пар чисел всего две:  $(4, 1)$ ,  $(8, 7)$ . Действительно  $4^2 - 1^2 = 15$  и  $8^2 - 7^2 = 15$ . Больше пар чисел, соответствующих ограничениям задачи, нет.

Поскольку фокусник не очень силен в математике, но обладает феноменальной памятью, то перед новым сезоном представлений просит написать программу, которая по заданному числу  $N$  вычисляет количество пар чисел, удовлетворяющих описанным выше условиям. В силу того, что число  $N$  невелико, то фокусник в состоянии запомнить все необходимые пары чисел  $A$  и  $B$ . В итоге у математически грамотных зрителей должно сложиться впечатление, что фокусник обладает феноменальными вычислительными способностями.

### Входные данные.

Дано единственное натуральное число  $N$  ( $1 \leq N \leq 1000$ ).

### Выходные данные.

Требуется вывести количество пар чисел  $A$  и  $B$  ( $1 \leq A, B \leq 500$ ), которые удовлетворяют условию задачи.

### Примеры.

Входные данные	Выходные данные
15	2

Группа 1:  $N \leq 100$  – 30 баллов

Группа 2:  $N \leq 500$  – 30 баллов

Группа 3: без ограничений – 40 баллов

## Н. Марафон

Сегодня Бит и его друг Байт бегут марафон.

Перед его началом жюри предложило участникам на выбор  $n$  маршрутов, каждый из которых начинается в точке  $(x_i, y_i)$  и заканчивается в точке  $(x_j, y_j)$ .

Длиной маршрута называется величина  $|x_i - x_j| + |y_i - y_j|$ . Так как Бит – очень ленивый мальчик, то он выберет себе самый короткий маршрут. Если же таких маршрутов несколько, то Бит выберет себе маршрут с наименьшим номером.

Байт тоже довольно ленив, поэтому он выберет себе самый короткий маршрут из тех, которые не выбрал Бит. Если же таких маршрутов несколько, то Байт выберет себе маршрут с наибольшим номером.

Какие же маршруты выберут себе Бит и Байт?

### Входные данные

Первая строка содержит целое число  $n$  ( $2 \leq n \leq 10^5$ ) – количество маршрутов.

Следующие  $n$  строк содержат по 4 целых числа  $x_i, y_i, x_j, y_j$  – координаты начала и конца очередного маршрута. Все координаты по модулю не превосходят  $10^9$ .

### Выходные данные

Выведите два индекса  $i$  и  $j$  ( $1 \leq i, j \leq n$ ) – номера маршрутов, которые выберут Бит и Байт соответственно

Пример.

Входные данные	Выходные данные
3 3 2 1 5 3 4 4 2 2 2 4 2	3 2

Группа 1:  $n \leq 10$  – 18 баллов

Группа 2:  $n \leq 1000$  – 30 баллов

Группа 3: без ограничений – 52 балла

## О. Спички

Как известно в различных цифровых устройствах для отображения цифр используется семисегментный идентификатор, который представляет собой семь лампочек. Различные комбинации активных и неактивных лампочек позволяют получить некоторую цифру.

В домашних условиях быстрее и удобнее моделировать работу семисегментного идентификатора с помощью спичек. На рис. 1 показано, каким образом можно получить любую цифру с помощью спичек.



Рис.1. Изображение цифр с помощью спичек

Ваня в детстве решал множество задач со спичками. Теперь, когда он решил стать программистом, чтобы не терять ранее приобретенные математические навыки, он поставил себе такую задачу: пусть дано  $N$  спичек, какое минимальное и максимальное число можно получить, используя все имеющиеся в запасе спички?

### **Входные данные.**

Дано единственное натуральное число  $N$  ( $1 \leq N \leq 100$ ) – количество спичек.

### **Выходные данные.**

Требуется вывести два числа – минимальное и максимальное число, которое может быть получено с помощью всех имеющихся спичек. Числа должны быть разделены ровно одним пробелом.

### **Примеры.**

<b>Входные данные</b>	<b>Выходные данные</b>
3	7 7
7	8 711