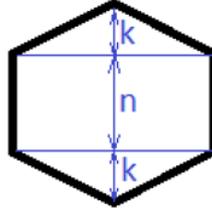
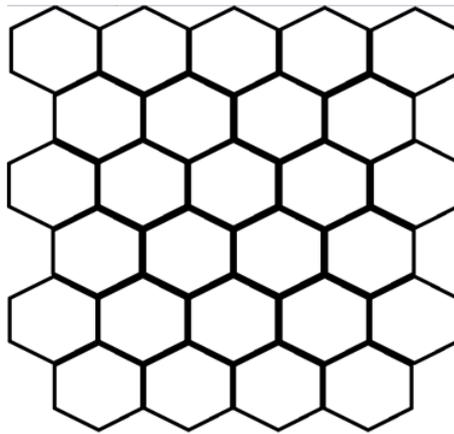


Задача 1. Творческая натура

Оля — творческая натура: даже в самых простых вещах она ценит уникальность и красоту исполнения. Оля решила повесить в прихожей своей квартиры зеркало, собрав большую зеркальную поверхность из плиток в форме шестиугольника, представленного на рисунке ниже: n и k — это размеры областей плитки в сантиметрах.



Для формирования зеркальной поверхности Оля планирует использовать t рядов плитки (при этом t всегда **четное**), укладывая их таким образом, чтобы в 1, 3, 5 и т.д. рядах было m плиток, а во 2, 4, 6 и т.д. было $m - 1$ плиток. Вариант укладки при $t = 6$ и $m = 5$ изображен ниже.



Оля хочет понять какой высоты у нее получится зеркало и сколько плиток ей нужно для этого купить, но у девочки не очень хорошо с математикой.

В таблице ниже приведены пять вариантов значений n , k , t , m :

n	k	t	m
10	5	4	5
17	7	8	10
64	16	14	15
99	33	20	25
49	7	50	49

Для каждого из них нужно посчитать два числа:

1. Высота зеркала в сантиметрах,
2. Количество плиток, составляющих зеркальную поверхность.

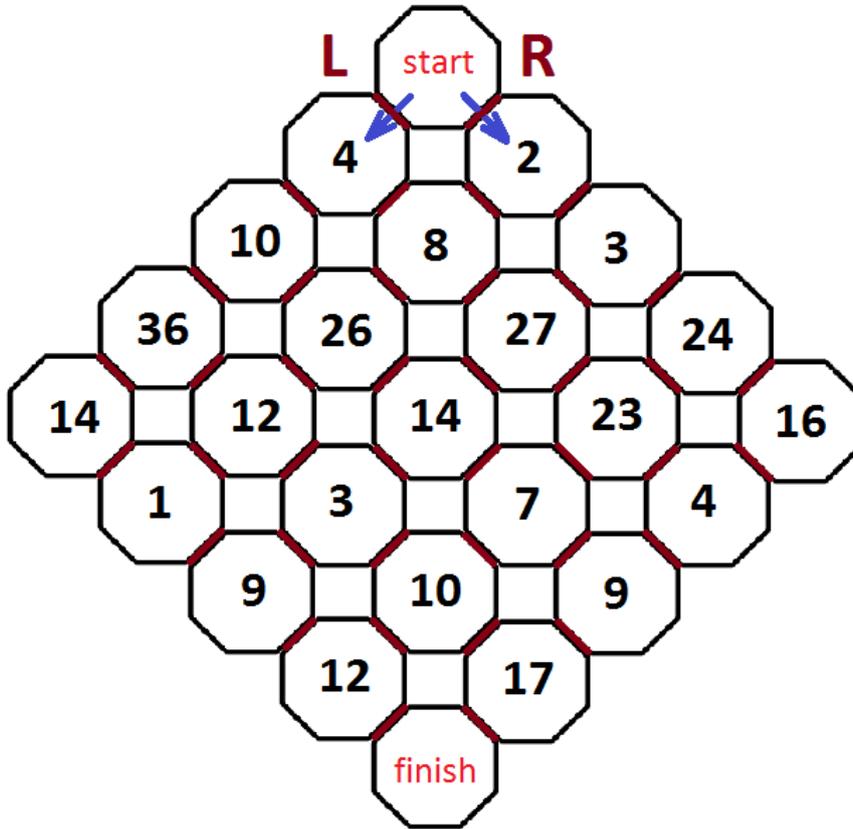
Ответ для каждого варианта значений должен быть записан в отдельной строке. В ответе для каждого варианта Вам нужно вывести через пробел **сначала высоту** зеркала в сантиметрах, а **затем количество плиток**, составляющих зеркальную поверхность.

Обратите внимание, что Вам необходимо в пяти строках указать по два целых числа (итого 10 чисел) — в каждой строке числа должны быть разделены одним пробелом. **Порядок записи чисел в ответе менять нельзя**. Если вы не можете найти какое-то из чисел, вместо него запишите любое целое число.

Для выполнения расчетов можно использовать калькулятор или среду программирования.

Задача 2. Лягушонок Пепе

Лягушонок Пепе путешествует по виртуальному замку и собирает биткоиины. Схема виртуального замка и количество биткоионов в каждой комнате представлены на схеме:



От каждой комнаты, кроме крайних существуют только два пути: налево (**L**) и направо (**R**), двигаться обратно через двери запрещено. Лягушонок начинает свой путь в самой верхней комнате, обозначенной словом **start**, а заканчивает — в комнате, обозначенной словом **finish**. Во всех комнатах, кроме начальной и конечной, лежат биткоиины. Путешествуя по комнатам, лягушонок забирает себе все биткоиины из каждой комнаты, которую посещает. Количество биткоионов указано на рисунке выше.

Вам нужно определить и записать в первой строке ответа максимальное количество биткоионов, которые может собрать лягушонок Пепе. Во второй строке необходимо записать маршрут, на котором он соберет эту сумму. Маршрут состоит из букв **L** и **R**, записанных без разделителей.

Например, если лягушонок сначала каждый раз будет входить в правую дверь (пока это возможно), а потом каждый раз будет входить в левую дверь, то ответом будет:

75

RRRRLLLL

Задача 3. Водолей

У исполнителя Водолей есть два сосуда: сосуд A объемом 3 литра и сосуд B объемом 7 литров. Также исполнитель Водолей умеет выполнять команды:

- наполнить сосуд A — сосуд наполняется водой до полного заполнения,
- наполнить сосуд B — сосуд наполняется водой до полного заполнения,
- опустошить сосуд A — вся вода из сосуда A выливается,
- опустошить сосуд B — вся вода из сосуда B выливается,
- перелить в B — вода из A переливается в B , пока не выльется полностью либо пока в B есть место; оставшаяся вода остается в сосуде A ,
- перелить в A — вода из B переливается в A , пока не выльется полностью либо пока в A есть место; оставшаяся вода остается в сосуде B .

У вас есть алгоритм для исполнителя Водолей из 10 команд, в которой не проставлены названия сосудов.

1. наполнить ...
2. перелить в ...
3. опустошить ...
4. наполнить ...
5. перелить в ...
6. опустошить ...
7. перелить в ...
8. опустошить ...
9. наполнить ...
10. перелить в ...

Вы не можете влиять на последовательность команд, но можете менять названия сосудов, задавая соответственно A — для первого сосуда и B — для второго.

Напишите последовательность названий сосудов, которые нужно подставить в указанный выше список команд, чтобы получить необходимое количество литров в **каждом** из представленных ниже случаев (в начальный момент выполнения алгоритма сосуда пусты):

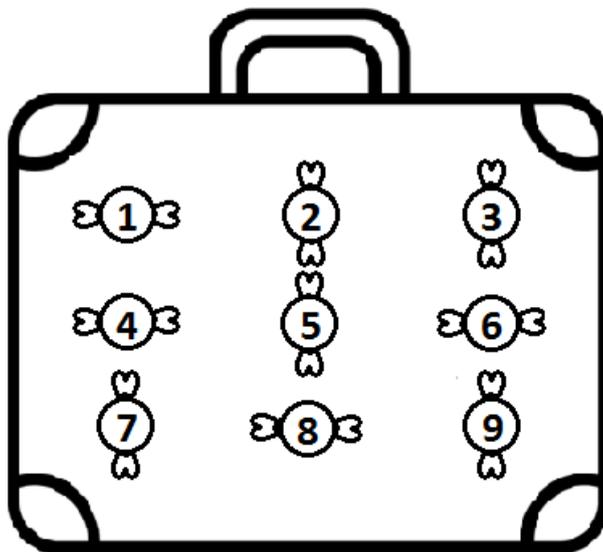
- $A = 0, B = 6$,
- $A = 3, B = 1$,
- $A = 0, B = 4$,
- $A = 3, B = 6$,
- $A = 2, B = 7$.

Для **каждой пары значений** вам необходимо в ответе записать **одну строку**, состоящую из **латинских** символов A и B , которые будут подставлены в приведенный выше алгоритм. В ответе должно быть ровно пять строк, состоящих из последовательностей символов A и B без разделителей. Длина корректной последовательности должна быть равна 10 символам. Если вы не смогли составить алгоритм для какого-то из случаев, то напечатайте в этой строке единственный символ A .

Например, если вам нужно получить в конце работы алгоритма в сосуде A 3 литра и в сосуде B 3 литра, то подойдет такая последовательность: $AAAAABVAAA$.

Задача 4. Тайна Карбофоса

Все секреты известного международного преступника и контрабандиста Карбофоса хранятся в его чемодане. Чемодан оборудован специальным замком, состоящим из девяти ручек, каждая из которых может быть либо в горизонтальном, либо в вертикальном положении.



Для того, чтобы открыть чемодан, нужно привести все ручки в горизонтальное положение, но сделать это не так просто как кажется: каждая из ручек при повороте также поворачивает несколько других.

Двум известным сыщикам, действующим под кодовыми именами «Шеф» и «Коллега», удалось найти схему, определяющую связь ручек:

1. 6 9
2. 3 8
3. 6 7 9
4. 2 3
5. 1 4 6
6. 2 7
7. 4 5 8
8. 3 9
9. 1 4

Таким образом, поворот ручки с номером 1 автоматически повернет ручки 6 и 9, при этом дальнейшие (рекурсивные) повороты ручек, зависящих от 6 и 9, не выполняются. Аналогично поворот ручки с номером 2 приведет к повороту только ручек 3 и 8 и так далее. Начальное положение ручек представлено на рисунке: ручки с номерами 2, 3, 5, 7, 9 находятся в вертикальном положении.

В ответе запишите последовательность номеров ручек, которые нужно повернуть, чтобы **наибольшее** количество ручек на чемодане оказались в горизонтальном положении. При этом ваша последовательность должна содержать **как можно меньше** чисел. Числа в последовательности разделяйте пробелами.

Задача 5. Фермер

Ограничение по времени: 1 секунда

Год выдался урожайным, и фермер Купер собрал N килограмм кукурузы со своих полей.

Весь урожай он продает своему старому другу Джону. Они договорились, что Джон возьмет у Купера всю его кукурузу по цене A рублей за килограмм с условием, что каждый K -й килограмм идет бесплатно.

Для данных значений N , A , K , посчитайте сколько выручит фермер Купер с продажи всей своей кукурузы.

Формат входных данных

Первая строка входных данных содержит значение N — количество килограмм кукурузы, которую собрал Купер ($1 \leq N \leq 2 \cdot 10^{10}$).

Во второй строке дано целое число A — цена килограмма кукурузы, по которой ее будет покупать Джон ($1 \leq A \leq 10$).

В третьей строке дано целое число K — номер каждого килограмма, который будет отдан Джону бесплатно ($2 \leq K \leq 1000$).

Обратите внимание, что значение N может быть больше, чем возможное значение 32-битной целочисленной переменной, поэтому необходимо использовать 64-битные числа (тип `int64` в языке Pascal, тип `long long` в C и C++, тип `long` в Java и C#).

Формат выходных данных

Программа должна вывести одно целое число — сколько выручит фермер Купер с продажи всей своей кукурузы.

Система оценки

Решения, правильно работающие при $N \leq 100$, будут оцениваться в 50 баллов.

Примеры

стандартный ввод	стандартный вывод
10 2 5	16
10 3 11	30

Замечание

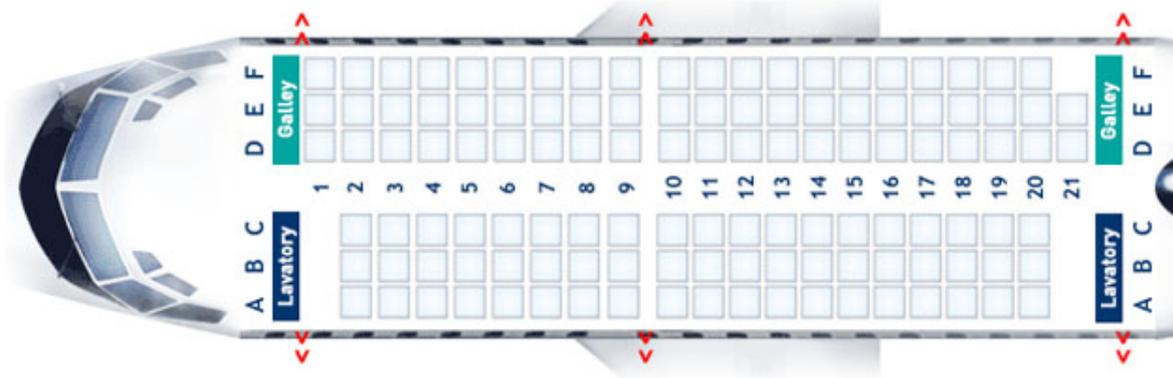
В первом примере фермер продаёт 10 килограмм кукурузы по 2 рубля, при этом каждый 5-й килограмм (то есть 5-й и 10-й) отдаются бесплатно. Будет оплачено 8 килограмм по 2 рубля, всего 16 рублей.

Во втором примере бесплатно идёт каждый 11-й килограмм, поэтому все 10 килограмм будут оплачены по 3 рубля, всего 30 рублей.

Задача 6. Боинг

Ограничение по времени: 1 секунда

Авиакомпания **A5** провела очередное обновление программного обеспечения. Теперь продажа билетов возможна только на первое незанятое место в самолете. На рисунке ниже представлена схема салона самолета **Boeing 737–600** с компоновкой **125E**, используемой этой авиакомпанией.



Номер места в салоне является комбинацией номера ряда (число от 1 до 21) и номера места в ряду (одна из букв «A», «B», «C», «D», «E», «F»). При этом в первом ряду только три места: **1D**, **1E** и **1F**, а в последнем ряду — два места: **21D** и **21E**.

Первым всегда продается место с номером **1D**, затем — место с номером **1E**, третьим — **1F**, четвертым — **2A** и т.д.

Напишите программу, которая по количеству проданных билетов определяет и выводит на экран номер первого свободного места или слово **full**, если все билеты проданы.

Формат входных данных

Программа получает на вход одно целое неотрицательное число n — количество проданных билетов ($0 \leq n \leq 119$).

Формат выходных данных

Выведите номер первого свободного места или слово **full**, если все билеты проданы.

Примеры

стандартный ввод	стандартный вывод
4	2B
100	18B

Задача 7. Интересное подмножество

Ограничение по времени: 1 секунда

Дано множество натуральных чисел (все элементы множества попарно различны), упорядоченное по возрастанию значений. *Интересным* подмножеством исходного множества будем называть такое подмножество (возможно, полностью совпадающее с исходным множеством), что каждый его элемент больше мощности этого подмножества. Мощностью подмножества называется количество элементов в нем.

Для данного множества необходимо найти размер наибольшего *интересного* подмножества, составленного из элементов этого множества.

Формат входных данных

Первая строка входных данных содержит целое число N — количество элементов исходного множества ($1 \leq N \leq 10^5$).

В следующих N строках записаны натуральные числа a_i по одному в строке — элементы исходного множества ($1 \leq a_i \leq 2 \cdot 10^9$), упорядоченные по возрастанию значений.

Формат выходных данных

Программа должна вывести одно целое число — размер наибольшего интересного подмножества.

Система оценки

Решения, правильно работающие при $N = 5$, будут оцениваться в 20 баллов.

Решения, правильно работающие при $N \leq 12$, будут оцениваться в 60 баллов.

Пример

стандартный ввод	стандартный вывод
5	2
1	
2	
3	
4	
5	

Замечание

В примере из условия в множестве пять чисел: 1, 2, 3, 4, 5. В качестве интересного подмножества можно взять, например, подмножество $\{3, 5\}$. Его мощность равна 2 и все элементы этого подмножества больше 2. Интересного подмножества большего размера в данном примере не существует.