

Лабораторная работа состоит из двух частей: задание №1 и задание №2

Перед кодом программы размещается текст задания(комментарий). Затем приводится набор тестов. Данные для тестов размещаются в отдельных файлах(*.txt). Код программы должен содержать комментарии, поясняющие основные части алгоритма. Код должен быть структурирован.

Лабораторная работа № 2

Задание #1

Цель работы. Решить задачу с использованием жадного алгоритма.

Варианты:

1.

Дана лекционная аудитория, в которой несколько профессоров хотят прочесть свои лекции. Для составления расписания профессора подали заявки, вида $[s_i, f_i)$ – время начала и конца лекции. Лекция считается открытым интервалом, то есть какая-то лекция может начаться в момент окончания другой, без перерыва. Составьте расписание занятий так, чтобы выполнить максимальное количество заявок.

Входные данные

В первой строке вводится натуральное число N , не более 1000 – общее количество заявок. Затем вводится N строк с описаниями заявок - по два числа в каждом s_i и f_i .

Гарантируется, что $s_i < f_i$. Время начала и окончания лекции – натуральное число, не превышает 1440 (в минутах с начала суток :)

Выходные данные

Выведите одно число – максимальное количество заявок, которые можно выполнить.

Пояснения к примерам

Во втором примере можно выполнить вторую и третью заявки.

Примеры

входные данные

1
5 10

выходные данные

1

входные данные

3
1 5
2 3
3 4

выходные данные

2

Мальчик Костя очень любит конфеты, но мама не разрешает ему брать их слишком много. Поэтому каждый раз, когда Костя хочет съесть конфету, мама предлагает ему сыграть в игру.

Брать конфеты можно в произвольном порядке, но одну и ту же можно брать не более одного раза.

Помогите Косте взять как можно больше конфет (вне зависимости от финального *уровня вредности*).

В первой строке входных данных записано целое число N ($1 \leq N \leq 1000$) – количество видов конфет. Во второй строке записаны N целых чисел A_i ($1 \leq A_i \leq 10^6$). В третьей строке записаны N целых чисел C_i ($1 \leq C_i \leq 10^9$).

В единственной строке выведите целое число, равное максимальному количеству конфет, которые может взять Костя.

Новый интернет-провайдер предоставляет услугу доступа в интернет с посекундной тарификацией. Для подключения нужно купить карточку, позволяющую пользоваться интернетом определенное количество секунд. При этом компания продает карточки стоимостью 1, 2, 4, ..., 2^{30} рублей на $a_0, a_1, a_2, \dots, a_{30}$ секунд соответственно.

Родители разрешили Пете пользоваться интернетом M секунд. Определите, за какую наименьшую сумму он сможет купить карточки, которые позволят ему пользоваться интернетом **не менее** M секунд. Естественно, что Петя может купить как карточки различного достоинства, так и несколько карточек одного достоинства.

В первой строке содержится единственное натуральное число M ($1 \leq M \leq 10^9$). Во второй строке задаются натуральные числа a_0, a_1, \dots, a_{30} , не превосходящие 10^9 .

Выведите единственное число – наименьшую сумму денег, которую Пете придется потратить.

```
11  
1 1 10 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1
```

5

После затянувшегося совещания директор фирмы решил заказать такси, чтобы развезти сотрудников по домам. Он заказал N машин — ровно столько, сколь у него сотрудников. Однако когда они подъехали, оказалось, что у каждого водителя такси свой тариф за 1 километр.

Директор знает, какому сотруднику сколько километров от работы до дома (к сожалению, все сотрудники живут в разных направлениях, поэтому нельзя отправить двух сотрудников на одной машине). Теперь директор хочет определить, какой из сотрудников на каком такси должен поехать домой, чтобы суммарные затраты на такси (а их несет фирма) были минимальны.

Входные данные

Сначала во входном файле записано натуральное число N ($1 \leq N \leq 1000$) — количество сотрудников компании (совпадающее с количеством вызванных машин такси). Далее записано N чисел, задающих расстояния в километрах от работы до домов сотрудников компании (первое число — для первого сотрудника, второе — для второго и т.д.). Все расстояния — положительные целые числа, не превышающие 1000. Далее записано еще N чисел — тарифы за проезд одного километра в такси (первое число — в первой машине такси, второе — во второй и т.д.). Тарифы выражаются положительными целыми числами, не превышающими 10000.

Выходные данные

В выходной файл выведите N чисел. Первое число — номер такси, в которое должен сесть первый сотрудник, второе число — номер такси, в которое должен сесть второй и т.д., чтобы суммарные затраты на такси были минимальны. Если вариантов рассадки сотрудников, при которых затраты минимальны, несколько, выведите любой из них.

Примеры

входные данные

```
3
10 20 30
50 20 30
```

выходные данные

```
1 3 2
```

входные данные

```
5
10 20 1 30 30
3 3 3 2 3
```

выходные данные

```
5 1 3 2 4
```

5.

Тетя Люба только что постирала все белье, и теперь перед ней стоит непростая задача: как его высушить, чтобы ни одна вещь не успела испортиться. Сразу после стирки i -я постиранная вещь имеет влажность w_i . Если она сушится на веревке, то за минуту ее влажность уменьшается на 1, а если на батарее — то на r (если влажность была меньше r , то она становится равной 0). Причем веревок у тети Любы много (хватает для одновременной сушки всех вещей), а батарея только одна, причем такая маленькая, что на ней нельзя сушить две вещи одновременно. i -я вещь испортится, если не высохнет за время d_i . Помогите тете Любе составить план, когда какую вещь повесить на батарею.

Входные данные

Первая строка входного файла содержит целые числа n ($1 \leq n \leq 10^5$) (количество мокрых вещей) и r ($1 \leq r \leq 10^9$). Следующие n строк содержат описания постиранных вещей — пары чисел w_i и d_i ($1 \leq w_i \cdot d_i \leq 10^9$).

Выходные данные

Выведите план сушки в виде пар целых чисел t_i и k_i , где t_i — измеренное в минутах время от начала сушки, а k_i — номер вещи, которую нужно повесить на батарею в этот момент.

Выводите пары в порядке увеличения ti . Пар не должно быть больше 100000. Не выводите числа больше 10^9 .

Если высушить все вещи невозможно, выведите слово **Impossible**.

Примеры

входные данные

```
3 3
2000 1000
2000 2000
2500 1500
```

выходные данные

```
0 1
500 3
```

входные данные

```
3 3
2000 1000
2000 1000
2000 1000
```

выходные данные

```
Impossible
```

6.

Толик придумал новую технологию программирования. Он хочет уговорить друзей использовать ее. Однако все не так просто. i -й друг согласится использовать технологию Толика, если его авторитет будет не меньше a_i (авторитет выражается целым числом). Как только i -й друг начнет ее использовать, к авторитету Толика прибавится число b_i (попадаются люди, у которых $b_i < 0$). Помогите Толику наставить на путь истинный как можно больше своих друзей.

Входные данные

На первой строке содержатся два целых числа: Количество друзей у Толика n и первоначальный авторитет Толика a_0 ($-10^9 \leq a_0 \leq 10^9$). Следующие n строк содержат пары целых чисел a_i и b_i ($-10^9 \leq a_i, b_i \leq 10^9$).

Выходные данные

На первой строке выведите число m — максимальное число друзей, которых может увлечь Толик. На второй строке выведите m чисел — номера друзей в том порядке, в котором их нужно агитировать.

Примеры

№	INPUT.TXT	OUTPUT.TXT
1	5 1 1 3 6 -5 6 -4 2 2 2 -1	4 1 4 3 5

7.

Андрей едет из пункта А в пункт В на автомобиле. Расстояние между этими пунктами равно N километров. Известно, что с полным баком автомобиль способен проехать k километров. Дана карта, на которой отмечены координаты бензоколонок, относительно пункта А. Определите минимальное число заправок, которые придется сделать Андрею чтобы успешно достичь пункта В. Известно, что при выезде из пункта А бак был полон.

Входные данные

В первой строке вводятся числа N и k (натуральные, не превосходят 1000). В следующей строке вводится количество бензоколонок S , потом следует S натуральных чисел, не превосходящих N — расстояния от пункта А до каждой заправки. *Заправки упорядочены по удаленности от пункта А.*

Выходные данные

Если при данных условиях пункта В достичь невозможно, то вывести число -1. Если решение существует, то вывести минимальное количество остановок на дозаправку, которое нужно чтобы достичь пункта В.

Примеры

входные данные

100 20

1 50

выходные данные

-1

входные данные

100 50

1 50

выходные данные

1

входные данные

100 100

3 10 20 80

выходные данные

0

8.

У Васи в комнате очень много коробок, которые валяются в разных местах. Васина мама хочет, чтобы он прибрался. Свободного места в комнате мало и поэтому Вася решил собрать все коробки и составить их одну на другую.

К сожалению, это может быть невозможно. Например, если на картонную коробку с елочными украшениями положить что-то железное и тяжелое, то вероятно следующий Новый год придется встречать с новыми игрушками.

Вася взвесил каждую коробку и оценил максимальный вес который она может выдержать. Помогите ему определить какое наибольшее количество коробок m он сможет составить одну на другую так, чтобы для каждой коробки было верно, что суммарный вес коробок сверху не превышает максимальный вес, который она может выдержать.

Входные данные

Первая строка входного файла содержит целое число n ($1 \leq n \leq 10^5$) - количество коробок в комнате. Каждая следующая из n строк содержит два целых числа w_i и c_i ($1 \leq w_i \leq 10^5, 1 \leq c_i \leq 10^9$), где w_i — это вес коробки с номером i , а c_i — это вес который она может выдержать.

Выходные данные

В выходной файл выведите одно число — ответ на задачу.

Примеры

входные данные

3

10 11

20 100

30 10

выходные данные

3

входные данные

3

выходные данные

1

входные данные

3

11 11

20 100

30 10

выходные данные

2

9.

Даны два вектора: $v_1=(x_1, x_2, \dots, x_n)$ и $v_2=(y_1, y_2, \dots, y_n)$. Скалярным произведением этих векторов называется значение, вычисляемое по формуле: $x_1 y_1 + x_2 y_2 + \dots + x_n y_n$.

Разрешено переставлять координаты каждого из векторов любым образом. Выберите такие их перестановки, чтобы скалярное произведение двух полученных векторов было минимальным и выведите его значение. $1 \leq n \leq 800$. $-100000 \leq x_i, y_i \leq 100000$

Входные данные

Первая строка входного файла содержит единственное целое число t - количество наборов тестовых данных. Далее следуют сами наборы по три строки в каждом. Первая строка тестового набора содержит единственное целое число n . Две следующие строки содержат по n целых чисел, задающих координаты соответствующего вектора, каждая.

Выходные данные

Для каждого набора выведите строку с номером этого набора и ответом на задачу - значением минимального скалярного произведения. Следуйте формату, указанному в примере.

Примеры

входные данные

2

3

1 3 -5

-2 4 1

5

1 2 3 4 5

1 0 1 0 1

выходные данные

Case #1: -25

Case #2: 6

10.

Вася написал на длинной полоске бумаги большое число и решил похвастаться своему старшему брату Пете этим достижением. Но только он вышел из комнаты, чтобы позвать брата, как его сестра Катя вбежала в комнату и разрежала полоску бумаги на несколько частей. В результате на каждой части оказалось одна или несколько идущих подряд цифр.

Теперь Вася не может вспомнить, какое именно число он написал. Только помнит, что оно было очень большое. Чтобы утешить младшего брата, Петя решил выяснить, какое максимальное число могло быть написано на полоске бумаги перед разрезанием. Помогите ему!

Входные данные

Входной файл содержит одну или более строк, каждая из которых содержит последовательность цифр. Количество строк во входном файле не превышает 100, каждая строка содержит от 1 до 100 цифр. Гарантируется, что хотя бы в одной строке первая цифра отлична от нуля.

Выходные данные

Выведите в выходной файл одну строку - максимальное число, которое могло быть написано на полоске перед разрезанием.

Примеры

входные данные

2
20
004
66

выходные данные

66220004

входные данные

3

выходные данные

3

11.

В этой задаче требуется найти разбиение натурального числа n на как можно большее количество различных натуральных слагаемых.

Входные данные

В первой строке входного файла записано натуральное число n ($1 \leq n \leq 1\,000\,000\,000$)

Выходные данные

В первой строке выходного файла выведите k – количество слагаемых. Во второй строке выведите k чисел через пробел – сами слагаемые. Если разбиений с максимальным количеством слагаемых несколько, можно вывести любое из них. Слагаемые можно выводить в любом порядке.

Примеры

входные данные

4

выходные данные

2
1 3

входные данные

6

выходные данные

3
1 2 3

12.

«Ну не гномы, а наказание какое-то!», – подумала Белоснежка, в очередной раз пытаясь уложить гномов спать. Одного уложишь – другой уже проснулся! И так всю ночь.

У Белоснежки n гномов, и все они очень разные. Она знает, что для того, чтобы уложить спать i -го гнома нужно a_i минут, и после этого он будет спать ровно b_i минут. Помогите Белоснежке узнать, может ли она получить хотя бы минутку отдыха, когда все гномы будут спать, и если да, то в каком порядке для этого нужно укладывать гномов спать.

Например, пусть есть всего два гнома, $a_1 = 1$, $b_1 = 10$, $a_2 = 10$, $b_2 = 20$. Если Белоснежка сначала начнет укладывать первого гнома, то потом ей потребуется целых 10 минут, чтобы уложить второго, а за это время проснется первый. Если же она начнет со второго гнома, то затем она успеет уложить первого и получит целых 9 минут отдыха.

Входные данные

Первая строка входного файла содержит число n ($1 \leq n \leq 10^5$), вторая строка содержит числа a_1, a_2, \dots, a_n , третья – числа b_1, b_2, \dots, b_n ($1 \leq a_i, b_i \leq 10^9$).

Выходные данные

Выведите в выходной файл n чисел – порядок, в котором нужно укладывать гномов спать. Если Белоснежке отдохнуть не удастся, выведите число -1.

Примеры

входные данные

```
2
1 10
10 20
```

выходные данные

```
2 1
```

входные данные

```
2
10 10
10 10
```

выходные данные

```
-1
```

13.

Саша устал играть со своими палочками и нашел более интересное занятие. Он написал на доске все числа от 1 до N в одну строчку (так он получил очень большое число) и теперь хочет стереть ровно M цифр таким образом, чтобы оставшееся написанное число было как можно меньше. Помогите ему!

Входные данные

Входной файл содержит два целых числа M и N ($1 \leq M \leq 30000$, $1 \leq N \leq 10000$). Входные данные корректны, поэтому хотя бы одна цифра будет содержаться в выходных данных.

Выходные данные

Выходной файл должен содержать минимальное число, которое Саша может получить на доске, вычеркнув M цифр. Лидирующие нули допустимы.

Примеры

входные данные

```
12
20
```

выходные данные

```
0111114151617181920
```

14

Мальчик Петя решил приготовить маме подарок на день рождения — праздничный завтрак. Он решил сделать вкусный чай и испечь блинчики. К сожалению, не отличаясь выдающимися кулинарными способностями, Петя не смог уследить за блинчиками. Каждый из них получился подгорелым с одной стороны и недожаренным с другой. В результате у Пети получилось N черно-белых блинчиков. Все блинчики он выложил на большую тарелку друг на друга. Теперь Петя хочет перевернуть их так, чтобы все они лежали светлой стороной вверх — Петя думает, что так они маме понравятся больше. Для переворачивания блинчиков у него есть лопаточка, которой он может взять несколько верхних блинчиков (от одного до всей стопки) и перевернуть их все вместе (таким образом, что верхний блин окажется на месте нижнего из взятых блинов).

За какое минимальное число таких действий Петя может перевернуть все блины светлой стороной вверх?

Входные данные

В первой строке входного файла дано число N ($1 \leq N \leq 100\,000$) — количество блинчиков. Далее в N строках описываются блинчики, сверху вниз. Если в i -й строке стоит символ W ,

то i -й блинчик лежит недожаренной стороной вверх, а если В, то подгоревшей стороной вверх.

Выходные данные

В выходной файл выведите единственное число — количество переворачиваний, которое должен сделать Петя, чтобы положить все блинчики недожаренной стороной вверх.

Примеры

входные данные

```
6
W
B
B
B
W
B
```

выходные данные

```
4
```

15.

Теплым весенним днем группа из N школьников-программистов гуляла в окрестностях города Кисловодска. К несчастью, они набрали на большую и довольно глубокую яму. Как это случилось — непонятно, но вся компания оказалась в этой яме.

Глубина ямы равна H . Каждый школьник знает свой рост по плечи h_i и длину своих рук l_i . Таким образом, если он, стоя на дне ямы, поднимет руки, то его ладони окажутся на высоте $h_i + l_i$ от уровня дна ямы. Школьники могут, вставая друг другу на плечи, образовывать вертикальную колонну. При этом любой школьник может встать на плечи любого другого школьника. Если под школьником i стоят школьники j_1, j_2, \dots, j_k , то он может дотянуться до уровня $h_{j_1} + h_{j_2} + \dots + h_{j_k} + h_i + l_i$.

Если школьник может дотянуться до края ямы (то есть $h_{j_1} + h_{j_2} + \dots + h_{j_k} + h_i + l_i \geq H$), то он может выбраться из нее. Выбравшиеся из ямы школьники не могут помочь оставшимся.

Найдите наибольшее количество школьников, которые смогут выбраться из ямы до прибытия помощи, и перечислите их номера.

Входные данные

В первой строке входного файла записано натуральное число N — количество школьников, попавших в яму.

Далее в N строках указаны по два целых числа: рост i -го школьника по плечи h_i и длина его рук l_i .

В последней строке указано целое число — глубина ямы H .

Выходные данные

В первой строке выведите K — максимальное количество школьников, которые смогут выбраться из ямы. Если $K > 0$, то во второй строке выведите их номера в том порядке, в котором они вылезают из ямы. Школьники нумеруются с единицы в том порядке, в котором они заданы во входном файле. Если существует несколько решений, выведите любое.

Примеры

входные данные

```
2
10 4
5 2
20
```

выходные данные

```
0
```

входные данные

```
6
6 7
```

3 1
8 5
8 5
4 2
10 5
30

выходные данные

4
1 4 2 5

16.

Найдите количество различных разбиений натурального числа n на натуральные слагаемые таких, что для любых двух различных чисел $a \neq b$, входящих в разбиение, верно, что количество чисел a и b в разбиении различны. Разбиение, отличающиеся только порядком слагаемых, различными не считаются.

Например, если $n=4$, то из пяти возможных разбиений этому условию удовлетворяют все, кроме разбиения на слагаемые 1 и 3: в этом разбиении количество единиц равно количеству троек.

$4 = 1 + 1 + 1 + 1$	4 единицы
$4 = 1 + 1 + 2$	3 единицы, 1 тройка
$4 = 1 + 3$	1 единица и 1 тройка!
$4 = 2 + 2$	2 двойки
$4 = 4$	1 четверка

Входные данные

В первой строке входного файла записано натуральное число n ($1 \leq n \leq 100$).

Выходные данные

В первой строке входного файла выведите количество разбиений числа n , удовлетворяющих заданным ограничениям.

Примеры

входные данные

4

выходные данные

4

входные данные

6

выходные данные

7

17.

В некоей воинской части есть сапожник. Рабочий день сапожника длится N минут. Заведующий складом оценивает работу сапожника по количеству починенной обуви, независимо от того, насколько сложный ремонт требовался в каждом случае. Дано k сапог, нуждающихся в починке. Определите какое максимальное количество из них сапожник сможет починить за один рабочий день.

Входные данные

В первой строке вводятся число N (натуральное, не превышает 1000), и число k (натуральное, не превышает 500). Затем идет k чисел – количество минут, которые требуются чтобы починить i -й сапог (времена – натуральные числа, не превосходят 100).

Выходные данные

Выведите единственное число – максимальное количество сапог, которые можно починить за один рабочий день.

Пояснение к примерам

В первом примере можно починить либо первый и второй, либо второй и третий сапоги.

Примеры

входные данные

10 3

6 2 8

выходные данные

2

входные данные

3 2

10 20

выходные данные

0

входные данные

100 4

2 6 7 8

выходные данные

4

18.

В игре Cookie Clicker игрок зарабатывает печеньки (cookies), щелкая мышкой по изображению большой печеньки. Тратя заработанные печенье, игрок может покупать различные усовершенствования (ферму, фабрику и т. д.), которые также производят дополнительные печенье.

Рассмотрим упрощенный вариант этой игры. Пусть игрок может сделать один щелчок мышкой в секунду, что приносит ему одну печеньку. Также в любой момент времени игрок может потратить C печенек на покупку фабрики (при этом у игрока должно быть не меньше C печенек, после покупки фабрики количество его печенек моментально уменьшается на C). Каждая купленная фабрика увеличивает ежесекундное производство печенек на P штук (то есть если у игрока одна фабрика, то он получает $1 + P$ печенек в секунду, две фабрики — $1 + 2P$ печенек, три фабрики — $1 + 3P$ печенек и т. д.). Игрок может приобрести неограниченное число фабрик стоимостью C печенек каждая. Фабрика начинает производить дополнительные печенье сразу же, например, если после какой-то секунды игры у игрока стало C печенек, то игрок может купить фабрику и уже на следующей секунде его производство печенек увеличится на P штук.

Оригинальная игра никогда не заканчивается, но мы будем считать, что целью игры является набрать N печенек. Определите минимальное время, за которое может быть достигнута цель игры.

Входные данные

Программа получает на вход три целых положительных числа, записанных в отдельных строках: C (стоимость фабрики), P (производительность одной фабрики) и N (необходимое количество печенек).

Выходные данные

Программа должна вывести одно целое число — минимальное время в секундах, за которое игрок может получить не менее N печенек.

Примечание

В первом тесте: через 50 секунд после начала игры у игрока будет 50 печенек, и он сможет купить фабрику. После этого он будет получать 4 печеньки в секунду, и на производство 100 печенек понадобится еще 25 секунд.

Во втором тесте: игрок сможет набрать 100 печенек за 100 секунд, при этом фабрику покупать нет смысла.

Примеры

входные данные

50
3
100

выходные данные

75

входные данные

99
10
100

выходные данные

100

19.

Длина автомобильной дороги составляет N километров. Часть дороги необходимо отремонтировать. При обследовании дорога была разбита на N участков длиной 1 километр, и для каждого участка было определено, нуждается ли он в ремонте или нет, после чего был составлен план дороги, на котором отмечены участки, нуждающиеся в ремонте.

Для ремонта дороги можно привлечь несколько компаний-подрядчиков. Каждая компания может отремонтировать только непрерывный фрагмент дороги. При этом из-за требований антимонопольного законодательства длина фрагмента дороги, который ремонтирует одна компания, не должна превышать L километров (даже если на фрагменте, который ремонтирует одна компания, есть не нуждающиеся в ремонте участки, общая длина данного фрагмента не должна превышать L километров).

Определите, какое наименьшее количество компаний-подрядчиков необходимо привлечь для ремонта дороги.

Входные данные

Первая строка входных данных содержит целое число L ($L > 0$) — максимальную длину фрагмента дороги, который может отремонтировать одна компания. Во второй строке входных данных записано целое число N ($N > 0$) — длина всей дороги. Следующие N строк содержат по одному числу, равному 0 или 1. Число 1 обозначает, что соответствующий участок дороги нуждается в ремонте, число 0 — что участок не требует ремонта.

Выходные данные

Программа должна вывести одно целое число — минимальное количество компаний-подрядчиков, которое необходимо привлечь для ремонта дороги.

Примечание

В тесте из примера первая компания может отремонтировать участок номер 3, вторая компания — участки с 5 по 7.

Примеры

входные данные

3
8
0
0
1

0
1
0
1
0

выходные данные

2

20.

У мальчика Вити скоро день рождения, который он хочет провести с друзьями. Какой же праздник без праздничной олимпиады? Для своих друзей Витя подготовил олимпиаду по программированию.

Когда все гости собрались, именинник распределил их по m командам. Витя очень старался сбалансировать силы команд, и у него это получилось. Каждую из предложенных n задач олимпиады любая команда решает ровно за a_i минут и всегда с первой попытки. Ребята не любят распыляться, поэтому если команда берется писать задачу i , то все ее участники непрерывно решают задачу в течении a_i минут. При этом команда может начинать решать следующую задачу сразу после того, как решила предыдущую.

Однако, Витя не хотел поссорить своих друзей, поэтому исключил элемент борьбы между командами — команды не соперничают, а помогают друг другу. Целью команд на олимпиаде является решение всех задач. Таким образом каждую задачу решает только одна команда. При этом друзьям засчитывается штрафное время, равное числу минут, прошедших от начала олимпиады до момента сдачи задачи. Единственная цель, которую преследуют друзья — сдать все задачи с наименьшим суммарным штрафным временем.

Например, пусть участвуют две команды, которым предложено в олимпиаде три задачи с временами решения 5, 10 и 15 минут. Пусть первая команда решает сначала вторую, а потом первую задачу. Таким образом за решение второй задачи друзья получают 10 штрафных минут, а за решение первой 15. Вторая команда с начала олимпиады решает только третью задачу, за которую друзья получают 15 штрафных минут. В сумме друзья Вити получают 40 штрафных минут.

В то время, когда друзья будут решать задачи, Витя будет управлять порядком, в котором команды будут их решать. Помогите Вите — напишите программу, которая вычислит наименьшее суммарное штрафное время, требуемое для сдачи всех задач.

Входные данные

В первой строке входного файла содержатся два числа n и m — число задач и команд, соответственно ($0 \leq n \leq 50000$, $1 \leq m \leq 10000$). Во второй строке содержится n целых чисел a_i — количество времени, необходимое для решения задачи i любой из команд ($0 \leq a_i \leq 30$).

Выходные данные

Выведите в выходной файл наименьшее суммарное штрафное время, которое могут получить друзья Вити за решение всех задач.

Примеры

входные данные

3 2
10 15 5

выходные данные

35

входные данные

21.

На межрегиональной олимпиаде по программированию роботов соревнования проводятся в один тур и в необычном формате. Задачи участникам раздаются последовательно, а не все в самом начале тура, и каждая i -я задача ($1 \leq i \leq n$) становится доступной участникам в свой момент времени s_i . При поступлении очередной задачи каждый участник должен сразу определить, будет он ее решать или нет. В случае, если он выбирает для решения эту задачу, то у него есть t_i минут на то, чтобы сдать ее решение на проверку, причем в течение этого времени он не может переключиться на решение другой задачи. Если же участник отказывается от решения этой задачи, то в будущем он не может к ней вернуться. В тот момент, когда закончилось время, отведенное на задачу, которую решает участник, он может начать решать другую задачу, ставшую доступной в этот же момент, если такая задача есть, или ждать появления другой задачи. При этом за правильное решение i -й задачи участник получает c_i баллов.

Артур, представляющий на межрегиональной олимпиаде один из региональных центров искусственного интеллекта, понимает, что важную роль на такой олимпиаде играет не только умение решать задачи, но и правильный стратегический расчет того, какие задачи надо решать, а какие пропустить. Ему, как и всем участникам, до начала тура известно, в какой момент времени каждая задача станет доступной, сколько времени будет отведено на ее решение и сколько баллов можно получить за ее решение. Артур является талантливым школьником и поэтому сможет успешно решить за отведенное время и сдать на проверку любую задачу, которую он выберет для решения на олимпиаде.

Требуется написать программу, которая определяет, какое максимальное количество баллов Артур сможет получить при оптимальном выборе задач, которые он будет решать, а также количество и перечень таких задач.

Формат входного файла

Первая строка входного файла содержит одно целое число n ($1 \leq n \leq 105$) количество задач на олимпиаде.

Последующие n строк содержат описания задач, по три числа на каждой строке: s_i момент появления i -й задачи в минутах, t_i время, отведенное на ее решение в минутах, и c_i сколько баллов получит участник за решение этой задачи ($1 \leq s_i, t_i, c_i \leq 109$).

Формат выходного файла

Первая строка выходного файла должна содержать одно число – максимальное количество баллов, которое сможет получить Артур на олимпиаде.

Вторая строка должна содержать одно целое число m - количество задач, которые надо решить при оптимальном выборе.

Третья строка должна содержать m разделенных пробелом целых чисел - номера этих задач в порядке их решения. Задачи пронумерованы, начиная с единицы, в порядке их описания во входном файле.

Если оптимальных ответов несколько, необходимо вывести любой из них.

Пояснения к примерам

В первом примере Артур успевает решить все задачи и получить три балла.

Во втором примере Артуру выгоднее решать последнюю задачу и получить за нее три балла, чем решать только первые две и получить два балла.

Примеры

входные данные

```
2
1 1 1
2 2 2
```

выходные данные

```
3
2
1 2
```

22.

Всем известно, что основной целью любого крупного правительственного проекта во Флатландии является освоение бюджетных средств (разумеется, по назначению). В настоящее время во Флатландии ведется работа над m национальным проектами, i -й проект может освоить s_i миллионов Флатландских тугриков в день.

Правительство Флатландии планирует выделить n грантов для финансирования проектов, каждый по p миллионов Флатландских тугриков. Деньги i -го из грантов будут доступны для освоения, начиная с дня r_i . Когда очередной грант становится доступен для освоения, его можно отдать некоторому проекту. Этот проект осваивает деньги гранта в течение $\lceil p s_i \rceil$ дней. Проект не может одновременно осваивать деньги нескольких грантов.

Премьер-министр Флатландии господин Тупиков хочет выяснить, за какое время удастся освоить все деньги, выделенные в рамках грантов. Помогите ему выяснить самый ранний день, когда можно полностью освоить все деньги грантов.

Входные данные

Первая строка входного файла **"budget.in"** содержит числа m , n и p ($1 \leq m \leq 100$, $1 \leq n \leq 100$, $1 \leq p \leq 10^9$). Вторая строка содержит m целых чисел: s_1, s_2, \dots, s_m ($1 \leq s_i \leq 10^9$). Третья строка содержит n целых чисел: r_1, r_2, \dots, r_n ($1 \leq r_i \leq 10^9$).

Выходные данные

Первая строка выходного файла **"budget.out"** должна содержать одно целое число — минимальный день, к которому можно полностью освоить все деньги грантов.

Примечание

Одна из возможных оптимальных схем освоения устроена следующим образом: грант 1 отдается первому проекту, который осваивает его в течение 11 дней. Остальные гранты отдаются второму проекту, грант 2 осваивается в течение дней 3–7, грант 3 в течение дней 8–12 и грант 4 в течение дней 13–17. Заметим, что грант 4 появляется в день 12, но назначается только в день 13.

Примеры

входные данные

```
2 4 22
2 5
1 3 8 12
```

выходные данные

```
17
```

23.

Дорожка замощена плитками в один ряд, плитки пронумерованы числами от 1 до 1000. На плитках с номерами A , B и C ($A < B < C$) сидят три кузнечика, которые играют в чехарду по следующим правилам:

1. На одной плитке может находиться только один кузнечик.

2. За один ход один из двух крайних кузнечиков (то есть с плитки A или с плитки C) может перепрыгнуть через среднего кузнечика (плитка B) и встать на плитку, которая находится ровно посередине между двумя оставшимися кузнечиками (то есть между B и C или A и B соответственно). Если между двумя оставшимися кузнечиками находится чётное число плиток, то он может выбрать любую из двух центральных плиток.

Например, если кузнечики первоначально сидели на плитках номер 1, 5, 10, то первым ходом кузнечик с плитки номер 10 может перепрыгнуть на плитку номер 3 (она находится посередине между 1 и 5), или кузнечик с плитки номер 1 может перепрыгнуть на плитку номер 7 или 8 (эти две плитки находятся посередине между плитками 5 и 10).

Даны три числа: A, B, C . Определите, какое наибольшее число ходов может продолжаться игра.

Входные данные

Программа получает на вход три целых числа A, B и C ($1 \leq A < B < C \leq 1000$), записанных в отдельных строках.

Выходные данные

Выведите одно число — наибольшее количество ходов, которое может продолжаться игра.

Примечание к примеру

В примере сначала кузнечик с плитки №6 прыгает на плитку №3. Затем кузнечик с плитки №4 прыгает на плитку №2.

Примеры

входные данные

1
4
6

выходные данные

2

24.

Вы по-прежнему работаете под руководством д.б.н., проф. О.Б. Ломова и изучаете интеллект обезьян. Ваши подопечные уже очень далеко ушли от столь элементарной задачи, как сбор квадрата. Теперь вы работаете над тем, чтобы обучить их намного более сложной задаче. Вы по-прежнему даёте обезьянам набор из N палочек, но на этот раз вы хотите, чтобы они собрали из этих палочек треугольник.

Конечно, решить эту задачу в элементарном варианте — выбрать три палочки и собрать из них треугольник — ваши подопечные могут без каких-либо проблем; вы же хотите их обучить, чтобы они собирали один большой треугольник из **всех** выданных им палочек сразу. Таким образом, они должны разбить палочки на три группы так, чтобы, сложив палочки каждой группы в один большой отрезок, получить три отрезка, из которых можно собрать треугольник. Полученный треугольник должен быть невырожденным, т.е. его площадь должна быть строго больше нуля.

Как и в прошлый раз, вам понадобилась программа, которая определит, разрешима ли задача для данного набора палочек.

Входные данные

На первой строке входного файла находится одно натуральное число N — количество палочек в наборе ($1 \leq N \leq 16000$). На второй строке находятся N натуральных чисел — длины палочек. Гарантируется, что суммарная длина палочек не превосходит 100000000.

Выходные данные

Если решения не существует, то в первую строку выходного файла выведите одно слово “no” (без кавычек). В противном случае в первую строку выведите одно слово “yes”, а в следующие

три строки выведите какой-нибудь способ собрать треугольник из данных палочек. Каждая из этих трёх строк должна описывать очередную сторону получившегося треугольника: в каждой строке сначала должно идти количество палочек, из которых состоит эта сторона, а потом длины этих палочек. Каждую палочку, конечно, можно использовать только один раз.

Если есть несколько способов собрать треугольник из данных палочек, выведите любой.

Примеры

входные данные

```
5
1 2 3 4 5
```

выходные данные

```
yes
2 4 3
1 5
2 1 2
```

входные данные

```
5
1 2 3 4 100
```

выходные данные

```
no
```

25.

В городе π недавно построили парк аттракционов, в котором есть павильон игровых автоматов. Каждый из автоматов рассчитан на одного человека. В программе Всероссийской олимпиады планируется посещение этого павильона.

Перед организаторами встала сложная задача — составить расписание игры участников олимпиады на автоматах таким образом, чтобы каждый из N участников олимпиады смог поиграть на каждом из автоматов, и при этом автобус, увозящий участников из парка олимпиады, смог бы отправиться к месту проживания как можно раньше.

Время перемещения участников между автоматами, а также между автобусом и павильоном считается равным нулю. Каждый из участников в любой момент времени может как играть на автомате, так и ждать своей очереди, например, гуляя по парку. Для каждого из M ($M \leq N$) автоматов известно время игры на нём t_i ($1 \leq i \leq M$). Прервать начатую игру на автомате невозможно. Автобус привозит всех участников олимпиады в парк одновременно в нулевой момент времени.

Требуется написать программу, которая по заданным числам N , M и t_i определяет оптимальное расписание игры на автоматах для каждого из участников.

Входные данные

В первой строке входного файла содержатся два числа: N и M ($1 \leq M \leq N \leq 100$). Во второй строке заданы M целых чисел t_i ($1 \leq t_i \leq 100$), каждое из которых задаёт время игры на i -м автомате ($1 \leq i \leq M$). Числа в строке разделяются одиночными пробелами.

Выходные данные

В первой строке необходимо вывести одно число — минимально возможное время отправления автобуса из парка аттракционов. Далее необходимо вывести N расписаний игр на автоматах, по одному для каждого из участников. Каждое расписание описывается в $(M + 1)$ строках, первая из которых — пустая, а далее следуют M строк, описывающих автоматы в порядке их посещения этим участником. Посещение автомата описывается двумя целыми числами: номером автомата j ($1 \leq j \leq M$) и временем начала игры участника на этом автомате.

Примеры

входные данные

```
2 1
2
```

выходные данные

4

1 0

1 2

входные данные

3 2

2 1

выходные данные

6

1 0

2 2

1 2

2 4

2 0

1 4

26.

В этой задаче Вася готовится к олимпиаде. Учитель дал ему N ($1 \leq N \leq 100\,000$) задач для тренировки. Для каждой из этих задач известно, каким умением a_i нужно обладать для её решения. Это означает, что если текущее умение Васи больше либо равно заданного умения для задачи, то он может ее решить. Кроме того, после решения i -й задачи Васино умение увеличивается на число b_i .

Исходное умение Васи равно A . Решать данные учителем задачи он может в произвольном порядке. Какое максимальное количество задач он сможет решить, если выберет самый лучший порядок их решения?

Входные данные

Сначала вводятся два целых числа N, A ($1 \leq N \leq 100\,000, 0 \leq A \leq 10^9$) — количество задач и исходное умение. Далее идут N пар целых чисел a_i, b_i ($1 \leq a_i \leq 10^9, 1 \leq b_i \leq 10^9$) — соответственно сколько умения нужно для решения i -й задачи и сколько умения прибавится после её решения.

Выходные данные

Выведите одно число — максимальное количество задач, которое Вася сможет решить.

Примеры**входные данные**

3 2

3 1

2 1

1 1

выходные данные

3

27.

Завтра Петя уезжает в кругосветное путешествие, в процессе которого собирается посетить N разных городов. Вспомнив о старинной традиции бросать монетки в фонтаны для того, чтобы когда-нибудь вернуться в это место, он решил запастись монетами заранее. Поскольку это всего лишь традиция, подумал Петя, то с него хватит оставить в каждом городе по одной копеечной монете — зачем тратить зря?

К сожалению, копеечные монеты – достаточно редкая вещь. В частности, таковых у Пети не нашлось. Купюр и монет всех остальных достоинств у него с избытком.

С этими мыслями Петя решил прогуляться до продуктового магазина – купить в дорогу немного еды. Из всего ассортимента ему подходило M видов товара (количество товаров каждого вида неограниченно), стоимость i -го равна a_i рублей b_i копеек. И тут его осенило. Если покупать товары в правильной последовательности, то он довольно быстро сможет скопить так нужные ему N копеечных монет!

Процесс покупки в магазине устроен следующим образом. Петя может заказать любой набор из подходящих ему товаров (каждого товара Петя может взять сколько угодно единиц). После чего он платит за них и получает сдачу минимальным числом купюр и монет (любых монет и купюр в кассе также с избытком). Это означает, например, что если ему должны сдать 11 рублей и 98 копеек сдачи, то он получит купюру в 10 рублей, монеты в 1 рубль, 50 копеек, 4 монеты в 10 копеек, одну монету в 5 копеек и три копеечных монеты. При этом он волен вносить любую сумму (лишь бы она была не меньше требуемой для оплаты) и платить любым набором купюр и монет, имеющихся у него в распоряжении.

После этого Петя может ещё раз подойти к кассе, сделать заказ, расплатиться имеющимися наличными (можно использовать и полученные до этого со сдачей) и так далее сколько угодно раз.

Петя хочет потратить в этом магазине как можно меньше денег. Помогите ему найти оптимальный способ обретения не менее N копеечных монет с минимальными затратами.

Комментарий

В России используются монеты и купюры достоинством 1, 5, 10, 50 копеек и 1, 2, 5, 10, 50, 100, 500, 1000 и 5000 рублей. 1 рубль равен 100 копейкам.

Входные данные

Сначала вводятся целые числа N и M ($0 \leq N \leq 10^8$, $0 \leq M \leq 100$) — количество городов, которые собирается посетить Петя, и количество подходящих ему видов товара. Далее идут M пар чисел a_i , b_i , обозначающих стоимость товара соответствующего типа ($0 \leq a_i \leq 100$, $0 \leq b_i \leq 99$). Стоимость товара всегда больше нуля.

Выходные данные

Если требуемое количество копеечных монет получить невозможно, выведите -1 . Иначе выведите минимальную сумму, которую должен потратить Петя на покупку товаров, чтобы получить N однокопеечных монет. Сумма должна быть выведена как два целых числа, задающих рубли и копейки (второе число обязано быть от 0 до 99).

Примеры

входные данные

3 1

0 2

выходные данные

0 2

входные данные

4 2

1 2

0 4

выходные данные

0 16

входные данные

1 3

0 1

0 4

28.

В стране Флатландия решили построить легкоатлетический манеж с M одинаковыми прямолинейными беговыми дорожками. Они будут покрыты полосами из синтетического материала пружинкина. На складе имеются N полос пружинкина, длины которых равны $1, 2, \dots, N$ метров соответственно (i -я полоса имеет длину i метров).

Было решено использовать все полосы со склада, что определило длину дорожек манежа. Полосы пружинкина должны быть уложены без перекрытий и промежутков. Разрезать полосы на части нельзя. Полосы укладываются вдоль дорожек, ширина полосы пружинкина совпадает с шириной беговой дорожки.

Требуется написать программу, которая определяет, можно ли покрыть всем имеющимся материалом M дорожек, и если это возможно, то распределяет полосы пружинкина по дорожкам.

Входные данные

Во входном файле содержатся два целых числа, разделенных пробелом: M — количество дорожек и N — количество полос пружинкина ($1 \leq M \leq 1000$, $1 \leq N \leq 30000$).

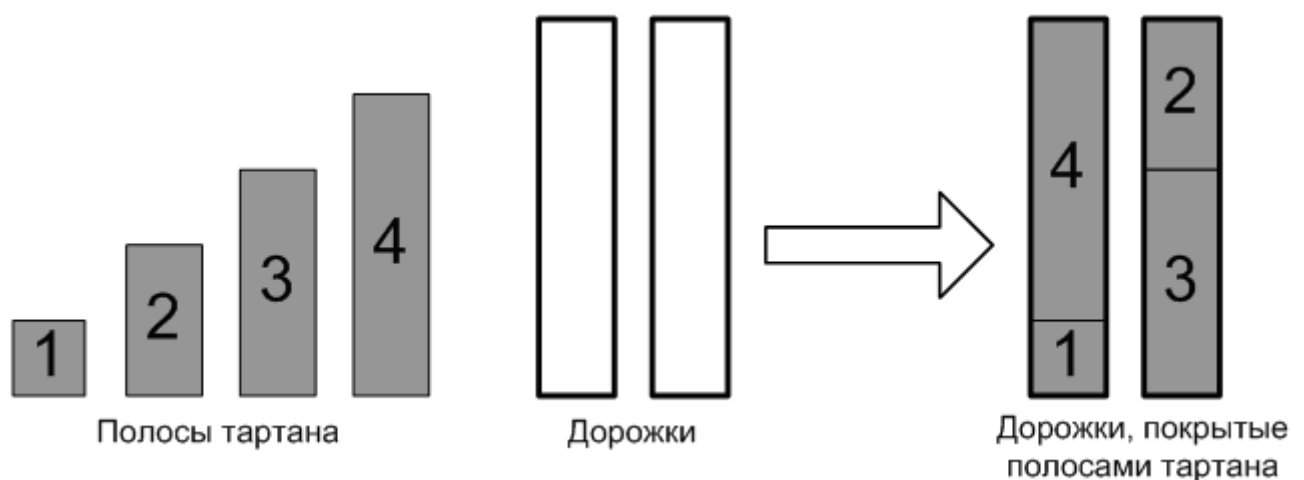
Выходные данные

В случае, если распределить имеющиеся полосы пружинкина на M дорожек одинаковой длины невозможно, то в выходной файл выведите слово «NO».

В противном случае, в первую строку выведите слово «YES». В последующих M строках дайте описание использованных полос для каждой дорожки в следующем формате: сначала целое число t — количество полос на дорожке, затем t целых чисел — длины полос, которые составят эту дорожку. Если решений несколько, можно вывести любое из них.

Примеры входных и выходных данных

Ввод	Вывод
2 4	YES 2 1 4 2 3 2
3 4	NO



29.

Скоро начнётся сезон роста бамбука. Скупщики бамбука принимают любое количество бамбука каждый день ровно в полдень. Однако цена бамбука каждый день меняется. Нам удалось узнать, по какой цене скупщики будут принимать бамбук. Кроме того, мы точно знаем, на сколько метров вырастает бамбук за каждые сутки (эта величина тоже меняется). В любой день можно либо срезать весь бамбук целиком и продать его, либо оставить бамбук расти дальше. После срезания бамбук продолжает расти. Требуется определить, какую максимальную прибыль от продажи бамбука можно получить. В полдень нулевого дня (начало сезона роста бамбука) длина бамбука равна 0. Сезон роста бамбука длится ровно N суток.

Входные данные

В первой строке входного файла находится натуральное число N , $1 \leq N \leq 100000$. В каждой из следующих N строк содержатся два целых положительных числа, разделенных пробелом: цена одного метра бамбука в определенный день и на сколько метров вырос бамбук за последние сутки. В $(i + 1)$ -й строке файла содержатся данные, относящиеся к полудню i -го дня.

Выходные данные

В единственной строке выходного файла следует выводить одно целое неотрицательное число – наибольшая возможная выручка от продажи бамбука. Гарантируется, что результат не превосходит $2^{63} - 1$.

Примеры

входные данные

```
8
2 7
4 1
3 3
5 5
2 4
5 2
4 7
1 1
```

выходные данные

```
139
```

Задание #2

Цель работы. Решить задачу с использованием метода динамического программирования.
Варианты:

1. Калькулятор

Имеется калькулятор, который выполняет три операции:

1. Прибавить к числу X единицу.
2. Умножить число X на 2.
3. Умножить число X на 3.

Определите кратчайшую последовательность операций, необходимую для получения из числа 1 заданное число N .

Входные данные

Программа получает на вход одно число N , не превосходящее 10^6 .

Выходные данные

Выведите строку, состоящую из цифр "1", "2" или "3", обозначающих одну из трех указанных операций, которая получает из числа 1 число N за минимальное число операций. Если возможных минимальных решений несколько, выведите любое из них.

Пример

Ввод	Вывод
1	
5	121
56234	333331222212221
0	3312

2. Разложение в сумму кубов

Дано натуральное число N . Необходимо представить его в виде суммы точных кубов, содержащей наименьшее число слагаемых. Программа должна вывести это число слагаемых и разложение в виде суммы. Если возможных решений несколько, выведите любое из них.

Входные данные

Программа получает на вход натуральное число N , не превосходящее 10^6 .

Выходные данные

Программа должна вывести число слагаемых и разложение в виде суммы кубов.

Примеры

входные данные

9

выходные данные

2

$2^3 + 1^3$

входные данные

27

выходные данные

1

3^3

3. Взрывоопасность-2

При переработке радиоактивных материалов образуются отходы трех видов — особо опасные (тип А), неопасные (тип В) и совсем не опасные (тип С). Для их хранения используются одинаковые контейнеры. После помещения отходов в контейнеры последние укладываются вертикальной стопкой. Стопка считается взрывоопасной, если в ней подряд идет более одного контейнера типа А. Стопка считается безопасной, если она не является взрывоопасной. Для заданного количества контейнеров N определить число безопасных стопок.

Входные данные

Одно число $1 \leq N \leq 20$.

Выходные данные

Одно число — количество безопасных вариантов формирования стопки.

Примечание

В примере из условия среди стопок длины 2 бывают безопасные стопки типов АВ, АС, ВА, ВВ, ВС, СА, СВ и СС. Стопки типа АА являются взрывоопасными.

Примеры

входные данные

2

выходные данные

8

4. Опасные ступеньки: количество способов

Вася каждый день поднимается по одной и той же лестнице. Одним шагом он может встать на следующую ступеньку или перешагнуть через одну ступеньку. Он уже знает, сколькими способами он может подняться на верхнюю ступеньку. Но недавно он обнаружил, что некоторые ступеньки обветшали, и ступать на них небезопасно. Он составил список таких ступенек, и теперь интересуется, сколькими способами можно подняться по лестнице, не наступая на эти ступеньки.

Входные данные

В первой строке вводится одно натуральное число N ($N < 40$): количество ступенек.

Во второй строке вводится одно натуральное число K ($K \leq N$): количество опасных ступенек.

В третьей строке вводятся K различных натуральных чисел в диапазоне от 1 до N : номера опасных ступенек.

Выходные данные

Выведите одно число: количество способов попасть на N -ю ступеньку.

Примеры

входные данные

10

3

5 1 2

выходные данные

0

входные данные

3

1

2

выходные данные

1

входные данные

7

2

3 5

выходные данные

2

5. Платная лестница

Мальчик подошел к платной лестнице. Чтобы наступить на любую ступеньку, нужно заплатить указанную на ней сумму. Мальчик умеет перешагивать на следующую ступеньку, либо перепрыгивать через ступеньку. Требуется узнать, какая наименьшая сумма понадобится мальчику, чтобы добраться до верхней ступеньки и его путь.



Входные данные

В первой строке входного файла вводится одно натуральное число $N \leq 100$ — количество ступенек.

В следующей строке вводятся N натуральных чисел, не превосходящих 100 — стоимость каждой ступеньки (снизу вверх).

Выходные данные

Выведите одно число — наименьшую возможную стоимость прохода по лесенке. В следующей строке выведите номера ступенек, по которым проходил мальчик.

Примеры

входные данные

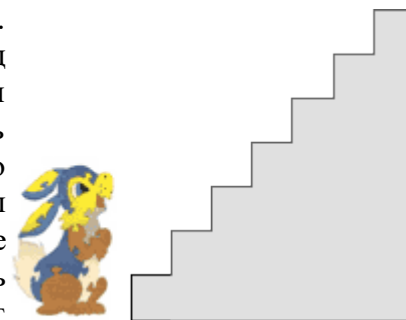
```
3
1 3 1
```

выходные данные

```
2
1 3
```

6. Зайчик

В нашем зоопарке появился заяц. Его поместили в клетку, и чтобы ему не было скучно, директор зоопарка распорядился поставить в его клетке лесенку. Теперь наш зайчик может прыгать по лесенке вверх, перепрыгивая через ступеньки. Лестница имеет определенное количество ступенек N . Заяц может одним прыжком преодолеть не более K ступенек. Для разнообразия зайчик пытается каждый раз найти новый путь к вершине лестницы. Директору любопытно, сколько различных способов есть у зайца добраться до вершины лестницы при заданных значениях K и N . Помогите директору написать программу, которая поможет вычислить это количество. Например, если $K=3$ и $N=4$, то существуют следующие маршруты: 1+1+1+1, 1+1+2, 1+2+1, 2+1+1, 2+2, 1+3, 3+1. Т.е. при данных значениях у зайца всего 7 различных маршрутов добраться до вершины лестницы.



Входные данные

В единственной строке входного файла INPUT.TXT записаны два натуральных числа K и N ($1 \leq K \leq N \leq 300$). K - максимальное количество ступенек, которое может преодолеть заяц одним прыжком, N – общее число ступенек лестницы.

Выходные данные

В единственную строку выходного файла OUTPUT.TXT нужно вывести количество возможных вариантов различных маршрутов зайца на верхнюю ступеньку лестницы без ведущих нулей.

Примеры

№	INPUT.TXT	OUTPUT.TXT
1	1 3	1
2	2 7	21
3	3 10	274

7. Башня

Петя в очередной купил себе набор из кубиков. На этот раз он выстроил из них настоящую крепость — последовательность из N столбиков, высота каждого столбика составляет A_i кубиков.

Вскоре ему стало интересно, насколько его крепость защищена от жуликов и воров. Для этого он ввел понятия башни. Башней называется любая последовательность из K столбиков подряд (где K — любимое число Пети). Защищенность башни определяется как суммарная высота всех столбиков этой башни (чем она больше, тем громаднее и ужаснее она кажется), умноженная на минимум высоты столбиков башни (т.к. враги, очевидно, будут пытаться проникнуть через самое слабое место башни). Неприступность крепости определяется как сумма защищенностей каждой из башен.

Петя решил как можно скорее посчитать, какова же неприступность его крепости. Однако вскоре он понял, что недостаточно знать высоту каждого из столбиков. В зависимости от того, как сгруппировать столбики в башни, получится разный результат. Разумеется, Петя выберет то разбиение на башни, при котором неприступность будет максимальна.

Петя успешно справился со своей задачей, но теперь Правительство Флатландии решило защитить свой горный курорт. Правительство уже построило крепость из кубиков (просто кубики были побольше). Теперь вы должны помочь Правительству посчитать неприступность этой крепости. Единственная трудность состоит в том, что у Правительства было очень много денег, и поэтому крепость была построена очень длинная.

Входные данные

В первой строке входного файла содержатся число N — количество столбиков в крепости и число K — любимое число Пети ($1 \leq K \leq N \leq 1000$). Далее на следующей строке содержатся N целых чисел, обозначающих A_i ($1 \leq A_i \leq 10^3$).

Выходные данные

На первой строке выведите число Q — количество башен в оптимальном разбиении. Далее выведите Q чисел — номера первых столбиков каждой башни.

Гарантируется, что в оптимальном разбиении неприступность крепости не превосходит 2×10^9 .

Примеры

входные данные

```
1 1
1
```

выходные данные

```
1
1
```

входные данные

```
2 1
1 1000
```

выходные данные

```
2
1 2
```

входные данные

```
8 3
1 2 3 4 1 6 7 8
```

выходные данные

```
2
2 6
```

8. Красим забор

Васин дедушка построил забор на даче из того, что попало под руку. Забор представляет собой ряд из N досок ширины 10 см, но, возможно, различной высоты.

Теперь Вася хочет покрасить забор таким способом. Он выбирает 5 произвольных подряд идущих досок и красит их в один цвет. Затем он выбирает 5 любых еще не покрашенных идущих подряд досок и красит их в другой цвет. И так продолжает до тех пор, пока может выбрать 5 подряд идущих не покрашенных досок.

Требуется определить, какую наибольшую площадь забора он сможет покрасить таким способом.

Входные данные

В первой строке входного файла вводится одно число N - количество досок.
Во второй строке входного файла вводятся N чисел - высоты 1-й, 2-й, ..., N -й доски забора в сантиметрах.
Все числа натуральные и не превосходят 100.

Выходные данные

Выведите одно число: наибольшую покрашенную площадь в квадратных сантиметрах.

Примеры

входные данные

```
6
1 2 3 4 5 6
```

выходные данные

```
200
```

входные данные

```
12
2 4 3 7 8 100 92 1 4 2 34 1
```

выходные данные

```
2550
```

9. Компьютерная игра (платформы)

Вы можете вспомнить хоть одного своего знакомого до двадцатилетнего возраста, который в детстве не играл в компьютерные игры? Если да, то может быть вы и сами не знакомы с этим развлечением? Впрочем, трудностей при решении этой задачи это создать не должно.

Во многих старых играх с двумерной графикой можно столкнуться с подобной ситуацией. Какой-нибудь герой прыгает по платформам (или островкам), которые висят в воздухе. Он должен перебраться от одного края экрана до другого. При этом при прыжке с одной платформы на соседнюю, у героя уходит $|y_2 - y_1|$ единиц энергии, где y_1 и y_2 — высоты, на которых расположены эти платформы. Кроме того, у героя есть суперприём, который позволяет перескочить через платформу, но на это затрачивается $3 \cdot |y_3 - y_1|$ единиц энергии. Конечно же, энергию следует расходовать максимально экономно.

Предположим, что вам известны координаты всех платформ в порядке от левого края до правого. Сможете ли вы найти, какое минимальное количество энергии потребуется герою, чтобы добраться с первой платформы до последней?

Входные данные

В первой строке записано количество платформ n ($1 \leq n \leq 30000$). Вторая строка содержит n натуральных чисел, не превосходящих 30000 — высоты, на которых располагаются платформы.

Выходные данные

Выведите единственное число — минимальное количество энергии, которую должен потратить игрок на преодоление платформ (конечно же в предположении, что cheat-коды использовать нельзя).

Примеры

входные данные

```
2
100 1
```

выходные данные

входные данные

3

1 100 80

выходные данные

119

10. Лотерея

На одном из телеканалов каждую неделю проводится следующая лотерея. В течение недели участники делают свои ставки. Каждая ставка заключается в назывании какого-либо M -значного числа в системе счисления с основанием K (то есть, по сути, каждый участник называет M цифр, каждая из которых лежит в диапазоне от 0 до $K-1$). Ведущие нули в числах допускаются.

В некоторый момент прием ставок на текущий розыгрыш завершается, и после этого ведущий в телеэфире называет выигравшее число (это также M -значное число в K -ичной системе счисления). После этого те телезрители, у кого первая цифра их числа совпала с первой цифрой числа, названного ведущим, получают выигрыш в размере A_1 рублей. Те, у кого совпали первые две цифры числа — получают A_2 рублей (при этом если у игрока совпала вторая цифра, но не совпала первая, он не получает ничего). Аналогично угадавшие первые три цифры получают A_3 рублей. И так далее. Угадавшие все число полностью получают A_M рублей. При этом если игрок угадал t первых цифр, то он получает A_t рублей, но не получает призы за угадывание $t-1$, $t-2$ и т.д. цифр. Если игрок не угадал первую цифру, он не получает ничего.

Напишите программу, которая по известным ставкам, сделанным телезрителями, находит число, которое должна назвать телеведущая, чтобы фирма-организатор розыгрыша выплатила в качестве выигрышей минимальную сумму. Для вашего удобства ставки, сделанные игроками, уже упорядочены по неубыванию.

Входные данные

В первой строке задаются числа N (количество телезрителей, сделавших свои ставки, $1 \leq N \leq 100000$), M (длина чисел $1 \leq M \leq 10$) K (основание системы счисления $2 \leq K \leq 10$). В следующей строке записаны M чисел A_1, A_2, \dots, A_M , задающих выигрыши в случае совпадения только первой, первых двух, ..., всех цифр ($1 \leq A_1 \leq A_2 \leq \dots \leq A_M \leq 100000$). В каждой из следующих N строк записано по одному M -значному K -ичному числу. Числа идут в порядке неубывания.

Выходные данные

В первой строке выведите искомое число (если решений несколько — выведите любое из них), а во второй строке — сумму, которую при назывании телеведущей первого числа придется выплатить в качестве выигрыша.

Примеры**входные данные**

```
10 3 2
1 3 100
000
000
001
010
100
100
100
```

100
110
111

выходные данные

011
6

входные данные

1 1 10
100
0

выходные данные

1
0

11. Игра со спичками-2

На столе лежит кучка из N спичек. Двое играют в такую игру. За один ход разрешается взять из кучки одну, две или три спички, так чтобы оставшееся количество спичек не было простым числом (Например, можно оставить в кучке 1 или 4 спички, но нельзя оставить 2 или 3). Выигрывает тот, кто забирает последнюю спичку. Требуется определить, кто из игроков имеет выигрышную стратегию.

Входные данные

Вводится одно число N ($1 \leq N \leq 10000$).

Выходные данные

Выведите число 1, если выигрышную стратегию имеет начинающий игрок, или число 2, если выигрышную стратегию имеет второй игрок.

Примеры

входные данные

2

выходные данные

1

входные данные

4

выходные данные

2

12. Треугольник Паскаля

Треугольник Паскаля строится следующим образом. Первая строка состоит из одного числа, равного единице. Каждая следующая содержит на одно число больше, чем предыдущая. Первое и последнее из этих чисел равны 1, а все остальные вычисляются как сумма числа, стоящего в предыдущей строке над ним и числа, стоящего в предыдущей же строке слева от него.

Входные данные

Вводится одно число N ($0 \leq N \leq 30$).

Выходные данные

Вывести N строк треугольника Паскаля (числа выводятся через пробел).

Примеры

входные данные

5

выходные данные

1

1 1

1 2 1

1 3 3 1

1 4 6 4 1

13. Шашку - в дамки

На шахматной доске (8x8) стоит одна белая шашка. Сколькими способами она может пройти в дамки?

(Белая шашка ходит по диагонали. на одну клетку вверх-вправо или вверх-влево. Шашка проходит в дамки, если попадает на верхнюю горизонталь.)

Входные данные

Вводятся два числа от 1 до 8: номер столбца (считая слева) и строки (считая снизу), где изначально стоит шашка.

Выходные данные

Вывести одно число - количество путей в дамки.

Примеры

входные данные

3 7

выходные данные

2

входные данные

1 8

выходные данные

1

входные данные

3 6

выходные данные

4

14. Вывести маршрут максимальной стоимости

В левом верхнем углу прямоугольной таблицы размером $N \times M$ находится черепашка. В каждой клетке таблицы записано некоторое число. Черепашка может перемещаться вправо или вниз, при этом маршрут черепашки заканчивается в правом нижнем углу таблицы.

Подсчитаем сумму чисел, записанных в клетках, через которую проползла черепашка (включая начальную и конечную клетку). Найдите наибольшее возможное значение этой суммы и маршрут, на котором достигается эта сумма.

Входные данные

В первой строке входных данных записаны два натуральных числа N и M , не превосходящих 100 — размеры таблицы. Далее идет N строк, каждая из которых

содержит M чисел, разделенных пробелами — описание таблицы. Все числа в клетках таблицы целые и могут принимать значения от 0 до 100.

Выходные данные

Первая строка выходных данных содержит максимальную возможную сумму, вторая — маршрут, на котором достигается эта сумма. Маршрут выводится в виде последовательности, которая должна содержать $N-1$ букву D, означающую передвижение вниз и $M-1$ букву R, означающую передвижение направо. Если таких последовательностей несколько, необходимо вывести ровно одну (любую) из них.

Примеры

входные данные

```
5 5
9 9 9 9 9
3 0 0 0 0
9 9 9 9 9
6 6 6 6 8
9 9 9 9 9
```

выходные данные

```
74
D D R R R R D D
```

15. K-ичные числа

Требуется вычислить количество N -значных чисел в системе счисления с основанием K , таких что их запись не содержит двух подряд идущих нулей.

Ограничения: $2 \leq K \leq 10$, N + $K \leq 18$.

Формат входных данных

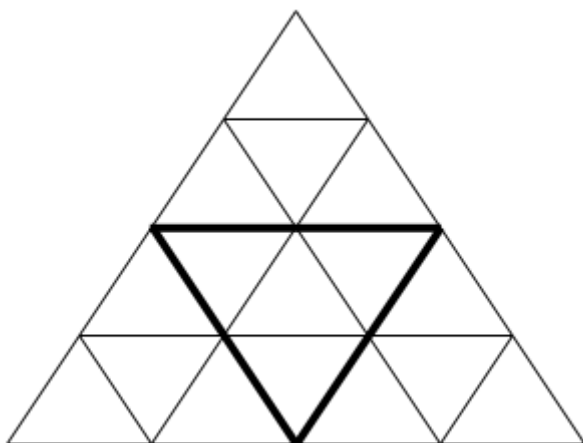
Числа N и K в десятичной записи, разделенные пробелом или переводом строки.

Формат выходных данных

Искомое число в десятичной записи.

16. Количество треугольников

Рассмотрим фигуру, аналогичную показанной на рисунке (большой равносторонний треугольник, составленный из маленьких равносторонних треугольников). На рисунке приведена фигура, состоящая из 4-х уровней треугольников.



Напишите программу, которая будет определять, сколько всего в ней треугольников (необходимо учитывать не только "маленькие" треугольники, а вообще все треугольники — в частности, треугольник, выделенный жирным, а также вся фигура, являются интересующими нас треугольниками).

Входные данные

Вводится одно число N — количество уровней в фигуре ($1 \leq N \leq 100000$).

Выходные данные

Выведите количество треугольников в такой фигуре.

Примеры

входные данные

1

выходные данные

1

входные данные

2

выходные данные

5

входные данные

4

выходные данные

27

17. Гвоздики

В дощечке в один ряд вбиты гвоздики. Любые два гвоздика можно соединить ниточкой. Требуется соединить некоторые пары гвоздиков ниточками так, чтобы к каждому гвоздику была привязана хотя бы одна ниточка, а суммарная длина всех ниточек была минимальна.

Входные данные

В первой строке входных данных записано число N — количество гвоздиков ($2 \leq N \leq 100$). В следующей строке заданы N чисел — координаты всех гвоздиков (неотрицательные целые числа, не превосходящие 10000).

Выходные данные

Выведите единственное число — минимальную суммарную длину всех ниточек.

Пример

Входные данные	Выходные данные
5 4 10 0 12 2	6

Примеры

входные данные

6

3 4 6 12 13 14

выходные данные

5

18. Маскарад

По случаю введения больших новогодних каникул устраивается великий праздничный бал-маскарад. До праздника остались считанные дни, поэтому срочно нужны костюмы для участников. Для пошивки костюмов требуется L метров ткани. Ткань продается в N магазинах, в которых предоставляются скидки оптовым покупателям. В магазинах можно купить только целое число метров ткани. Реклама магазина номер i гласит: "Мы с радостью продадим Вам метр ткани за P_i бурлей, однако если Вы купите не менее R_i метров, то получите прекрасную скидку - каждый купленный метр обойдется Вам всего в Q_i бурлей". Чтобы воплотить в жизнь лозунг "экономика страны должна быть экономной", правительство решило потратить на закупку ткани для костюмов минимальное количество бурлей из государственной казны. При этом ткани можно купить больше, чем нужно, если так окажется дешевле. Ответственный за покупку ткани позвонил в каждый магазин и узнал, что:

- 1) реклама каждого магазина содержит правдивую информацию о ценах и скидках;
- 2) магазин номер i готов продать ему не более F_i метров ткани.

Ответственный за покупку очень устал от проделанной работы и поэтому поставленную перед ним задачу <закупить ткань за минимальные деньги> переложил на своих помощников. Напишите программу, которая определит, сколько ткани нужно купить в каждом из магазинов так, чтобы суммарные затраты были минимальны.

Формат входных данных

В первой строке входного файла `s.in` записаны два целых числа N и L ($1 < N \leq 100$, $0 < L \leq 100$). В каждой из последующих N строк находится описание магазина номер i - 4 целых числа P_i , R_i , Q_i , F_i ($1 \leq Q_i \leq P_i \leq 1000$, $1 \leq R_i \leq 100$, $0 \leq F_i \leq 100$).

Формат выходных данных

Первая строка выходного файла `s.out` должна содержать единственное число - минимальное необходимое количество бурлей.

Во второй строке выведите N чисел, разделенных пробелами, где i -е число определяет количество метров ткани, которое нужно купить в i -м магазине.

Если в i -м магазине ткань покупать не будет, то на i -м месте должно стоять число 0. Если вариантов покупки несколько, выведите любой из них.

Если ткани в магазинах недостаточно для пошивки костюмов, выходной файл должен содержать единственное число -1.

Примеры

c.in	c.out
2 14	88
7 9 6 10	10 4
7 8 6 10	
1 20	-1
1 1 1 1	

19. Скобки

Задан шаблон, состоящий из круглых скобок и знаков вопроса. Требуется определить, сколькими способами можно заменить знаки вопроса круглыми скобками так, чтобы получилось правильное скобочное выражение.

Входные данные

Первая строка входного файла содержит заданный шаблон длиной не более 80 символов.

Выходные данные

Выведите в выходной файл искомое количество способов. Исходные данные будут таковы, что это количество не превзойдет $2 \cdot 10^9$.

Пример входного файла

????(?)

Пример выходного файла

2

20. Красивый номер телефона

Шахматная ассоциация решила оснастить всех своих сотрудников такими телефонными номерами, которые бы набирались на кнопочном телефоне ходом коня. Например, ходом коня набирается телефон 340-49-27. При этом телефонный номер не может начинаться ни с цифры 0, ни с цифры 8.

7	8	9
4	5	6
1	2	3
	0	

Напишите программу, определяющую количество телефонных номеров длины N , набираемых ходом коня .

Входные данные

Во входном файле записано целое число N ($1 \leq N \leq 100$).

Выходные данные

Выведите в выходной файл искомое количество телефонных номеров.

Пример входного файла

2

Пример выходного файла

16

21. Плитки

У Пети имеется неограниченный набор красных, синих и зеленых плиток размером $1 * 1$. Он выбирает ровно N плиток и выкладывает их в полосу. Например, при $N = 10$ она может выглядеть следующим образом:

К	К	К	С	З	К	К	З	К	С
---	---	---	---	---	---	---	---	---	---

(буквой К обозначена красная плитка, С — синяя, З — зеленая). После этого Петя заполняет следующую таблицу:

	Красный	Синий	Зеленый
Красный	Y	Y	Y
Синий	Y	N	Y
Зеленый	Y	Y	N

В клетке на пересечении строки, отвечающей цвету А, и столбца, отвечающего цвету Б, он записывает "Y", если в его полоске найдется место, где рядом лежат плитки цветов А и Б и "N" в противном случае. Считается, что плитки лежат рядом, если у них есть общая сторона. (Очевидно, что таблица симметрична относительно главной диагонали — если плитки цветов А и Б лежали рядом, то рядом лежали и плитки цветов Б и А.) Назовем такую таблицу *диаграммой смежности* данной полосы.

Так, данная таблица представляет собой диаграмму смежности приведенной выше

Петя хочет узнать, сколько различных полосок имеет определенную диаграмму смежности. Помогите ему.

(Заметьте, что полоски, являющиеся отражением друг друга, но не совпадающие, считаются разными. Так, полоска

С	К	З	К	К	З	С	К	К	К
---	---	---	---	---	---	---	---	---	---

не совпадает с полоской, приведенной в начале условия.)

Формат входных данных

Первая строка входного файла содержит число N ($1 \leq N \leq 100$). Следующие три строки входного файла, содержащие по три символа из набора {"Y", "N"}, соответствуют трем строкам диаграммы смежности. Других символов, включая пробелы, во входном файле не содержится. Входные данные корректны, т.е. диаграмма смежности симметрична

Формат выходных данных

Выведите в выходной файл количество полосок длины N , имеющих приведенную во входном файле диаграмму смежности.

22. Альтернативные пути

Задано прямоугольную таблицу размером M строк на N столбиков. В каждой клеточке записано натуральное число, не превышающее 200. Путник должен пройти по этой таблице из левого верхнего угла в правый нижний, на каждом шаге перемещаясь либо на 1 клеточку направо, либо на 1 клеточку вниз. Очевидно, таких путей много. Для каждого пути можно вычислить сумму чисел в пройденных клеточках. Среди этих сумм, очевидно, есть максимальная.

Будем снисходительными к Путнику, считая <хорошими> не только пути, на которых в точности достигается максимально возможная сумма, а еще и пути, сумма которых отличается от максимальной не более чем на K .

Количество <хороших> путей гарантированно не превышает 10^9 .

Задание

Напишите программу, находящую значение максимально возможной суммы и количества <хороших> путей.

Входные данные

Первая строка входного файла GOODWAYS.DAT содержит три целых числа M ($2 \leq M \leq 200$), N ($2 \leq N \leq 200$) и K ($0 \leq K \leq 200$). Каждая из последующих M строк содержит N чисел, записанных в соответствующих клеточках.

Выходные данные

Первая строка выходного файла GOODWAYS.SOL должна содержать максимальную возможную сумму; вторая строка - количество маршрутов, сумма чисел которых отличается от максимальной не более чем на K .

Пример входных и выходных данных

GOODWAYS.DAT	GOODWAYS.SOL
2 3 3	20
1 9 7	2
2 5 3	

23. Покупка билетов

За билетами на премьеру нового мюзикла выстроилась очередь из N человек, каждый из которых хочет купить 1 билет. На всю очередь работала только одна касса, поэтому продажа билетов шла очень медленно, приводя «постояльцев» очереди в отчаяние. Самые сообразительные быстро заметили, что, как правило, несколько билетов в одни руки кассир продаёт быстрее, чем когда эти же билеты продаются по одному. Поэтому они предложили нескольким подряд стоящим людям отдавать деньги первому из них, чтобы он купил билеты на всех.

Однако для борьбы со спекулянтами кассир продавала не более 3-х билетов в одни руки, поэтому договориться таким образом между собой могли лишь 2 или 3 подряд стоящих человека.

Известно, что на продажу i -му человеку из очереди одного билета кассир тратит A_i секунд, на продажу двух билетов — B_i секунд, трех билетов — C_i секунд. Напишите программу, которая подсчитает минимальное время, за которое могли быть обслужены все покупатели.

Обратите внимание, что билеты на группу объединившихся людей всегда покупает первый из них. Также никто в целях ускорения не покупает лишних билетов (то есть билетов, которые никому не нужны).

Входные данные

На вход программы поступает сначала число N — количество покупателей в очереди ($1 \leq N \leq 5000$). Далее идет N троек натуральных чисел A_i, B_i, C_i . Каждое из этих чисел не превышает 3600. Люди в очереди нумеруются, начиная от кассы.

Выходные данные

Требуется вывести одно число — минимальное время в секундах, за которое могли быть обслужены все покупатели.

Примеры

входные данные

```
2
3 4 5
1 1 1
```

выходные данные

```
4
```

24. Выражение

Петя - большой любитель математических головоломок. Недавно он прочитал в одном популярном журнале о новой головоломке. Он пытался ее решить несколько дней, но это ему так и не удалось. Помогите Пете справиться с не поддающейся задачей.

В ряд выписаны n чисел. Требуется поставить между каждой парой соседних чисел один из знаков "+" или "×" таким образом, чтобы значение получившегося выражения было как можно больше. Использовать скобки не разрешается.

Например, для последовательности чисел 1, 2, 3, 1, 2, 3 оптимально расставить знаки следующим образом: $1 + 2 \times 3 \times 1 \times 2 \times 3$. Значение выражения в этом случае равно 37.

Входные данные

В первой строке вводится число n ($2 \leq n \leq 200000$). Вторая строка содержит n целых чисел - числа, между которыми следует расставить знаки. Все числа находятся в диапазоне от 0 до 109.

Выходные данные

Выведите оптимальное выражение. В качестве знака "×" выводите символ "*" (звездочку). Если оптимальных решений несколько, выведите любое из них.

Примеры

входные данные

```
6
1 2 3 1 2 3
```

выходные данные

```
1+2*3*1*2*3
```

25. Школа танцев

В школу бальных танцев профессора Падеграса записались n учеников — мальчиков и девочек. Профессор построил их в один ряд, и хочет отобрать из них для первого занятия группу стоящих подряд учеников, в которой количество мальчиков и девочек одинаково. Сколько вариантов выбора есть у профессора?

Входные данные

В первой строке задано число n ($1 \leq n \leq 10^6$). Во второй строке задается описание построенного ряда из мальчиков и девочек — строка из n символов а и b (символ а соответствует девочке, а символ b — мальчику).

Выходные данные

В единственной строке должно содержаться единственное число — количество вариантов выбора требуемой группы.

Примеры

входные данные

```
8
```

26. Валютные махинации

Петя, изучая, как меняется курс рубля по отношению к доллару и евро, вывел закон, по которому происходят эти изменения (или думает, что вывел). По этому закону Петя рассчитал, каков будет курс рубля по отношению к доллару и евро в ближайшие N дней.

У Пети есть 100 рублей. В каждый из дней он может обменивать валюты друг на друга по текущему курсу без ограничения количества (при этом курс доллара по отношению к евро соответствует величине, которую можно получить, обменяв доллар на рубли, а потом эти рубли — на евро). Поскольку Петя будет оперировать не с наличной валютой, а со счетом в банке, то он может совершать операции обмена с любым (в том числе и нецелым) количеством единиц любой валюты.

Напишите программу, которая вычисляет, какое наибольшее количество рублей сможет получить Петя к исходу N -го дня.

Законы изменения курсов устроены так, что в течение указанного периода рублевый эквивалент той суммы, которая может оказаться у Пети, не превысит 10^8 рублей.

Входные данные

Первая строка входного файла содержит одно число N ($1 \leq N \leq 5000$). В каждой из следующих N строк записано по 2 числа, вычисленных по Петиним законам для соответствующего дня — сколько рублей будет стоить 1 доллар, и сколько рублей будет стоить 1 евро. Все эти значения не меньше 0.01 и не больше 10000. Значения заданы точно и выражаются вещественными числами не более, чем с двумя знаками после десятичной точки.

Выходные данные

В выходной файл выведите искомую величину с точностью не менее двух знаков после десятичной точки.

Примеры

входные данные

1

30.00 9999.99

выходные данные

100.00

27. Фермер

Фермер решил на своем квадратном участке земли вспахать пашню квадратной формы максимальной площади, т.к. он посчитал, что именно квадратная форма пашни наиболее удобна для обработки. Но на его участке присутствуют деревья и хозяйственные постройки, которые он никуда не хочет переносить, а также иные места, не пригодные для пашни. Для удобства он составил квадратную карту местности $N \times N$ в форме матрицы и пометил нулями непригодные для пашни зоны, в остальные зоны он поставил единицу.

Необходимо помочь фермеру определить максимальную площадь пашни.

Входные данные

В первой строке входного файла INPUT.TXT записано единственное натуральное число N ($1 \leq N \leq 1000$) – длина стороны квадратного участка фермы. Далее, следует N строк, в каждой из которых находится последовательность (без пробелов) нулей и единиц, описывающих ферму.

Выходные данные

В выходной файл OUTPUT.TXT необходимо вывести максимально возможную площадь пашни.

Пример

№	INPUT.TXT	OUTPUT.TXT
	7	9
	1101101	
	1111110	
1	1011100	
	0011100	
	1000010	
	1100111	
	1001110	

28. День Рождения

На день рождения Пете подарили коробку кубиков. На каждом кубике написано некоторое целое число. Петя выложил все n своих кубиков в ряд, так что числа на кубиках оказались расположены в некотором порядке $a[1], a[2], \dots, a[n]$. Теперь он хочет раскрасить кубики в разные цвета таким образом, чтобы для каждого цвета последовательность чисел на кубиках этого цвета была строго возрастающей. То есть, если кубики с номерами $i[1], i[2], \dots, i[k]$ покрашены в один цвет, то $a[i[1]] < a[i[2]] < \dots < a[i[k]]$. Петя хочет использовать как можно меньше цветов. Помогите ему!

Входные данные

Первая строка входного файла содержит число n - количество кубиков у Пети ($1 \leq n \leq 250000$). Затем следует n чисел, разделенных пробелами и/или переводами строки - $a[1], a[2], \dots, a[n]$

Выходные данные

На первой строке выходного файла выведите число L - наименьшее количество цветов, которое потребуется Пете. На следующей строке выведите n чисел из диапазона от 1 до L - цвета, в которые Петя должен покрасить кубики.

Примеры

входные данные

10
2 3 1 3 2 1 2 2 4 3

выходные данные

5
1 1 2 2 3 4 4 5 1 3

Лабораторная работа № 9с.

Цель работы. Решить задачу с использованием метода динамического программирования.

29. Робот исполнитель

Квадрат разлинован на $N \times N$ клеток ($1 < N < 17$). Исполнитель Робот может перемещаться по клеткам, выполняя за одно перемещение одну из двух команд: вправо или вниз. По команде вправо Робот перемещается в соседнюю правую клетку, по команде вниз – в соседнюю нижнюю. При попытке выхода за границу квадрата Робот разрушается. Перед каждым запуском Робота в каждой клетке квадрата лежит монета достоинством от 1 до 100. Посетив клетку, Робот **забирает монету с собой только в том случае, если её номинал – число, кратное 3; если номинал монеты – число, не кратное 3, то Робот не берёт монету**; это также относится к начальной и конечной клетке маршрута Робота. Определите максимальную и минимальную денежную сумму, которую может собрать Робот, пройдя из левой верхней клетки в правую нижнюю. В ответе укажите два числа – сначала максимальную сумму, затем минимальную.

30. Робот исполнитель и ямы

Квадрат разлинован на $N \times N$ клеток ($1 < N < 17$). В любой клетке поля может быть стена (стены обозначены значениями больше 100, но меньше 500) или яма (ямы обозначены значениями меньше 0, но больше -400). Робот может двигаться только вниз или вправо. При попытке зайти на клетку со стеной Робот разрушается. При попытке зайти на клетку с ямой Робот застревает в ней и не может двигаться дальше.

Определите максимальную и минимальную денежную сумму, которую может собрать Робот, пройдя из левой верхней клетки в правую нижнюю, не разрушившись и не застряв в яме. Известно, что такой путь существует. В ответе укажите два числа – сначала максимальную сумму, затем минимальную.

31. Робот исполнитель и четность

Квадрат разлинован на $N \times N$ клеток ($1 < N < 17$). Исполнитель Робот может перемещаться по клеткам, выполняя за одно перемещение одну из двух команд: вправо или вниз. По команде вправо Робот перемещается в соседнюю правую клетку, по команде вниз – в соседнюю нижнюю. При попытке выхода за границу квадрата Робот разрушается. Перед каждым запуском Робота в каждой клетке квадрата записано число от 10 до 99. Вначале счёт равен 0. Посетив клетку с нечетным значением, Робот увеличивает счет на 1; иначе увеличивает счёт на 2.

Определите максимальное и минимальное значение счета, который может собрать Робот, пройдя из левой верхней клетки в правую нижнюю. В ответе укажите два числа – сначала максимальную сумму, затем минимальную.

Пример входных данных:

1	8	8	4
10	1	1	3
1	3	12	2

2	3	5	6
---	---	---	---

Для указанных входных данных ответом должна быть пара чисел 12 9.

32. Грузовики

Квадрат разлинован на $N \times N$ клеток ($3 < N < 15$), где N – нечетное число. На поле работает 4 исполнителя Грузовичок, которые начинают движение из центральной клетки. Например, для $N = 5$ из клетки С3. Каждый исполнитель движется в один из углов – левый верхний, правый верхний, левый нижний или правый нижний – и может двигаться соответственно только – налево и вверх, направо и вверх, вниз и влево, вниз и вправо.

Исполнители работают независимо друг от друга на своей копии поля. Каждая пройденная клетка содержит число – массу в килограммах забираемого груза. Цель исполнителя – забрать как можно большую массу груза (в килограммах). Необходимо найти наибольшую массу собранного груза для каждого Грузовичка. В ответе запишите четыре числа в порядке возрастания.

Пример входных данных:

1	8	8	4	10
10	1	1	3	2
1	3	12	2	8
2	3	5	6	11
5	19	14	11	5

Для указанных входных данных ответом должна быть четверка чисел – результаты работы четырёх исполнителей в порядке возрастания:

30 35 55 47

33. Робот и нули

Квадрат разлинован на $N \times N$ клеток ($3 < N < 17$). В каждой клетке записано целое число. Исполнитель Робот может перемещаться по клеткам, выполняя за одно перемещение одну из двух команд: вправо или вниз. По команде вправо Робот перемещается в соседнюю правую клетку, по команде вниз – в соседнюю нижнюю. При попытке выхода за границу квадрата Робот разрушается. В каждой клетке квадрата записано число от 10 до 99 или 0. Посетив клетку, Робот прибавляет к счету значение, записанное в этой клетке.

Робот движется из левой верхней клетки в правую нижнюю. Необходимо найти максимальный и минимальный результаты работы исполнителя Робот в заданном поле. Запрещается посещать одну клетку дважды, а также клетки с нулевым значением. Известно, что как минимум один путь из начальной клетки в конечную точно существует.

Пример входных данных:

1	8	8	4	10
---	---	---	---	----

10	1	1	0	2
1	3	12	0	8
2	0	0	0	11
5	19	14	11	5

Для указанных входных данных ответом должна быть пара чисел – минимальное и максимальное значения: 57 68

34. Дружище

Квадрат разлинован на $N \times N$ клеток ($3 < N < 17$). В каждой клетке лежат конфеты, количество которых соответствует записанному числу. На поле работает исполнитель Дружище, который съедает все конфеты в клетке. Также, если исполнитель проходит между двумя четными или двумя нечетными значениями, то Добрый Волшебник дает ему еще 10 конфет, которые он, конечно же, сразу съедает. Так, например, если исполнитель приходит в клетку C3 из клетки B3, считается, что он прошел между клетками C2 и C4, если в C3 из C2 – между B3 и D3. Исполнитель может перемещаться по клеткам, выполняя за одно перемещение одну из двух команд: вправо или вниз. По команде вправо Дружище перемещается в соседнюю правую клетку, по команде вниз – в соседнюю нижнюю. При попытке выхода за границу квадрата Дружище расстраивается, что ему не дают конфеты, и отказывается идти дальше.

Нам важно, чтобы Дружище съел как можно меньше конфет и при этом добрался из левой верхней клетки в правую нижнюю.

Пример входных данных:

1	8	8	4	10
10	1	1	3	2
1	3	12	2	8
2	3	5	6	11
5	19	14	11	5

Для указанных входных данных ответом должно быть число – минимально возможное съедаемое количество конфет – 53.

35. Робот и четность

Квадрат разлинован на $N \times N$ клеток ($1 < N < 17$). Исполнитель Робот может перемещаться по клеткам, выполняя за одно перемещение одну из двух команд: вправо или вниз. По команде вправо Робот перемещается в соседнюю правую клетку, по команде вниз – в

соседнюю нижнюю. При попытке выхода за границу квадрата Робот разрушается. Робот может брать монеты **только с тех клеток, где количество монет чётно**. Если количество монет нечётно, то Робот не берёт в этой клетке ни одной монеты. Определите максимальную и минимальную денежную сумму, которую может собрать Робот, пройдя из левой верхней клетки в правую нижнюю. В ответе укажите два числа – сначала максимальную сумму, затем минимальную.

36. Робот и контейнеры

Квадрат разлинован на $N \times N$ клеток ($1 < N < 17$). Исполнитель Робот может перемещаться по клеткам, выполняя за одно перемещение одну из двух команд: вправо или вниз. По команде вправо Робот перемещается в соседнюю правую клетку, по команде вниз – в соседнюю нижнюю. При попытке выхода за границу квадрата Робот разрушается. С каждой клетки Робот забирает наибольшее количество контейнеров вместимостью 8 монет каждый, **полностью заполненных** монетами. Определите максимальную и минимальную денежную сумму, которую может собрать Робот, пройдя из левой НИЖНЕЙ клетки в правую ВЕРХНЮЮ. В ответе укажите два числа – сначала максимальную сумму, затем минимальную.

37. С севера на юг

Исходные данные для Робота записаны в виде таблицы прямоугольной формы. Роботу нужно перейти через поле с севера (верхняя строка) на юг (нижняя строка). Он может начать переход с любой клетки верхней строки и закончить на любой клетке нижней строки. С каждым шагом Робот переходит в следующий ряд и может за одно перемещение попасть в одну из трех клеток следующей строки (на клетку прямо или боковые с ней). Ходы только вбок (без смены строки) и/или назад запрещены. В каждой клетке поля лежит монета достоинством от 1 до 100. Робот собирает все монеты по пройденному маршруту.

Определите максимальную и минимальную денежную сумму, которую может собрать Робот, пройдя с северной границы поля (сверху) до южной границы поля (снизу). В ответе укажите два числа – сначала максимальную сумму, затем минимальную.

38. С запада на восток

Исходные данные для Робота записаны в виде таблицы прямоугольной формы. Роботу нужно перейти через поле с запада (левый столбец) на восток (правый столбец). Он может начать переход с любой клетки левого столбца и закончить на любой клетке правого столбца. С каждым шагом Робот переходит в следующий столбец и может за одно перемещение попасть в одну из трех клеток следующего столбца (на клетку прямо или боковые с ней). Ходы только вбок (без смены столбца) и/или назад запрещены. В каждой клетке поля лежит монета достоинством от 1 до 100. Робот собирает все монеты по пройденному маршруту. Определите максимальную и минимальную денежную сумму, которую может собрать Робот, пройдя с западной границы поля (слева) до восточной границы поля (справа). В ответе укажите два числа: сначала максимальную сумму, затем минимальную.

39. С севера на юг 2

Исходные данные для Робота записаны электронной таблицей прямоугольной формы. Роботу нужно перейти через поле с севера (верхняя строка) на юг (нижняя строка). Он

может начать переход с любой клетки верхней строки и закончить на любой клетке нижней строки. С каждым шагом Робот переходит в следующий ряд и может за одно перемещение попасть в одну из трех клеток следующей строки (на клетку прямо или боковые с ней). Ходы только вбок (без смены строки) и/или назад запрещены. В каждой клетке поля лежит монета достоинством от 1 до 100. Робот собирает все монеты по пройденному маршруту.

Известно, что Робот собрал максимальное количество монет, пройдя с северной границы поля (сверху) до южной границы поля (снизу). В ответе укажите два числа – количество монет из первой и последней клетки маршрута.

40. С юга на север

Исходные данные для Робота записаны в виде электронной таблицы прямоугольной формы. Роботу нужно перейти через поле с юга (нижняя строка) на север (верхняя строка). Он может начать переход с любой клетки нижней строки и закончить на любой клетке верхней строки. С каждым шагом Робот переходит в следующий ряд и может за одно перемещение попасть в одну из трех клеток следующей строки (на клетку прямо или боковые с ней). Ходы только вбок (без смены строки) и/или назад запрещены. В каждой клетке поля лежит монета достоинством от 1 до 100. Робот собирает все монеты по пройденному маршруту.

Робот собрал минимальное количество монет, пройдя с южной границы поля (снизу) до северной границы поля (сверху). В ответе укажите два числа: количество монет из первой и последней клетки маршрута.

41. Влево, вниз и влево-вниз

Квадрат разлинован на $N \times N$ клеток ($1 < N < 20$). Исполнитель Робот может перемещаться по клеткам, выполняя за одно перемещение одну из трёх команд: влево, вниз или влево-вниз. По команде влево Робот перемещается в соседнюю левую клетку, по команде вниз – в соседнюю верхнюю, а по команде влево-вниз – на одну клетку влево и вниз по диагонали. При попытке выхода за границу квадрата Робот разрушается. Перед каждым запуском Робота в каждой клетке квадрата записана величина вознаграждения от 1 до 100. Попад в клетку после хода влево или вниз, Робот получает указанное в ней вознаграждение, а если он попал в клетку после выполнения команды влево-вниз, вознаграждение удваивается. Это также относится к начальной и конечной клетке маршрута Робота. Определите максимальное и минимальное вознаграждение, которое может получить Робот, пройдя из правой верхней клетки в левую нижнюю. В ответе укажите два числа – сначала максимальное вознаграждение, затем минимальное.

42. Ладья

Квадрат разлинован на $N \times N$ клеток ($1 < N < 20$), в каждой клетке записано целое число. В правом верхнем углу квадрата стоит ладья. За один ход ладья может переместиться в пределах квадрата на любое количество клеток влево или вниз (вправо и вверх ладья ходить не может). Определите минимальную и максимальную сумму чисел в клетках, в которых может остановиться ладья при перемещении из правого верхнего угла в левый нижний.

43. Арифметическая прогрессия

Имеется набор данных, состоящий из N различных положительных чисел. Необходимо из этих чисел построить самую длинную возрастающую арифметическую прогрессию с шагом от 1 до 100 включительно и вывести её длину.

Входные данные: Даны два входных файла: каждый из которых содержит в первой строке количество чисел N ($1 \leq N \leq 100000$). Каждая из следующих N строк содержит одно натуральное число, не превышающее 10^8 .

Пример входного файла:

```
6
1
4
7
3
20
5
```

Для указанных входных данных самая большая арифметическая прогрессия будет $\{1, 3, 5, 7\}$ с шагом 2 и длиной 4. Программа должна вывести ответ 4. В ответе укажите два числа: сначала искомое значение для файла А, затем для файла В.

44. Сумма оканчивающаяся на 50

В файле записана последовательность натуральных чисел. Гарантируется, что все числа различны. Рассматриваются всевозможные группы чисел, состоящие из любого количества элементов последовательности. Необходимо найти наибольшую сумму такой группы, заканчивающуюся на 50. Программа должна вывести эту сумму.

Входные данные: Даны два входных файла: каждый из которых содержит в первой строке количество чисел N ($1 \leq N \leq 100000$). Каждая из следующих N строк содержит одно натуральное число, не превышающее 10^8 .

Пример входного файла:

```
6
21
29
12
72
14
28
```

Для указанных данных можно выбрать следующие группы: $\{21, 29\}$; $\{21, 29, 72, 28\}$. Сумма элементов данных групп равна 50 и 150. Программа должна вывести наибольшую из этих сумм – 150. В ответе укажите два числа: сначала искомое значение для файла А, затем для файла В.

45. Сумма кратная 25

В файле записана последовательность натуральных чисел. Гарантируется, что все числа различны. Рассматриваются всевозможные группы чисел, состоящие из любого количества элементов последовательности. Необходимо найти наибольшую сумму такой группы, кратную 25. Программа должна вывести эту сумму.

Входные данные: Даны два входных файла: каждый из которых содержит в первой строке количество чисел N ($1 \leq N \leq 100000$). Каждая из следующих N строк содержит одно натуральное число, не превышающее 10^8 .

Пример входного файла:

```
5
16
```

34
7
25
13

Для указанных данных можно выбрать следующие группы: {16, 34}; {16, 34, 25}. Суммы элементов данных групп равны 50 и 75. Программа должна вывести наибольшую из этих сумм – 75. В ответе укажите два числа: сначала искомое значение для файла А, затем для файла В.

45. Количество групп

В файле записана последовательность натуральных чисел. Гарантируется, что все числа различны. Рассматриваются всевозможные группы чисел, состоящие из любого количества элементов последовательности. Необходимо найти количество таких групп, для которых сумма элементов кратна 3.

Входные данные: Даны два входных файла: каждый из которых содержит в первой строке количество чисел N ($1 \leq N \leq 100000$). Каждая из следующих N строк содержит одно натуральное число, не превышающее 10^8 .

Пример входного файла:

4
5
7
12
23

Для указанных данных можно выбрать следующие группы: {12}; {7, 23}; {7, 12, 23}; {5, 7}; {5, 7, 12}. Программа должна вывести количество этих групп – 5. В ответе укажите два числа: сначала искомое значение для файла А, затем для файла В.

46. Группа из 4х элементов

В файле записана последовательность натуральных чисел. Гарантируется, что все числа различны. Из этой последовательности нужно выбрать четыре числа, чтобы их сумма делилась на 6 и была наибольшей. Какую наибольшую сумму можно при этом получить?

Входные данные: Даны два входных файла: каждый из которых содержит в первой строке количество чисел N ($1 \leq N \leq 100000$). Каждая из следующих N строк содержит одно натуральное число, не превышающее 10^8 .

Пример входного файла:

6
6
4
13
11
10
8

Для указанных данных можно выбрать четвёрки 4, 13, 11, 8 (сумма 36) и 13, 11, 10, 8 (сумма 42). Наибольшая из сумм – 42. В ответе укажите два числа: сначала искомое значение для файла А, затем для файла В.

47. Группа из 3х элементов

В файле записана последовательность натуральных чисел. Гарантируется, что все числа различны. Из этой последовательности нужно выбрать три числа, чтобы их сумма делилась на 3 и была наименьшей. Какую наименьшую сумму можно при этом получить?

Входные данные: Даны два входных файла: каждый из которых содержит в первой строке количество чисел N ($1 \leq N \leq 100000$). Каждая из следующих N строк содержит одно натуральное число, не превышающее 10^8 .

Пример входного файла:

5
5
4
13
7
10

Для указанных данных можно выбрать тройки 4, 13 и 7 (сумма 24), 4, 13 и 10 (сумма 27), 4, 7 и 10 (сумма 21) или 13, 7 и 10 (сумма 30). Наименьшая из сумм – 21. В ответе укажите два числа: сначала искомое значение для файла А, затем для файла В.