

Содержание

Распродажа баллов	2
Задача -1А. Лестница [0.1, 256]	2
Задача -1В. Tree height [0.2, 256]	3
Задача -1С. Ядра [0.25, 256]	4
Задача -1D. Спуск с горы [0.1, 256]	5
Задача -1Е. Экзамен по истории [0.4, 256]	6
Задача -1F. incrementator [0.6, 256]	7
Задача -1G. Мультиграф [0.2, 256]	8
Задача -1H. Дорешивание [0.4, 256]	9
Задача -1I. Сумма не без разнообразия [0.4, 256]	10
Задача -1J. Покрытие k отрезками [0.2, 256]	11

У вас не получается читать/выводить данные?

Воспользуйтесь примерами (**c++**) (**python**).

Обратите внимание, входные данные лежат в **стандартном потоке ввода** (он же stdin), вывести ответ нужно в **стандартный поток вывода** (он же stdout).

Обратите внимание на GNU C++ компиляторы с суффиксом inc.

Подни можно пользоваться **дополнительной библиотекой** (optimization.h).

То есть, использовать быстрый ввод-вывод: **пример про числа и строки**.

И быструю аллокацию памяти (ускоряет vector-set-map-весь-STL): **пример**.

Для тех, кто хочет разобраться, как всё это работает.

Короткая версия быстрого ввода-вывода (**тык**) и короткая версия аллокатора (**тык**).

Распродажа баллов

Задача -1А. Лестница [0.1, 256]

У лестницы n ступенек, пронумерованных числами $1, 2, \dots, n$ снизу вверх. На каждой ступеньке написано число. Начиная с подножия лестницы (его можно считать ступенькой с номером 0), требуется взобраться на самый верх (ступеньку с номером n). За один шаг можно подниматься на одну или на две ступеньки. После подъёма числа, записанные на посещённых ступеньках, складываются. Нужно подняться по лестнице так, чтобы сумма этих чисел была как можно больше.

Формат входных данных

В первой строке входного файла записано целое число n ($1 \leq n \leq 100$). Во второй строке заданы целые числа a_1, a_2, \dots, a_n через пробел ($-10\,000 \leq a_i \leq 10\,000$) — это числа, записанные на ступеньках.

Формат выходных данных

В первой строке выходного файла выведите одно число — максимальную сумму, которую можно получить, поднявшись по данной лестнице.

Примеры

stdin	stdout
2 1 2	3
2 2 -1	1
3 -1 2 1	3

Задача -1В. Tree height [0.2, 256]

You are given a rooted tree. Calculate its height, maximum number of vertices on path from the root to a leaf.

Формат входных данных

The first line contains number of vertices n ($1 \leq n \leq 10^5$). The second line contains n integer numbers from -1 to $n - 1$ – parents of vertices. If i -th one is -1 , vertex i is the root, else it's 0-based index of parent of i -th vertex. There is exactly one root.

Формат выходных данных

The only integer number – height of the tree.

Примеры

stdin	stdout
5 4 -1 4 1 1	3

Замечание

$O(n^2)$ solution will get TLE.

Задача -1С. Ядра [0.25, 256]

Капитан Вася всегда держит на своем корабле запас пушечных ядер для борьбы с пиратами. Так как он привык во всем поддерживать порядок, он хранит ядра в виде пирамид. Каждый из слоев одной пирамиды является равносторонним заполненным ядрами треугольником, сторона которого содержит ровно k ядер. Сторона основания пирамиды состоит из n ядер, в следующем слое сторона состоит из $n - 1$ ядра, и т.д., пока на вершину не будет положено одно ядро (которое является равносторонним треугольником со стороной 1).

Например, пирамида размера 3 состоит из трех уровней, выглядящих так (сверху вниз):

```
X  
  
X  
X X  
  
X  
X X  
X X X
```

Ясно, что каждый из треугольников может содержать только 1, 3, 6, 10 и т.д. ядер. Таким образом, пирамида может содержать только 1, 4, 10, 20, и т.д. ядер.

Вася отправляется в плавание и берет с собой ровно m ядер. Какое минимальное число пирамид требуется ему сложить из них на своем корабле?

Формат входных данных

В первой строке входного файла записано количество тестов $1 \leq T \leq 20$. В последующих T строках задается количество ядер в i -м тесте $1 \leq m_i \leq 300\,000$.

Формат выходных данных

Для каждого из T тестов входного файла выведите в отдельной строке минимальное количество пирамид.

Пример

stdin	stdout
5	1
1	2
5	3
9	3
15	2
91	

Задача -1D. Спуск с горы [0.1, 256]

В одном из горнолыжных курортов Италии проводятся соревнования по горнолыжному спуску. Каждому спортсмену предстоит скатиться с горы на лыжах. На любом этапе спуска участник получает определенное число очков. После прохождения трассы очки суммируются. Участник, набирающий наибольшее количество очков, выигрывает. Гора представляет собой треугольник, в качестве элементов которого выступают целые числа — очки за прохождение этапа. На каждом уровне спортсмену предоставляется выбор — двигаться вниз влево или вниз вправо. Начало спуска — в самой высокой точке горы, конец в одной из самых низких.

```
  1
 4 3
5 6 7
8 9 0 9
```

Требуется найти максимальное количество очков, которое может набрать спортсмен.

Формат входных данных

Во входном файле содержится целое число n — число этапов ($1 \leq n \leq 100$), далее n строк, каждая из которых характеризует свой уровень. В строке с номером i содержится ровно i целых чисел: a_1, a_2, \dots, a_i ($-100 \leq a_k \leq 100, 1 \leq k \leq i$) — количество очков в каждой из позиций.

Формат выходных данных

В результирующем файле должно находиться искомое целое число.

Пример

stdin	stdout
4	20
1	
4 3	
5 6 7	
8 9 0 9	

Задача -1Е. Экзамен по истории [0.4, 256]

Даны два списка дат.

Найти количество дат во втором списке, которые присутствуют в первом.

Формат входных данных

На первой строке длина первого списка N ($1 \leq N \leq 15\,000$).

Следующие N строк содержат целые числа от 1 до 10^9 – даты из первого списка.

Далее длина второго списка M ($1 \leq M \leq 10^6$).

Следующие M строк содержат целые числа от 1 до 10^9 – даты из второго списка.

Формат выходных данных

Одно число – ответ на задачу.

Примеры

stdin	stdout
2	2
1054	
1492	
4	
1492	
65536	
1492	
100	

Задача -1F. incrementator [0.6, 256]

Ваша задача — написать программу, моделирующую простое устройство, которое умеет прибавлять целые значения к целочисленным переменным.

Формат входных данных

Входной файл состоит из одной или нескольких строк, описывающих операции. Строка состоит из названия переменной и числа, которое к этой переменной надо добавить. Все числа не превосходят 100 по абсолютной величине. Изначально все переменные равны нулю. Названия переменных состоят из не более чем 100 000 маленьких латинских букв. Размер входного файла не превосходит 2 мегабайта.

Формат выходных данных

Для каждой операции выведите на отдельной строке значение соответствующей переменной после выполнения операции.

Примеры

stdin	stdout
a 2	2
b 3	3
a -1	1
c 4	4
b 17	20
xuz 23	23

Замечание

Стандартная структура из STL получит ОК. Но не `map`.

Задача -1G. Мультиграф [0.2, 256]

Дан неориентированный невзвешенный граф. В графе возможны петли и кратные рёбра. Постройте такой новый граф без петель и кратных рёбер, что для любых двух вершин в нём расстояние равно расстоянию в исходном графе. Если вершины не связны, расстояние между ними бесконечность.

Формат входных данных

На первой строке число вершин n и число рёбер m ($1 \leq n, m \leq 100\,000$). Следующие m строк содержат пары чисел от 1 до n – рёбра графа.

Формат выходных данных

Новый граф в таком же формате. Рёбра можно выводить в произвольном формате.

Примеры

stdin	stdout
3 5	3 3
1 1	1 2
1 3	2 3
2 1	3 1
1 2	
2 3	

Замечание

Не бойтесь графов. Это очень простая задача.

Задача -1Н. Дорешивание [0.4, 256]

Легенда

Как известно, после обеда в ЛКШ проходит много интересных мероприятий, но все равно каждый ЛКШонок старается хотя бы ненадолго заглянуть в комповник, чтобы дорешать задачи, не сделанные во время практики.

В этом году погода стоит особо жаркая, поэтому в комповнике очень душно и важно следить за тем, чтобы в комповнике не находилось одновременно очень много школьников. Поэтому завуч записал время прихода и ухода из комповника каждого ЛКШонка.

Теперь завуч хочет узнать, сколько ЛКШат встретил в комповнике каждый ЛКШонок.

Задача

Даны отрезки на прямой. Для каждого отрезка найти количество пересекающихся с ним.

Формат входных данных

В первой строке записано количество ЛКШат N ($1 \leq N \leq 10^5$). В i -й из следующих N строк через пробел записаны целые числа S_i и T_i ($0 \leq S_i \leq T_i \leq 10^9$) — время прихода в комповник и ухода из него i -го ЛКШонка.

Формат выходных данных

Программа должна вывести N целых чисел, i -е число должно быть равно количеству ЛКШат, которых встретил в комповнике i -й ЛКШонок.

Если в некоторый момент времени один ЛКШонок приходит в комповник, а другой уходит из неё, то они встречаются друг с другом.

Пример

stdin	stdout
4	3
1 10	3
2 5	2
5 6	2
1 4	

Задача -1I. Сумма не без разнообразия [0.4, 256]

Задана последовательность целых чисел A_1, A_2, \dots, A_N .

Необходимо выбрать из нее подпоследовательность из подряд стоящих чисел A_i, A_{i+1}, \dots, A_j так, чтобы она содержала не менее K различных чисел, и сумма $S = A_i + A_{i+1} + \dots + A_j$ была максимальной.

Формат входных данных

Первая строка ввода содержит целые числа N и K ($1 \leq K \leq N \leq 200\,000$).

Вторая строка содержит N целых чисел A_1, A_2, \dots, A_N ($|A_i| \leq 1\,000\,000\,000$).

Формат выходных данных

В первой строке необходимо вывести максимальное возможное значение суммы S . Во второй строке выведите индексы первого и последнего элементов найденной оптимальной подпоследовательности. Если существует несколько решений, подойдет любое из них.

Если не существует подпоследовательностей, удовлетворяющих решению задачи, выведите одну строку со словом “IMPOSSIBLE” (без кавычек).

Примеры

stdin	stdout
7 3 -99 1 2 -100 3 2 3	-89 2 7
3 2 1 1 1	IMPOSSIBLE

Задача -1J. Покрытие k отрезками [0.2, 256]

Даны n точек на прямой, нужно покрыть их k отрезками одинаковой длины ℓ .
Найдите минимальное ℓ .

Формат входных данных

На первой строке n ($1 \leq n \leq 10^5$) и k ($1 \leq k \leq n$). На второй n чисел x_i ($|x_i| \leq 10^9$).

Формат выходных данных

Минимальное такое ℓ , что точки можно покрыть k отрезками длины ℓ .

Примеры

stdin	stdout
6 2 1 2 3 9 8 7	2