

Задача А. Сортировка Хакуна Матата

Имя входного файла: `sort.in`
Имя выходного файла: `sort.out`
Ограничение по времени: 2 секунды
Ограничение по памяти: 256 мегабайт

После веселой прогулки по африканской долине Тимон и Пумба решили пообедать. Для этого они достали из-под камня n жуков, которые имеют различные целые массы от 1 до n и расставили их на ближайшем бревне. Чтобы обед прошел веселее, Пумба предложил есть жуков по возрастанию масс.

Чтобы отсортировать жуков по возрастанию масс, Тимон и Пумба используют сортировку Хакуна Матата. Суть сортировки заключается в том, что Тимон и Пумба выбирают жуков, отличающихся массой не более чем на единицу, после чего друзья меняют их местами. Например, если Пумба выбрал жука с массой 2, тогда Тимон может взять жука либо с массой 1, либо с массой 3. Так как друзья много раз пользовались этой сортировкой, им известно, что она всегда работает.

За всем этим процессом внимательно наблюдал Зазу. Он впервые видит нечто подобное, и ему интересно, как таким способом можно отсортировать жуков. Ваша задача — написать программу, которая по изначальному расположению жуков выведет последовательность ходов для сортировки жуков по возрастанию масс.

Формат входного файла

В самой первой строке написано число n — количество жуков ($1 \leq n \leq 100$). Во второй строке заданы n разделенных пробелами различных чисел m_i ($1 \leq m_i \leq n$), которые обозначают массу жука с номером i .

Формат выходного файла

Выведите на первой строчке выходного файла число t — количество ходов в возможной сортировке Хакуна Матата. На следующих t строчках выведите последовательность ходов сортировки Хакуна Матата таким образом, чтобы на строчке с номером s было выведено два числа i и j , что значит, что на шаге с номером s Пумба взял жука стоящего на позиции i , а Тимон на позиции j . Выведите любой способ, количество операций в котором не превышает 50000.

Пример

<code>sort.in</code>	<code>sort.out</code>
3	2
2 3 1	2 3
	2 1

Задача В. Математика

Имя входного файла: `math.in`
Имя выходного файла: `math.out`
Ограничение по времени: 2 секунды
Ограничение по памяти: 256 мегабайта

К сожалению, даже в дикой природе без математики никуда. Чтобы быть хорошим королем саванны, нужно быть не только сильным и ловким, но и уметь правильно распределить ресурсы, а тут никак не обойтись без сложения, умножения и деления. Это знал и Муфаса, и поэтому в один прекрасный день он заставил Симбу читать книгу по математике. Книга была очень сложной, а сложнее всего Симбе было со знаком $\hat{\cdot}$.

Дело в том, что в этой книге таким знаком обозначено две совсем разные вещи. Во-первых, таким знаком обозначили возведение в степень. Во-вторых — сноски. Сноски — это ссылки на объяснения некоторых кусков текста, которые приведены в конце книги. В этой книге после каждого символа $\hat{\cdot}$ следует некоторое число, после которого может быть либо знак препинания, либо пробел. Симба знает, что числа, которые стоят в сносках после символа $\hat{\cdot}$, являются последовательным натуральными числами. Число возле самой первой сноски равно единице. Про каждый символ $\hat{\cdot}$ можно сказать, является ли он возведением в степень или сноской.

Рассмотрим все возможные варианты разбиения знаков $\hat{\cdot}$ на степени и сноски. Будем считать, что авторы учебника постарались объяснить все как можно более подробно, а поэтому, корректными будем считать только те варианты, в которых количество сносок максимально.

Про каждый символ $\hat{\cdot}$ Симба хочет знать, может ли он являться возведением в степень в каком-нибудь из корректных разбиений. Аналогично, он хочет знать, может ли он являться сноской в каком-нибудь из корректных разбиений. Помогите ему!

Формат входного файла

В первой и единственной строке задан текст книги, который содержит не более 10^6 символов. Текст может состоять из строчных и заглавных латинских букв, цифр, символов $\hat{\cdot}$, а также точек, запятых, вопросительных знаков, пробелов и двоеточий. Гарантируется, что в тексте после каждого символа $\hat{\cdot}$ следует натуральное число (возможно, довольно большое). После каждого такого числа следует либо пробел, либо знак препинания (точка, запятая, двоеточие или вопросительный знак).

Формат выходного файла

В первой строке выведите количество символов $\hat{\cdot}$ в тексте. Во второй строке для каждого символа $\hat{\cdot}$ выведите Y , если он может являться степенью, и N в противном случае. Аналогично в третьей строке для каждого символа $\hat{\cdot}$ выведите Y , если он может являться сноской и N в противном случае.

Примеры

<code>math.in</code>	<code>math.out</code>
<code>sample: 2 plus 2 equals 2².¹ easy, ² isnt it? 1¹ equals 1.</code>	<code>4 YNNY NYYN</code>

Комментарий

Обратите внимание на то, что символы, стоящие перед $\hat{\cdot}$, никак не влияют на то, может ли этот символ являться сноской или возведением в степень.

Задача С. Родственные связи

Имя входного файла: `relatives.in`
Имя выходного файла: `relatives.out`
Ограничение по времени: 2 секунды
Ограничение по памяти: 256 мегабайта

Еще недавно королевство было совсем маленьким, все всех знали, и определение родственных связей не представляло никакой проблемы. Так сложилось, что в последние годы население резко выросло, и стало сложно определить кто кому кем приходится.

Мало кто знает, но у каждого зверя есть паспорт, в котором указан его номер — целое неотрицательное число. Паспортная система королевства хороша, но не идеальна, поэтому эти номера совсем не обязательно уникальны. Хороша она, собственно, тем, что если в десятичной записи номеров паспортов двух зверей есть хотя бы одна общая цифра, то они являются родственниками. Например, звери с номерами 47 и 107 — родственники, а с номерами 74 и 931 — нет.

Король Лев, как настоящий правитель, хочет знать все о своих подданных, поэтому просит вас посчитать количество различных пар зверей, которые являются родственниками.

Формат входного файла

Первая строка входного файла содержит число N ($1 \leq N \leq 500000$) - количество зверей в королевстве.

Вторая строка содержит n целых неотрицательных чисел, не превышающих 10^9 , разделенных пробелами — номера зверей.

Формат выходного файла

Выведите единственное число — ответ на задачу.

Примеры

<code>relatives.in</code>	<code>relatives.out</code>
5 10 74 47 77 301	4

Комментарий

Родственные связи есть у зверей 10 и 301, 74 и 47, 74 и 77, 47 и 77. Всего - 4 пары.

Задача D. Ужин из жуков

Имя входного файла: `supper.in`
Имя выходного файла: `supper.out`
Ограничение по времени: 2 секунды
Ограничение по памяти: 256 мегабайт

Тимон нашел много вкусных жуков под упавшим деревом и громко зовет гуляющего Пумбу на ужин.

Недалеко от Тимона и Пумбы находится искусственно высаженная роща. Деревья в этой роще растут только в точках с целыми координатами, которые лежат внутри и на границе некоторой окружности.

Пумба знает, что для того, чтобы попасть к Тимону и поужинать, ему придется перемещаться по прямой, соединяющей точки плоскости, в которых находится он и Пумба. Теперь он хочет знать, сколько деревьев из этой рощи находятся на одной прямой с ним, Тимоном и сытным ужином.

Формат входного файла

В первой строке содержатся три целых числа x, y, r ($1 \leq r \leq 10^9$) — координаты центра и радиус окружности рощи. Во второй строке расположены четыре целых числа x_t, y_t, x_p и y_p — координаты Тимона и Пумбы. Гарантируется, что ни Пумба, ни Тимон не находятся внутри рощи. Все координаты по модулю не превышают 10^9 .

Формат выходного файла

Выведите одно число — количество деревьев на пути Пумбы к Тимону.

Пример

<code>supper.in</code>	<code>supper.out</code>
0 0 5 -10 1 10 1	9

Комментарий

Обратите внимание на то, что Пумбу интересуют не только деревья, лежащие на отрезке между ним и Тимоном, но также и точки, лежащие вне этого отрезка, но на одной прямой с ними.

Задача E. Бутерброды из жуков

Имя входного файла: sandwiches.in
Имя выходного файла: sandwiches.out
Ограничение по времени: 2 секунды
Ограничение по памяти: 256 мегабайт

Сколько... Но питательно!

Король лев

Тимон и Пумба очень любят есть жуков, особенно бутерброды из них. А их лучший друг Симба относится к этому лакомству равнодушно. Поэтому Тимон и Пумба хотят доказать Симбе, что вкуснее бутербродов из жуков ничего нет. В дупле одного из деревьев они нашли n жуков и k кусков хлеба. Теперь они хотят сделать несколько бутербродов для Симбы.

Бутерброд делается следующим образом: кладется один кусок хлеба, сверху на него кладется один жук, на жука кладется еще один кусок хлеба и т.д. в итоге получится конструкция, в которой снизу лежит один кусок хлеба, дальше жуки и куски хлеба чередуются, причем наверху всей конструкции может лежать как жук, так и кусок хлеба. Тимон и Пумба считают, что если Симба съест бутерброд, в котором будет t жуков, то его удовлетворение увеличится на a_t . Они хотят, чтобы Симба получил от их бутербродов как можно большее удовлетворение, причем задействовать нужно всех жуков и все куски хлеба. Помогите им в этом нелегком деле!

Формат входного файла

В первой строке дано два целых числа n и k , где n — количество жуков, k — количество кусков хлеба ($1 \leq n \leq 500, 1 \leq k \leq 10^9$). Во второй строке содержится n чисел a_i ($1 \leq a_i \leq 10^7$).

Формат выходного файла

В единственной строке выходного файла выведите максимальное удовлетворение, которое Симба может получить, съев бутерброды. Если собрать бутерброды, использовав при этом всех жуков и хлеб, невозможно, выведите «Impossible».

Пример

sandwiches.in	sandwiches.out
3 3 1 3 2	4
3 2 1 2 3	Impossible

Задача F. Побег

Имя входного файла: `stdin`
Имя выходного файла: `stdout`
Ограничение по времени: 2 секунды
Ограничение по памяти: 256 мегабайта

Это интерактивная задача.

Шрам хочет остаться единственным претендентом на титул короля Саванны. Ради этого он коварно расправился с Муфасой, после чего приказал своим приспешникам-гиенам уничтожить Симбу. К счастью, Симбе улыбнулась удача, и ему предоставился шанс спастись от стаи разъяренных гиен.

Гиены преследуют Симбу на квадратном клетчатом поле, длина и ширина которого равны k клеткам. Как столбцы, так и строки этого поля пронумерованы числами от одного до k . Симба находится в одной из клеток поля, находящейся в первой строке. Стая гиен находится в некоторой клетке поля, не совпадающей с той, в которой находится Симба.

Симба и стая по очереди перемещаются по полю. За один ход Симба должен перебежать в одну из клеток, соседних по вертикали, горизонтали, или диагонали с той, в которой он находится. После этого гиены могут переместиться в любую клетку, расстояние до которой по горизонтали совпадает с расстоянием по вертикали. Проще говоря, гиены могут переместиться в любую клетку, находящуюся на одной диагонали с той, в которой они располагаются в данный момент. Заметим, что из этого определения следует, что они могут вообще не перемещаться и остаться в той же клетке.

Если после чьего-либо хода гиены оказываются в одной клетке с Симбой, будущий король погибает в неравной схватке. Если же после очередного хода львенка он оказывается в любой из клеток k -й строки, в которой нет гиен, он благополучно пересекает поле и скрывается в джунглях. Помогите Симбе сбежать от своих преследователей.

Протокол взаимодействия с программой жюри:

В первой строке программа жюри передает вашей программе одно целое число k ($3 \leq k \leq 100$) — размер поля, которое необходимо пересечь Симбе. В следующей строке программа жюри передает одно целое число b ($1 \leq b \leq k$) — номер столбца, в котором находится клетка с Симбой. В следующей строке программа жюри передает два целых числа x и y ($1 \leq x, y \leq k$) — номера строки и столбца клетки, в которой находится стая гиен. Гарантируется, что эта клетка не является клеткой с Симбой. После этого несколько раз повторяются следующие действия.

Ваша программа передает два целых положительных числа, не превышающих k — номера строки и столбца клетки, в которую необходимо переместиться Симбе. После этого программа жюри передает вашей программе два целых положительных числа, не превышающих k — номера строки и столбца клетки, в которую перемещается стая гиен.

В случае, если после очередного хода Симбы он оказался в клетке строки с номером k , в которой нет гиен, завершите программу — это будет означать успешный побег Симбы. За все время работы программы Симба должен сделать не более 1000 ходов.

Примеры

<code>stdin</code>	<code>stdout</code>
4	2 1
1	3 2
4 4	4 2
2 2	
3 1	

Комментарий

Для корректной работы программы после каждой операции вывода данных вам необходимо делать следующие операции:

- В языке Pascal: `flush(output);`
- В C/C++: `fflush(stdout);`
- В Java: `System.out.flush();`
- В Python: `sys.stdout.flush();`

Кроме этого, не забывайте после каждой выведенной строки ставить перевод строки.

Задача G. Стражи антилопы и гиены

Имя входного файла:	<code>stdin</code>
Имя выходного файла:	<code>stdout</code>
Ограничение по времени:	2 секунды
Ограничение по памяти:	256 мегабайт

Это интерактивная задача.

На ежегодное совещание по делам Саванны, проходящее в этом году во владениях Шрама, приглашён также и Муфаса. Муфаса считает, что путь из его владений во владения Шрама небезопасен и не намерен идти напрямик. Вместо этого, он решил поставить на своём пути антилоп, которые будут следить за безопасностью и, в случае непредвиденных проблем, заблаговременно предупредят Муфасу. Шрам прознал про это и решил помешать Муфасе организовать безопасный путь, приказывая своим гиенам занимать стратегически важные позиции, не отдавая их антилопам Муфасы.

Будем считать, что Саванна является клетчатым прямоугольником $n \times m$, владения Шрама находятся в клетке $(1, 1)$, а владения Муфасы — в клетке (n, m) . Владения Шрама и Муфасы уже считаются Муфасой безопасными. Шрам и Муфаса могут по очереди приказывать своим приспешникам (антилопам и гиенам) занимать некоторые клетки этого поля. Если некоторая клетка занята антилопами, она считается безопасной. Ни одна клетка, занятая гиенами, никогда не будет безопасной.

Муфаса хочет добиться того, чтобы из его владений во владения Шрама существовал путь, проходящий только по безопасным клеткам, каждые две последовательные клетки в котором являются соседними по стороне. Шрам же хочет не дать Муфасе возможности это сделать. Помогите Шраму добиться своего.

Протокол взаимодействия с программой жюри:

В первой строке входного файла заданы числа n и m ($3 \leq n, m \leq 20$) — размеры Саванны. Далее, до конца работы программы, выполняются следующие действия:

- Если вы считаете, что с данной расстановкой гиен для Муфасы не существует и уже никогда не будет существовать безопасного пути, выведите `x_X` и завершите работу программы.
- Иначе, выведите пару целых чисел (x, y) — координаты очередной гиены, такую, что:

- $1 \leq x \leq n, 1 \leq y \leq m$.
- $(x, y) \neq (1, 1)$.
- $(x, y) \neq (n, m)$.
- (x, y) не выводилось ранее ни программой участника, ни программой жюри.

После этого считайте со стандартного ввода пару чисел, соблюдающую такие же ограничения — координаты клетки, занятой очередной антилопой.

Ваша программа должна сделать не более $\lfloor \frac{nm}{2} \rfloor + 1$ действий, при этом, последним должен быть вывод `x_X`, и такой вывод должен быть всего один.

Пример

stdin	stdout
3 3	2 2
2 1	3 1
3 2	x_X
4 4	1 4
2 1	2 3
1 2	3 2
2 2	4 1
3 2	x_X

Комментарий

Для корректной работы программы после каждой операции вывода данных вам необходимо делать следующие операции:

- В языке Pascal: `flush(output);`
- В C/C++: `fflush(stdout);`
- В Java: `System.out.flush();`
- В Python: `sys.stdout.flush();`

Кроме этого, не забывайте после каждой выведенной строки ставить перевод строки.

Задача Н. Колевская бухгалтерия

Имя входного файла: `feed.in`
Имя выходного файла: `feed.out`
Ограничение по времени: 2 секунды
Ограничение по памяти: 256 мегабайта

Бухгалтерия в виде Зазу работала при Муфасе и будет работать при Симбе. Но когда Шрам запер Зазу в клетку, бухгалтерия ушла в вынужденный отпуск и потеряла все данные. И теперь Симба очень хочет их восстановить.

Бухгалтерия отвечала за количество антилоп и зебр в саванне. Кроме этого, она также считала количество останков животных. Одна единица останков образуется в результате поедания одной антилопы или зебры хищниками. Симба посчитал количество антилоп, зебр и останков в данный момент. Он точно знает, что с момента заточения Зазу не появилось ни одной новой зебры и ни одной новой антилопы, но некоторые зебры и антилопы могли быть съедены и превращены в останки. Кроме этого, Симба помнит, что в момент заточения Зазу антилоп было больше, чем зебр. Помогите ему посчитать количество различных наборов антилоп, зебр и останков, которые могли быть в момент заточения Зазу.

Формат входного файла

В строке заданы три целых числа a , b и c ($0 \leq a, b, c \leq 10^3$) — текущее количество антилоп, зебр и останков соответственно.

Формат выходного файла

В единственной строке выведите одно целое число k — количество возможных троек этих чисел в момент заточения Зазу.

Примеры

<code>feed.in</code>	<code>feed.out</code>
0 0 1	1
2 0 1	3

Комментарий

Ответом на первый тест из примера является единственная тройка чисел $(1, 0, 0)$.

Ответами на второй тест из примера являются следующие тройки чисел:

- $(2, 1, 0)$
- $(3, 0, 0)$
- $(2, 0, 1)$