

Задача А. Три буквы

Имя входного файла: стандартный ввод
Имя выходного файла: стандартный вывод
Ограничение по времени: 2 секунды
Ограничение по памяти: 256 мегабайт

Напомним, что строка $B = b_1b_2b_3\dots b_m$, является подпоследовательностью строки $A = a_1a_2a_3\dots a_n$, если существует строго возрастающая последовательность $\{i_1, i_2, i_3, \dots, i_m\}$ индексов A , такая, что для всех $j \in \{1, \dots, m\}$, выполняется $A_{i_j} = B_j$. Например, $B = "aba"$ является подпоследовательностью строки $A = "abacaba"$. Последовательность индексов в этом случае может быть такой: $\{1, 2, 3\}$.

Пусть Вам дана строка S состоящая только из маленьких букв латинского алфавита. Ваша задача заключается в том, чтобы посчитать количество ее подпоследовательностей «abc».

Формат входных данных

В единственной строке входного файла записана строка S , длиной не более 100 000 символов.

Формат выходных данных

В выходной файл выведите ответ на задачу.

Примеры

стандартный ввод	стандартный вывод
abc	1
ab	0

Задача В. Удаление скобок 2.0

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

Дана строка, составленная из круглых, квадратных и фигурных скобок. Определите, какое наименьшее количество символов необходимо удалить из этой строки, чтобы оставшиеся символы образовывали правильную скобочную последовательность.

Формат входных данных

Во входном файле записана строка из круглых, квадратных и фигурных скобок. Длина строки не превосходит 100 символов.

Формат выходных данных

Выведите максимальную длину строки, являющуюся правильной скобочной последовательностью, которую можно получить из исходной строки удалением некоторых символов.

Пример

<code>erase.in</code>	<code>erase.out</code>
<code>([)]</code>	2

Задача С. Удаление скобок 2.0

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

Дана строка, составленная из круглых, квадратных и фигурных скобок. Определите, какое наименьшее количество символов необходимо удалить из этой строки, чтобы оставшиеся символы образовывали правильную скобочную последовательность.

Формат входных данных

Во входном файле записана строка из круглых, квадратных и фигурных скобок. Длина строки не превосходит 100 символов.

Формат выходных данных

Выведите строку максимальной длины, являющейся правильной скобочной последовательностью, которую можно получить из исходной строки удалением некоторых символов.

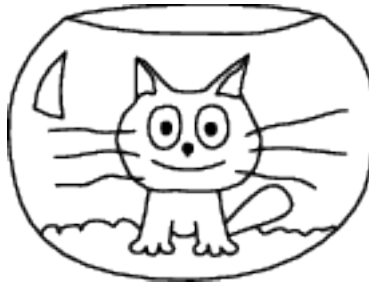
Пример

<code>erase.in</code>	<code>erase.out</code>
<code>([)]</code>	<code>[]</code>

Задача D. Продавец аквариумов

Имя входного файла: `aquarium.in`
Имя выходного файла: `aquarium.out`
Ограничение по времени: 1 секунда
Ограничение по памяти: 256 мегабайт

Продавец аквариумов для кошек хочет объехать n городов, посетив каждый из них ровно один раз. Помогите ему найти кратчайший путь.



Формат входных данных

Первая строка входного файла содержит натуральное число n ($1 \leq n \leq 13$) — количество городов. Следующие n строк содержат по n чисел — длины путей между городами.

В i -й строке j -е число — $a_{i,j}$ — это расстояние между городами i и j ($0 \leq a_{i,j} \leq 10^6$; $a_{i,j} = a_{j,i}$; $a_{i,i} = 0$).

Формат выходных данных

В первой строке выходного файла выведите длину кратчайшего пути. Во второй строке выведите n чисел — порядок, в котором нужно посетить города.

Примеры

aquarium.in	aquarium.out
5	666
0 183 163 173 181	4 5 2 3 1
183 0 165 172 171	
163 165 0 189 302	
173 172 189 0 167	
181 171 302 167 0	

Задача Е. Разбиение на пары

Имя входного файла: стандартный ввод
Имя выходного файла: стандартный вывод
Ограничение по времени: 2 секунды
Ограничение по памяти: 64 мегабайта

Однажды n ребят решили поучаствовать в VK Cup. Для этого им нужно разбиться на команды по два человека. Однако выяснилось, что не все ребята хорошо ладят друг с другом. Вы выяснили для каждой пары ребят, могут ли они участвовать в одной команде. Помогите им сформировать максимальное число команд.

Формат входных данных

Первая строка содержит число песен n ($1 \leq n \leq 18$), далее n строк по n символов описывают отношения ребят. Символ j строки i равен «Y», если ребята i и j могут участвовать в команде.

Формат выходных данных

Выведите одно число — максимальное число участников в командах, которые можно сформировать.

Примеры

стандартный ввод	стандартный вывод
5 NYYYY YNNNN YNNNN YNNNN YNNNN	2
5 NYYYY YNNNN YNNNY YNNNY YNYYN	4

Задача F. Простые пути в дереве

Имя входного файла: стандартный ввод
Имя выходного файла: стандартный вывод
Ограничение по времени: 2 секунды
Ограничение по памяти: 256 мегабайт

Дан неориентированный связный граф из n вершин и $n - 1$ ребра. Требуется для каждого ребра посчитать суммарную длину простых путей, проходящих через это ребро. Длиной пути здесь называется количество ребер в пути.

Формат входных данных

На первой строке целое число n ($2 \leq n \leq 300\,000$). Следующие $n - 1$ строка содержат пары чисел от 1 до n — ребра графа.

Формат выходных данных

$n - 1$ строка. i -я строка должна содержать целое число — ответ для i -го ребра.

Пример

стандартный ввод	стандартный вывод
5	13
1 2	8
2 3	8
2 4	9
5 1	

Задача G. Выбор вершин дерева

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

Дан граф, являющийся деревом. Множество вершин графа называется *допустимым*, если никакие две вершины этого множества не соединены ребром.

Рассмотрим все допустимые множества вершин графа. Для каждого такого множества посчитаем количество вершин в нём. Каково максимальное из этих количеств?

Формат входных данных

Граф в этой задаче задан в виде *корневого дерева*. В графе выделена вершина — *корень дерева*. Для каждой вершины i , не являющейся корнем, задан номер вершины-предка p_i в корневом дереве. Дерево, заданное таким образом, состоит из рёбер $i-p_i$ для всех вершин i , кроме корня.

В первой строке входного файла записано целое число n — количество вершин в графе ($1 \leq n \leq 100$). В следующих n строках задан граф. В i -й из этих строк записано целое число p_i — номер вершины-предка i -й вершины. Для корня дерева $p_i = 0$; для всех остальных вершин $1 \leq p_i \leq n$.

Гарантируется, что заданный во входном файле граф является деревом.

Формат выходных данных

В первой строке выходного файла выведите одно число — максимальное количество вершин в допустимом множестве.

Примеры

<code>select.in</code>	<code>select.out</code>
5 0 1 1 2 3	3
6 5 6 5 1 0 1	3

Задача Н. Дубы

Имя входного файла:	<code>oaks.in</code>
Имя выходного файла:	<code>oaks.out</code>
Ограничение по времени:	2 секунды
Ограничение по памяти:	256 мегабайт

На аллее перед зданием Министерства Обороны в ряд высажены n дубов. В связи с грядущим приездом главнокомандующего, было принято решение срубить несколько деревьев для придания аллее более милитаристического вида.

Внутренние распоряжки министерства позволяют срубить дуб только в двух случаях:

- если и ближайший дуб слева, и ближайший дуб справа строго ниже, чем данный дуб;
- если и ближайший дуб слева, и ближайший дуб справа строго выше, чем данный дуб.

В частности, согласно этому правилу, нельзя срубить крайний левый и крайний правый дубы.

Министр хочет выработать такой план вырубki, чтобы в итоге осталось несколько дубов, высоты которых образуют неубывающую последовательность, то есть чтобы каждый дуб был не ниже, чем все дубы, стоящие слева от него. При этом, как человек любящий флору, министр хочет, чтобы было срублено минимальное возможное количество деревьев.

Помогите сотрудникам министерства составить оптимальный план вырубki аллеи или выяснить, что срубить дубы соответствующим образом невозможно.

Формат входных данных

Первая строка входного файла содержит целое число n — количество дубов, растущих на аллее ($2 \leq n \leq 200$). Вторая строка содержит n чисел — высоты дубов, приведенные слева направо. Высоты дубов — положительные целые числа, не превышающие 1 000.

Формат выходных данных

Если оставить последовательность дубов с неубывающими высотами невозможно, выходной файл должен содержать только одно число -1 .

В случае, если искомый план существует, в первую строку выходного файла выведите целое число m — минимальное количество дубов, которые необходимо срубить. В следующие m строк выведите оптимальный план вырубki деревьев — номера дубов в том порядке, в котором их следует срубить, по одному номеру на строке.

Дубы нумеруются слева направо натуральными числами от 1 до n .

Если планов с наименьшим числом срубаемых дубов несколько, выведите любой из них.

Пример

<code>oaks.in</code>	<code>oaks.out</code>
5	2
3 2 4 8 5	4
	2

Задача I. Покраска забора

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

Мэр города Берендея решил построить перед своим домом забор из N деревянных досок и нанять лучшего маляра города для его покраски. Поскольку забор должен стать главной достопримечательностью города, лучший дизайнер города для каждой доски назначил тщательно выбранный цвет, в который она должна быть покрашена.

Для покраски главный маляр решил применить новейшую технологию, специально разработанную им для выполнения этого задания. Покраской забора будет заниматься специальный робот, который за один час может покрасить произвольный отрезок забора (набор соседних досок) в некоторый цвет. Поскольку задание должно быть выполнено как можно быстрее, требуется составить программу для робота, которая позволит достичь требуемой раскраски за минимальное время.

Оставить какую-то из досок непокрашенной, естественно, запрещается.

Формат входных данных

В первой строке входного файла записано число N ($1 \leq N \leq 300$), где N количество досок в заборе. Вторая строка содержит строку из N символов, описывающую требуемую покраску забора. Цвета обозначаются заглавными латинскими буквами.

Формат выходных данных

В первой строке выходного файла выведите M — наименьшее возможное время покраски забора в часах. Следующие M строк должны содержать программу покраски для робота. Каждая строка должна содержать два числа L_i и R_i , а также заглавную букву латинского алфавита, задающую цвет C_i и означает, что робот должен покрасить участок забора с L_i по R_i доску в цвет C_i (если длина забора N , должно выполняться $1 \leq L_i \leq R_i \leq N$).

Примеры

<code>paint.in</code>	<code>paint.out</code>
1	1
A	1 1 A
2	1
AA	1 2 A