

## Содержание

<b>Must have</b>	2
<b>Задача 8А. Почтовые отделения [1, 256]</b>	2
<b>Задача 8В. Гроб [1, 256]</b>	3
<b>Задачи здорового человека</b>	4
<b>Задача 8С. Выбор вершин взвешенного дерева [1, 256]</b>	4
<b>Задача 8D. Свертка [1, 256]</b>	5
<b>Задача 8Е. Шаблоны [1, 256]</b>	6
<b>Задача 8F. Восстановление [1, 256]</b>	7
<b>Для искателей острых ощущений</b>	8
<b>Задача 8G. Удаление скобок — 2 [1, 256]</b>	8
<b>Задача 8H. Свертка [1, 256]</b>	9

---

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**).

## Must have

### Задача 8А. Почтовые отделения [1, 256]

Вдоль прямой дороги расположены деревни. Дорога представляется целочисленной осью, а расположение каждой деревни задается одним целым числом — координатой на этой оси. Никакие две деревни не имеют одинаковых координат. Расстояние между двумя деревнями вычисляется как модуль разности их координат.

В некоторых, не обязательно во всех, деревнях будут построены почтовые отделения. Деревня и расположенное в ней почтовое отделение имеют одинаковые координаты. Почтовые отделения необходимо расположить в деревнях таким образом, чтобы общая сумма расстояний от каждой деревни до ближайшего к ней почтового отделения была минимальной.

#### Формат входных данных

В первой строке содержатся два целых числа: количество деревень  $n$  ( $1 \leq n \leq 300$ ) и количество почтовых отделений  $m$  ( $1 \leq m \leq 30$ ),  $m \leq n$ . Вторая строка содержит  $n$  целых чисел в возрастающем порядке, являющихся координатами деревень. Для каждой координаты  $x$  верно  $1 \leq x \leq 10^4$ .

#### Формат выходных данных

Первая строка выходного файла должна содержать одно целое число — общую сумму расстояний от каждой деревни до её ближайшего почтового отделения. Вторая строка должна содержать  $m$  целых чисел в возрастающем порядке. Эти числа являются искомыми координатами почтовых отделений. Если для заданного расположения деревень есть несколько решений, необходимо найти любое из них.

#### Пример

stdin	stdout
10 5	9
1 2 3 6 7 9 11 22 44 50	2 7 22 44 50

#### Подсказка по решению

Разобрана на практике.

### Задача 8В. Гроб [1, 256]

*Генри Гроб* — швейцарский шахматист, международный мастер. Шахматный литератор. Художник.

Википедия

Генри Гроб решил сыграть в новую игру, отдалённо напоминающую шахматы. Изначально фишка стоит в корне подвешенного дерева, затем игроки по очереди передвигают её. За ход необходимо переставить фишку в одного из детей текущей вершины. Проигрывает тот, кто не может ходить.

Генри начинает первым. Сможет ли он выиграть, если оба игрока играют оптимально?

#### Формат входных данных

Все числа в файле целые.

$1 \leq N \leq 10^5$  — количество вершин дерева.

Затем  $N$  чисел  $0 \leq p_i \leq N$  —  $i$ -ая вершина имеет родителя с номером  $p_i$ . Для корня дерева  $p_i = 0$ , для остальных вершин  $p_i \neq 0$ .

#### Формат выходных данных

«YES», если Гроб выигрывает, «NO» — в противном случае.

#### Пример

stdin	stdout
1 0	NO
2 0 1	YES
2 2 0	YES

#### Подсказка по решению

Динамика по дереву. «Кто выигрывает из данной вершины?»

## Задачи здорового человека

### Задача 8С. Выбор вершин взвешенного дерева [1, 256]

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

Рассмотрим все допустимые множества вершин графа. Для каждого такого множества вычислим сумму чисел, написанных в его вершинах. Какова максимальная из этих сумм?

#### 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 записаны через пробел два целых числа  $p_i$  и  $q_i$ ; здесь  $p_i$  — номер вершины-предка  $i$ -ой вершины, а  $q_i$  — число, записанное в этой вершине. Для корня дерева  $p_i = 0$ ; для всех остальных вершин  $1 \leq p_i \leq n$ . Числа  $q_i$  не превосходят по модулю 10 000.

Гарантируется, что заданный во входном файле граф является деревом.

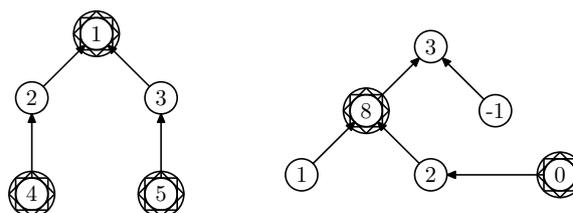
#### Формат выходных данных

В первой строке выходного файла выведите одно число — максимальную сумму чисел в допустимом множестве.

#### Примеры

stdin	stdout
<pre>5 0 1 1 2 1 3 2 4 3 5</pre>	10
<pre>6 5 8 6 0 5 -1 1 1 0 3 1 2</pre>	8

На рисунке показаны графы, заданные в примерах. В каждом графе выделено допустимое множество с максимальной суммой чисел в вершинах.



#### Подсказка по решению

Динамика по дереву. Похожая разобрана на практике.

### Задача 8D. Свертка [1, 256]

Петя хочет сократить запись последовательности, состоящей из заглавных латинских букв. Для этого он может свернуть ее повторяющиеся подпоследовательности. Например, последовательность `AAAAAAAAABABABCCD` может быть записана как `10(A)2(BA)B2(C)D`.

Формальное определение свернутой последовательности и соответствующей ей операции развертки дается следующим образом:

- Последовательность, которая содержит единственный символ от 'A' до 'Z' представляет из себя свернутую последовательность. При развертке такой последовательности получается она сама.
- Если  $S$  и  $Q$  — свернутые последовательности, то  $SQ$  также свернутая последовательность. Если при развертке строки  $S$  получается строка  $S'$ , а при развертке  $Q$  получается  $Q'$ , то при развертке  $SQ$  получается строка  $S'Q'$ .
- Если  $S$  — свернутая последовательность, то  $X(S)$  также свернутая последовательность, где  $X$  это десятичное представление целого числа большего единицы. Если при развертке строки  $S$  получается строка  $S'$ , то при развертке  $X(S)$  получается строка  $S'$ , повторенная  $X$  раз.

Петя хочет свернуть заданную последовательность таким образом, чтобы результат содержал наименьшее число символов.

#### Формат входных данных

Входной файл содержит непустую строку, состоящую из заглавных латинских букв. Длина строки не превышает 100 символов.

#### Формат выходных данных

В выходной файл выведите одну строку, содержащую наименьшую последовательность развертка которой даст строку, заданную во входном файле.

Если ответов несколько — выведите любой из них.

#### Пример

stdin	stdout
AAAAAAAAABABABCCD	9(A)3(AB)CCD
NEERCYESYESYESNEERCYESYESYES	2(NEERC3(YES))

#### Подсказка по решению

Разобрана на практике. Динамика по подотрезкам.

Восстановление ответа — рекурсивная функция.

### Задача 8Е. Шаблоны [1, 256]

Многие операционные системы используют шаблоны для ссылки на группы объектов: файлов, пользователей, и т. д. Ваша задача — реализовать простейший алгоритм проверки шаблонов для имен файлов.

В этой задаче алфавит состоит из маленьких букв английского алфавита и точки ('.'). Шаблоны могут содержать произвольные символы алфавита, а также два специальных символа: '?' и '\*'. Знак вопроса ('?') соответствует ровно одному произвольному символу. Звездочка '\*' соответствует подстроке произвольной длины (возможно, нулевой). Символы алфавита, встречающиеся в шаблоне, отображаются на ровно один такой же символ в проверяемой строке. Строка считается подходящей под шаблон, если символы шаблона можно последовательно отобразить на символы строки таким образом, как описано выше. Например, строки "ab", "aab" и "beda." подходят под шаблон "\*a?", а строки "bebe", "a" и "ba" — нет.

#### Формат входных данных

Первая строка входного файла определяет шаблон  $P$ . Вторая строка  $S$  состоит только из символов алфавита. Ее необходимо проверить на соответствие шаблону. Длины обеих строк не превосходят 10 000. Строки могут быть пустыми — будьте внимательны!

#### Формат выходных данных

Если данная строка подходит под шаблон, выведите YES. Иначе выведите NO.

#### Примеры

stdin	stdout
k?t*n kitten	YES
k?t?n kitten	NO

#### Подсказка по решению

Разобрана на первой практике по динамике.

Вам нужна квадратная динамика с линией памяти.

Как сделать линию памяти? Хранить только 1 строку динамики.

### Задача 8F. Восстановление [1, 256]

Денис обнаружил ошибку в своей программе, которая должна удалять все символы из строки кроме “(” и “)”. Оказывается, некоторые скобки заменяются на что-то нечитаемое.

Теперь его заинтересовал вопрос, сколько различных правильных скобочных последовательностей могут являться результатом правильного алгоритма.

#### Формат входных данных

Единственная строка входного файла содержит строку из круглых скобок и знаков вопроса, где вопросами обозначены утраченные символы. Вопрос можно заменить на ровно одну любую скобку. Длина строки не превосходит 10 000, но может быть нечетной.

#### Формат выходных данных

Выведите одно число — количество различных скобочных последовательностей, удовлетворяющих шаблону Дениса, по модулю  $10^9 + 7$ .

#### Пример

stdin	stdout
(??( )?)	2

#### Подсказка по решению

Вам нужна квадратная динамика с линией памяти.

Как сделать линию памяти? Хранить только 1 строку динамики.

## Для искателей острых ощущений

### Задача 8G. Удаление скобок — 2 [1, 256]

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, являющуюся правильной скобочной последовательностью, которую можно получить из исходной строки удалением некоторых символов. Если возможных ответов несколько, выведите любой из них.

#### Примеры

stdin	stdout
([])	[]
{([[]{}])}	([]{})
]{}[	

#### Подсказка по решению

Динамика по подотрезкам.

### Задача 8Н. Свертка [1, 256]

Это такая же задача, но с более высокими ограничениями.

#### Формат входных данных

Входной файл содержит непустую строку, состоящую из заглавных латинских букв. Длина строки **не превышает 1000 символов**.

#### Формат выходных данных

В выходной файл выведите одну строку, содержащую наименьшую последовательность развертка которой даст строку, заданную во входном файле.

Если ответов несколько — выведите любой из них.

#### Пример

stdin	stdout
AAAAAAAAAABABABCCD	9(A)3(AB)CCD
NEERCYESYESYESNEERCYESYESYES	2(NEERC3(YES))

#### Подсказка по решению

Всё ещё динамика по подотрезкам. Аккуратно написанный  $\mathcal{O}(n^3)$  зайдёт.