## RegDiv3. Учимся делить на три!

В условиях данной задачи разрешим в двоичной записи чисел присутствие ведущих нулей. Кроме того, разрешим записывать число 0 в двоичной записи как $\epsilon$ (то есть как пустую строку).

Таким образом, строки «101» и «000101» являются двоичными записями числа 5, а строки «000000» и «» — двоичными записями числа 0.

Выведите регулярное выражение, проверяющее, что данное число, записанное в двоичной записи, делится на 3.

Например, регулярное выражение «(0|11)∗» правильно работает для записей двоичных чисел от 0 до 8, но дает неправильный результат для числа 9.

## RegExp. Регулярное выражение

Имя входного файла: `regexp.in`
Имя выходного файла: `regexp.out`

Дано регулярное выражение $R$ над алфавитом {«a», «b»}.

Постройте минимальный ДКА, принимающий тот же язык, что и $R$.

### Формат входного файла

Входной файл содержит регулярное выражение $R$. В записи регулярного выражения могут использоваться следующие конструкции:

- «a», «b» — задает язык, состоящий из одного слова — соответствующей буквы;
- «0» (цифра «ноль») — задает язык $\emptyset$ (пустое множество);
- «e» — задает язык $\{\epsilon\}$, состоящий из одного слова — пустого;
- $(x)$ — задает тот же язык, что и регулярное выражение $x$;
- $x*$ — задает замыкание Клини языка, задаваемого регулярным выражением $x$;
- $xy$ — задает конкатенацию языков, задаваемых регулярными выражениями $x$ и $y$;
- $x|y$ — задает язык, состоящий из объединения языков, задаваемых $x$ и $y$.

Операции перечислены в порядке уменьшения приоритета.

Длина регулярного выражения $R$ не превышает 20 символов.

### Формат выходного файла

Выведите описание минимального ДКА, эквивалентного регулярному выражению $R$, в формате, описанном в задаче «MinDFA», но без первой строки (строки с алфавитом).

### Пример

| regexp.in | regexp.out |
|---|---|
| (a\|b)*(aa\|bb)(a\|b)* | 4<br>4 1 3<br>3 2<br>1 3<br>3 3<br>1 2 |
| (a\|e)(ba)*(b\|e) | 4<br>4 3 1 2 4<br>3 2<br>1 3<br>3 3<br>1 2 |
| ab0ab | 1<br>1 0<br>1 1 |

### Замечание

Первое регулярное выражение задает язык, состоящий из строк, в которых встречаются две одинаковые буквы подряд. Второе выражение — язык из строк, в которых не встречаются две одинаковые буквы подряд. Третье регулярное выражение задает пустой язык.