Электрические схемы орбитальной станции

Много лет назад космическая станция была одним из самых совершенных творений человека. Это позволило отслеживать активность на Земле и проводить важные испытания для будущего человечества в космосе. Но в какой-то момент у станции начались технические проблемы. Команда станции начала посылать радиосигналы на Землю для диагностики проблем. Но из-за сильных электромагнитных помех передачи часто искажались, и команда на Земле не могла их легко прочитать. Они поняли, что им нужно найти решение для кодирования сигналов перед их отправкой. После нескольких дней исследований они пришли к выводу, что система кодирования с использованием логических выражений будет эффективным решением. И диагностировать работу станции можно будет, оценивая эти логические выражения.

В логических выражениях используются только следующие обозначения: `&&` - логическое И, `||` - логическое ИЛИ, `!` - логического ОТРИЦАНИЯ, 1 – для TRUE, 0 – для FALSE. Приоритеты вычислений: ОТРИЦАНИЕ (!) – операция III порядка, И (`&&`) – операция II порядка, наконец, ИЛИ (`||`) – операция I порядка (высшая); вначале выполняются операции низшего порядка, но также и слева направо, в том порядке, в котором они записываются. Пример выражения: `1&&0||1&&!1`.

Задание. Разработать программу, которая помогала бы инженерам диагностировать работу станции, вычисляя логические выражения.

Входные данные. Файл `circuite.in` содержит целое число `N`, представляющее количество логических выражений, которые нужно вычислить. Каждое выражение размещается на новой строке и состоит только из разрешенных обозначений.

Выходные данные. Файл `circuite.out` содержит N строк. В каждой строке отображается 0 или 1 в зависимости от конечного значения выражения, а после значения выражения выводится пробел и пара круглых скобок, внутри которых находится количество операций в данном выражении.

Ограничения. $1 \leq N \leq 10^5$. Выражение может содержать не более 100 символов. Время выполнения не должно превышать `1` секунду. Программа должна использовать не более `8` мегабайт оперативной памяти. Исходный файл должен называться `circuite.pas`, `circuite.c` или `circuite.cpp`.

Пример.

`circuite.in`

```
3
1&&0||1
0||1||0
!0&&1&&!1
```

`circuite.out`

```
1 (2)
1 (2)
0 (4)
```

Объяснение. Первое выражение содержит две операции. Первой операцией будет операция второго порядка – 1&&0, результатом которой будет 0. Следующая операция || будет производиться между полученным значением 0 и 1, а конечным результатом будет 1. Во втором выражении имеем две операции того же порядка - || , поэтому выполняем операции слева направо: $[(0||1)_1||0]_2$ получаем итоговую 1. Последнее выражение содержит 4 операции (два отрицания ! и две логические конъюнкции &&), выполняемые по схеме, указанной в индексах скобок, которые группируют операцию $\{[\ (!0)_1\ \&\&\ 1\ ]_3\ \&\&\ (!1)_2\ \}_4$ получая значение 0.