

Задача А. Разделение выражения на лексемы

Имя входного файла: `lexem.in`
 Имя выходного файла: `lexem.out`
 Ограничение по времени: 2 секунды
 Ограничение по памяти: 64 мегабайта

Задано числовое выражение, заканчивающееся точкой. Необходимо разбить его на лексемы и вывести каждую на новой строке. Гарантируется, что исходное выражение корректно.

В выражении могут встречаться знаки сложения, вычитания, умножения и скобки. Приоритет операций стандартный. Все числа в выражении целые и принадлежат диапазону `LongInt`.

Формат входного файла

Первая строка входного файла содержит заданное выражение. Его длина не превосходит 100 знаков. Гарантируется, что выражение заканчивается точкой.

Формат выходного файла

Выведите все встречающиеся лексемы выражения по порядку и ровно по одной на каждой строке.

Пример

lexem.in	lexem.out
1+(2*2-3).	1
	+
	(
	2
	*
	2
	-
	3
)

Задача В. Значение логического выражения

Имя входного файла: `logexpr.in`
 Имя выходного файла: `logexpr.out`
 Ограничение по времени: 2 секунды
 Ограничение по памяти: 64 мегабайта

Задано логическое выражение, заканчивающееся точкой. Необходимо посчитать его значение или сказать, что оно содержит ошибку. В выражении могут встречаться знаки ! (отрицание), & (логическое «и»), | (логическое «или»), ^ (XOR) и скобки. Самый большой приоритет у отрицания, чуть меньше у &, у оставшихся операций приоритет одинаково маленький. Все числа в выражении либо 0, либо 1.

Формат входного файла

Первая строка входного файла содержит заданное выражение. Его длина не превосходит 100 знаков. Гарантируется, что выражение заканчивается точкой.

Формат выходного файла

Выведите в выходной файл значение этого выражения (0 или 1).

Пример

logexpr.in	logexpr.out
1 (0&0^1).	1

Задача С. Значение выражения

Имя входного файла: `expr.in`
 Имя выходного файла: `expr.out`
 Ограничение по времени: 2 секунды
 Ограничение по памяти: 64 мегабайта

Задано числовое выражение, заканчивающееся точкой. Необходимо посчитать его значение или сказать, что оно содержит ошибку. В выражении могут встречаться знаки сложения, вычитания, умножения и скобки. Приоритет операций стандартный. Все числа в выражении целые и принадлежат диапазону `LongInt`. Также гарантируется, что все промежуточные вычисления уместятся в этот тип.

Формат входного файла

Первая строка входного файла содержит заданное выражение. Его длина не превосходит 100 знаков. Гарантируется, что выражение заканчивается точкой.

Формат выходного файла

Выведите в выходной файл значение этого выражения или слово «WRONG», если значение не определено.

Пример

expr.in	expr.out
1+(2*2-3).	2
1+a+1.	WRONG

Задача D. Новогоднее выражение

Имя входного файла: `expr2.in`
 Имя выходного файла: `expr2.out`
 Ограничение по времени: 2 секунды
 Ограничение по памяти: 64 мегабайта

Задано новогоднее выражение, заканчивающееся точкой. Необходимо посчитать его значение или сказать, что оно содержит ошибку. В новогоднем выражении могут встречаться знаки сложения, вычитания, умножения и скобки, константы Ded Moroz, Moroz и Snegurochka, а также вызов функции Podarok, которая принимает одно число на вход и возвращает его, увеличенное на 5, если оно было положительно, или возвращает его модуль, если оно было меньше либо равно 0. Приоритет операций стандартный. Все числа в выражении целые и принадлежат диапазону `LongInt`. Также гарантируется, что все промежуточные вычисления уместятся в этот тип.

Формат входного файла

Первая строка входного файла содержит заданное выражение. Его длина не превосходит 200 знаков. Гарантируется, что выражение заканчивается точкой.

Значения констант:

Ded Moroz	2010
Moroz	-30
Snegurochka	10

Формат выходного файла

Выведите в выходной файл значение этого новогоднего выражения или слово «WRONG», если значение не определено.

Пример

expr2.in	expr2.out
Podarok(Moroz-Ded Moroz)*2.	4080
Snegurochka-30.	-20