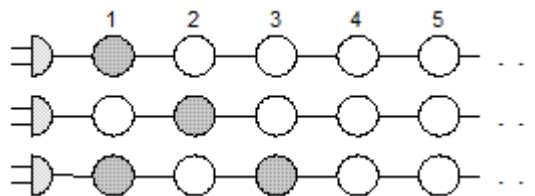


Задача А. Гирлянда

Имя входного файла: input.txt
Имя выходного файла: output.txt
Ограничение по времени: 2 секунды
Ограничение по памяти: 64 мегабайта

Новогодняя елка украшена гирляндой бесконечной длины, которая состоит из последовательно соединенных лампочек. Когда гирлянду включают, загорается только первая лампочка, считая от выключателя, которая горит одну секунду. Далее гирлянда начинает мигать по следующему правилу. Каждую секунду для каждой лампочки проверяется условие: если ровно одна из ее соседних лампочек горит, то эта лампочка будет гореть на следующей секунде; иначе – не будет гореть. У первой лампочки только одна соседняя.



Напишите программу, которая по номеру секунды находит количество лампочек гирлянды, которые будут гореть на протяжении этой секунды.

Формат входного файла

Единственная строка входного файла содержит одно целое число N ($1 \leq N \leq 10^9$) – номер секунды.

Формат выходного файла

Единственная строка выходного файла должна содержать целое число – количество лампочек, которые будут гореть на секунде N .

Примеры входных и выходных файлов

Входные данные	Выходные данные
7	4
5	2
20	2

Задача В. Billing

Имя входного файла: input.txt
Имя выходного файла: output.txt
Ограничение по времени: 2 секунды
Ограничение по памяти: 256 мегабайт

Девочка Катя подключилась к тарифу «Очень выгодный», на котором можно только звонить. Все входящие звонки бесплатны. В случае исходящего звонка не более k_1 первых секунд звонка стоят p_1 копеек, и позвонить можно только если эти деньги на счету есть. За следующие k_2 секунд Катя платит по p_2 копеек за секунду, а все остальное время девочка платит по p_3 копеек за секунду. Деньги снимаются мгновенно после каждой секунды. Как только баланс становится неположительным, связь обрывается. Известно, что Катя положила N копеек на счет, чтобы поговорить со своим лучшим другом. Причем, она хочет потратить все N копеек на этот один телефонный звонок. Посчитайте, сколько максимально секунд Катя сможет наслаждаться беседой.

Формат входного файла

Во входном файле записаны через пробел 6 целых чисел: $0 \leq N \leq 1000000$, $1 \leq k_1, k_2 \leq 1000000$, $1 \leq p_1, p_2, p_3 \leq 1000000$.

Формат выходного файла

В выходной файл выведите одно число: максимальное количество секунд, которое при заданных условиях могла выговорить девочка Катя в течение одного телефонного разговора.

Примеры

Входные данные	Выходные данные
20 3 3 3 4 2	9
3 15 3 5 2 3	0
2 1 1 1 3 1	2

Задача С. Определите порядок действий

Имя входного файла: input.txt
Имя выходного файла: output.txt
Ограничение по времени: 1 секунда
Ограничение по памяти: 64 мегабайта

Ученику второго класса рассказали правила, как нужно выполнять арифметические действия, чтобы вычислить значение арифметического выражения, состоящего из чисел, скобок и знаков арифметических операций + (сложение) и * (умножение). После этого ему дали упражнения — несколько задач, в которых требуется расставить порядок выполнения действий. Помогите ему.

Правила вычисления выражения, рассказанные ученику, звучат так. Если в выражении вообще нет скобок, то сначала выполняются все операции умножения слева направо, а затем — операции сложения также слева направо. Если же в выражении есть скобки, то находится самая левая закрывающая скобка и соответствующая ей открывающая. Выражение между ними не содержит скобок и может быть вычислено по вышеописанным правилам. Дальше это выражение (вместе со скобками) мысленно удаляется из выражения и заменяется числом – результатом. Если в выражении остались скобки, то процедура повторяется.

Напишите программу, которая для корректного выражения будет определять порядок выполнения арифметических действий. Поскольку сами числа в этой задаче нам будут не существенны, мы заменим их на знаки #.

Формат входного файла

Во входном файле записана одна строка, состоящая из символов #, +, *, (,). Длина строки не превышает 250 символов. Строка соответствует правильному арифметическому выражению.

Формат выходного файла

В выходной файл нужно вывести ту же строку, заменив знаки операций + и * в ней натуральными числами, задающими порядок выполнения действий в соответствии с описанными правилами.

Примеры входных и выходных файлов

Входные данные	Выходные данные
##+*#	#2#1#
###+(##)	#2#3(#1#)
#+(##*#)*##	#4(#2#1#)3#5#

Задача D. Сдача

Имя входного файла: input.txt
Имя выходного файла: output.txt
Ограничение по времени: 5 секунд
Ограничение по памяти: 64 мегабайта

Покупатель хочет приобрести товар стоимостью S рублей. У него есть N банкнот номиналом P_1, P_2, \dots, P_N рублей. У продавца есть M банкнот номиналом Q_1, Q_2, \dots, Q_M рублей. Определите, смогут ли они рассчитаться.

Формат входного файла

Программа получает на вход сумму S . Далее идет число N – количество банкнот у покупателя, затем P_1, P_2, \dots, P_N – номиналы этих банкнот. Далее идет число M – количество банкнот у продавца, затем Q_1, Q_2, \dots, Q_M – номиналы этих банкнот. Количество банкнот у продавца и покупателя и их номиналы не превосходят 100.

Формат выходного файла

Если продавец сможет рассчитаться с покупателем, выведите наименьшее количество банкнот, которое придется отдать продавцу на сдачу.

Если они не могут рассчитаться, выведите число -1 (минус 1)

Примеры входных и выходных файлов

Входные данные	Выходные данные
10 4 3 3 9 3 2 6 2	1
100 3 74 35 8 2 19 6	-1

Задача Е. Вслепую по лабиринту

Имя входного файла: input.txt
Имя выходного файла: output.txt
Ограничение по времени: 4 секунды
Ограничение по памяти: 64 мегабайта

Это интерактивная задача.

Ваша цель — написать программу, управляющую роботом, идущим вслепую по лабиринту. Лабиринт состоит из N на M ($1 \leq N, M \leq 30$) клеток. Каждая из клеток может быть свободной или заблокированной (непроходимой). Все клетки на границе лабиринта непроходимые. Робот начинает работу в свободной клетке, он может переместиться на юг, запад, север или восток в свободную клетку. Робот не имеет оптических сенсоров, только сенсор удара, так что при попытке перемещения в заблокированную клетку сработает сенсор, и робот останется в той же клетке. Робот должен побывать во всех проходимых клетках лабиринта. Из начальной клетки гарантированно можно попасть во все достижимые клетки лабиринта.

Программа должна выводить на стандартный вывод одну строку с действием робота и ждать строки в стандартном вводе с ответом, затем выводить очередную строку с действием и считывать ответ и так далее до тех пор, пока все проходимые клетки лабиринта не будут посещены. Программа должна завершать работу только когда все проходимые клетки будут посещены. Проходимые клетки могут быть посещены несколько раз. Допустимо передвигаться даже если все проходимые клетки уже посещены.

Формат входного файла

Каждая строка входных данных представляет собой ответ на действие робота. Это может быть строка EMPTY если робот успешно переместился в заданном направлении или строка BLOCKED если робот не смог переместиться из-за того, что клетка, в которую он хотел попасть, непроходима.

Формат выходного файла

Каждая строка выходных данных должна представлять собой команду для робота. Это должна быть одна из пяти возможных строк: SOUTH, WEST, NORTH, EAST, или DONE. Строка DONE должна быть напечатана после посещения роботом всех проходимых клеток. После вывода DONE программа должна завершать свою работу. Необходимо очищать выходной буфер после вывода каждой команды (flush).

Примеры входных и выходных файлов

Входные данные	Выходные данные
NORTH	BLOCKED
EAST	BLOCKED
SOUTH	EMPTY
EAST	BLOCKED
SOUTH	BLOCKED
WEST	EMPTY
SOUTH	BLOCKED
WEST	BLOCKED

NORTH WEST WEST NORTH EAST NORTH DONE	EMPTY EMPTY BLOCKED BLOCKED EMPTY BLOCKED
---	--