

## Задача 1. Программирование квадрокоптеров

Имя входного файла:	стандартный ввод
Имя выходного файла:	стандартный вывод
Ограничение по времени:	2 секунды
Ограничение по памяти:	512 мегабайт

Школьники готовятся к участию в соревновании по программированию квадрокоптеров. Квадрокоптер, который используется в соревновании, может выполнять две команды: подняться вверх на 1 метр и опуститься вниз на 1 метр. Команда подъёма обозначается символом «(», а команда спуска — символом «)».

Программа для квадрокоптера представляет собой последовательность команд. Программа считается *корректной*, если, начав её исполнение на уровне земли и выполнив последовательно все команды, квадрокоптер снова оказывается на уровне земли. При этом в процессе выполнения программы квадрокоптер не должен пытаться опуститься ниже уровня земли.

Например, следующие программы являются корректными: «( ) ( )», «( ( ( ) ) )». Программа «( ( ( (» не является корректной, поскольку квадрокоптер завершает её выполнение на высоте 3 метра над уровнем земли, программа «( ) ) (» также не является корректной, поскольку при выполнении третьей команды квадрокоптер пытается опуститься ниже уровня земли.

Участник соревнования написал корректную программу для квадрокоптера, состоящую из  $n$  команд, пронумерованных от 1 до  $n$ . Он загрузил её в память квадрокоптера для демонстрации во время соревнования. К сожалению, после загрузки программы в память квадрокоптера участник случайно удалил её на своём компьютере, а квадрокоптер не позволяет выгрузить программу из своей памяти.

К счастью, квадрокоптер поддерживает специальный режим отладки программы. В этом режиме квадрокоптер с загруженной в него программой может отвечать на специальные запросы. Каждый запрос представляет собой два целых числа:  $l$  и  $r$ ,  $1 \leq l \leq r \leq n$ . В ответ на запрос квадрокоптер сообщает, является ли фрагмент загруженной в него программы, состоящий из команд с  $l$ -й по  $r$ -ю включительно, корректной программой для квадрокоптера, либо нет. Участник хочет с помощью режима отладки восстановить загруженную в квадрокоптер программу.

Требуется написать программу-решение, которая взаимодействует с программой жюри, моделирующей режим отладки квадрокоптера, и в итоге восстанавливает загруженную в квадрокоптер программу.

### Протокол взаимодействия

Это интерактивная задача.

Сначала на вход подаётся целое число  $n$  — количество команд в программе квадрокоптера ( $2 \leq n \leq 50\,000$ ).

Для каждого теста жюри зафиксировано число  $k$  — максимальное количество запросов. Гарантируется, что  $k$  запросов достаточно, чтобы решить задачу. Это число не сообщается программе-решению. Ограничения  $k$  в различных подзадачах приведены в таблице системы оценивания. Если программа-решение делает более  $k$  запросов к программе жюри, то на этом тесте она получает в качестве результата тестирования «Неверный ответ».

Чтобы сделать запрос, программа-решение должна вывести строку вида «?  $l$   $r$ », где  $l$  и  $r$  — целые положительные числа, задающие фрагмент программы квадрокоптера ( $1 \leq l \leq r \leq n$ ).

В ответ на запрос программы-решения программа жюри подаёт ей на вход либо строку «Yes», либо строку «No», в зависимости от того, является ли запрошенный фрагмент программы квадрокоптера корректной программой.

Если программа-решение определила ответ на задачу, то она должна вывести строку «!  $c_1 c_2 \dots c_n$ », где символ  $c_i$  задаёт  $i$ -ю команду в программе квадрокоптера и равен либо «(», либо «)».

После этого программа-решение должна завершиться.

Гарантируется, что в каждом тесте программа в памяти квадрокоптера является фиксированной корректной программой, которая не меняется в зависимости от запросов, произведённых программой-решением.

## Примеры

стандартный ввод	стандартный вывод
4	? 1 4
Yes	? 1 3
No	? 1 2
Yes	? 3 4
Yes	! ( ) ( )
6	? 3 4
Yes	? 1 2
No	? 2 5
Yes	! (( ( ) ))

## Пояснения к примерам

Приведённые примеры иллюстрируют взаимодействие программы-решения с программой жюри «по шагам», для чего в них добавлены дополнительные пустые строки. При реальном тестировании лишние пустые строки вводиться не будут, выводить пустые строки также не требуется.

В первом примере  $n = 4$ . Единственная возможная корректная программа из двух команд это «()», поэтому из результатов третьего и четвёртого запросов можно сделать вывод, что программа в памяти квадрокоптера — «() ( )». Поэтому, в частности, ответ на второй запрос действительно «No», так как фрагмент программы «() (» не является корректной программой: если квадрокоптер исполнит именно эти три команды, он останется на уровне одного метра над землёй.

В втором примере  $n = 6$ , и произведённых запросов достаточно, чтобы однозначно определить, что программа в памяти квадрокоптера — «(( ( ) ))».

В тестах из условия  $k = 150$ , то есть, разрешается произвести не более 150 запросов.

## Система оценивания

Подзадача	Баллы	Ограничения		Необходимые подзадачи	Результаты во время тура
		$n$	$k$		
1	21	$2 \leq n \leq 16$	$k = 150$		Потестовые
2	28	$2 \leq n \leq 100$	$k = 10\,000$	1	Потестовые
3	26	$2 \leq n \leq 1000$	$k = 10\,000$	1, 2	Потестовые
4	25	$2 \leq n \leq 50\,000$	$k = 100\,000$	1 – 3	Потестовые