

Задача А. Пинбол в треугольнике Паскаля

Ограничение по времени: 2 секунды
Ограничение по памяти: 64 Мб

Пинбол в треугольнике Паскаля — игра для одного человека с незаурядными арифметическими способностями. Поле для игры выглядит следующим образом:

```
0-я строка:                1
1-я строка:              1   1
2-я строка:            1   2   1
3-я строка:          1   3   3   1
4-я строка:        1   4   6   4   1
5-я строка:      1   5  10  10  5   1
6-я строка:    1   6  15  20  15  6   1
               . . . . .
```

В каждом ряду слева и справа стоят единицы, а всякое внутреннее число получается как сумма двух чисел, стоящих над ним.

В начале игры фишка ставится на верхнюю единицу. За ход игрок передвигает фишку на одну строчку вниз и либо на полстолбца влево, либо на полстолбца вправо. Например, с четверки игрок может походить либо на пятерку, либо на десятку. Сделав этот ход, игрок получает количество очков, равное сумме цифр в том числе, на которое он походил. Для удобства будем считать, что за начальное положение фишки (за самую верхнюю единицу) игрок также получает бесплатное первое очко.

Задача игрока - дойти фишкой до n -й строки и заработать при этом максимальное количество очков. Точнее, это задача для Вашей программы.

Формат входных данных

Во входном файле содержится целое неотрицательное число n — номер строки, до которой надо добраться ($n \leq 30$).

Формат выходных данных

В выходной файл выведите максимальное количество очков, которое можно заработать в этой игре.

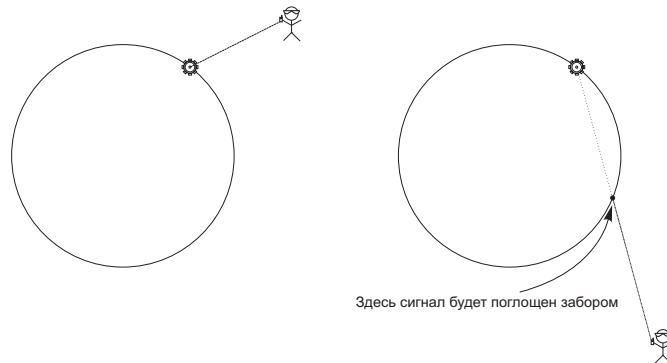
Примеры

ВВОД	ВЫВОД
2	4
6	22

Задача В. Шпион

Ограничение по времени: 2 секунды
Ограничение по памяти: 64 Мб

Секретный объект обнесен забором, выполненным в форме окружности радиуса r с центром в точке $(0, 0)$. Шпион, пытающийся разведать точное местоположение секретного объекта, для преодоления забора установил бомбу точно на заборе в точке (a_x, a_y) . Бомба активизируется дистанционным взрывателем, сигнал от которого распространяется по прямой, причём качество его приёма не зависит от пройденного расстояния. Из секретных источников известно, что забор полностью поглощает проходящий сквозь него сигнал (на сигнал, проходящий по касательной, забор не влияет). Шпион, находясь вне области, окруженной забором, в точке (b_x, b_y) , решил взорвать бомбу. Помогите ему узнать, дойдет ли сигнал до бомбы или будет поглощен.



Формат входных данных

В первой строке входного файла записано число r ($r > 0$). Во второй строке входного файла содержатся два числа (a_x, a_y) . В третьей строке входного файла содержатся два числа (b_x, b_y) . Все числа во входном файле целые и не превышают по модулю 10^3 .

Формат выходных данных

В единственной строке выходного файла выведите слово YES, если сигнал дойдет до бомбы, и NO в противном случае.

Примеры

ВВОД	ВЫВОД
1 1 0 2 0	YES
5 3 4 5 -1	NO
5 -3 -4 0 -10	YES

Задача С. Крестики-нолики

Ограничение по времени: 2 секунды
Ограничение по памяти: 64 Мб

Крестики-нолики на бесконечном поле — это игра для двух человек. Поле представляет собой бесконечный клетчатый лист бумаги. Игроки ходят по очереди. За один ход игрок ставит в какую-нибудь свободную клетку игрового поля свой символ. Символ первого игрока — крестик, а символ второго игрока — нолик.

Если после очередного хода игрока на поле появляются **пять** стоящих в ряд (по горизонтали, вертикали или диагонали) символов этого игрока, то он объявляется победителем партии, и игра заканчивается.

Однажды одаренный школьник Антон обнаружил у себя в тетради некоторую картинку из крестиков и ноликов. Он не может вспомнить, что это такое: поле его поединка с другом Лёшей или просто произвольная картинка из крестов и нулей.

Напишите программу, которая по заданной картине из крестиков и ноликов определяет, могла ли такая позиция возникнуть на поле в результате игры — либо законченной, либо незаконченной. Предполагается, что игроки всегда соблюдают все правила.

Формат входных данных

Во входном файле находится картина из тетради Антона. Пустые клетки обозначаются символом ‘.’ (точка). Символы игроков обозначаются символами ‘X’ и ‘O’ (заглавные латинские буквы ‘икс’ и ‘о’).

Количество строчек во входном файле не превосходит 100. Количество символов в каждой строке также не превосходит 100. Пустых строчек в файле нет. Гарантируется, что во входном файле будет хотя бы один крестик или нолик.

Формат выходных данных

В выходной файл выведите слово **CORRECT**, если данная позиция могла возникнуть в результате игры в крестики-нолики на бесконечном поле (в конце игры или в середине). В противном случае выведите слово **INCORRECT**.

Примеры

ВВОД	ВЫВОД
.X ..X ..OXO ...XO	CORRECT
.... .XO... .XO.. .XO... .XO.. .XO...	INCORRECT

Задача D. Муравьи на кубической Земле

Ограничение по времени: 2 секунды
Ограничение по памяти: 64 Мб

В условиях данной задачи будем пользоваться предположением, что Земля имеет форму куба, и каждая грань — квадрат $m \times m$, расчерченный на клетки размером 1×1 .

В начальный момент времени n муравьев стоят на верхней грани этого куба. Каждый муравей направлен в одну из четырех сторон — на север, на юг, на запад или на восток.

В определенный момент муравьи начинают двигаться по прямой, каждый в своем направлении. Когда муравей доползает до ребра куба, он переползает через него и продолжает движение по следующей грани. При этом он все время движется перпендикулярно тому ребру, которое переполз.

Такое движение продолжается бесконечно долго. Выясните, сколько есть клеток на кубе, на которых ни разу во время этого процесса не побывает ни один муравей.

Формат входных данных

В первой строчке входного файла содержатся два натуральных числа — n и m — количество муравьев на земле и длина стороны планеты ($1 \leq n \leq 100000$; $1 \leq m \leq 15000$).

В каждой из следующих n строчек находится описание начального положения очередного муравья. Сначала идут два натуральных числа x и y — координаты муравья на верхней грани, а затем символ, задающий направление муравья — 'N', 'S', 'W' или 'E'. Числа и символ разделяются ровно одним пробелом.

Оси координат и направления сторон света приведены на рисунке. Все координаты лежат в интервале от 1 до m включительно.

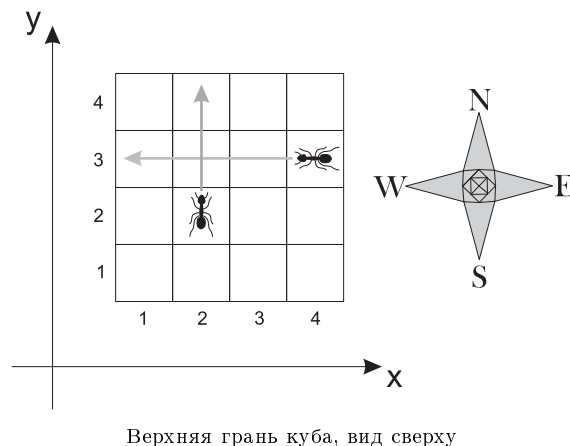
Несколько муравьев как в начальный момент времени, так и в любой другой, могут оказаться на одной клетке. Это никак не влияет на траектории их движения.

Формат выходных данных

В выходной файл выведите одно число — количество клеток, никогда не посещаемых муравьями.

Пример

ВВОД	ВЫВОД
2 4 2 2 N 4 3 W	66



Задача Е. Цифры

Ограничение по времени: 2 секунды
Ограничение по памяти: 64 Мб

Назовем *суммой цифр числовой последовательности* сумму цифр всех ее чисел. Например, для последовательности чисел 14, 22, 239 сумма цифр будет равна $(1 + 4) + (2 + 2) + (2 + 3 + 9) = 23$.

Ваша задача — при данном n найти сумму цифр следующей числовой последовательности:

$$1, 2, 3, \dots, 10^n - 1$$

Формат входных данных

На первой строке входного файла находится целое число n ($1 \leq n \leq 100000$).

Формат выходных данных

Выведите в выходной файл одно число — искомую сумму цифр числовой последовательности.

Пример

ВВОД	ВЫВОД
1	45

Задача F. Эндшпиль

Ограничение по времени: 2 секунды
Ограничение по памяти: 64 Мб

Партия в шахматы подходила к концу, и на поле оставалось всего две фигуры: белый и черный короли. Однако, вопреки шахматным правилам, игроки решили доиграть эту партию до самого конца, руководствуясь девизом “остаться должен только один!”. Но короли, в целях государственной безопасности, были заменены “охранниками”. Охранник — это фигура, которая в каждый свой ход перемещается ровно на одну клетку по горизонтали или вертикали, и бьет эти же клетки. Вам необходимо сказать, чем закончится партия при оптимальной игре обоих соперников.

Формат входных данных

Во входном файле дано через пробел расположение охранников (сначала координаты белого, затем черного), а затем информация о том, кто будет ходить первым (символ ‘B’ — если первым будет ходить черный охранник, и символ ‘W’ в противном случае). Расположение фигур дано в стандартной шахматной записи — сначала указывается символ столбца шахматного поля (строчная латинская буква от a до h), затем указывается номер строки шахматного поля (цифра от 1 до 8).

Формат выходных данных

Необходимо вывести, чем закончится партия при оптимальной игре обоих соперников: **White wins** — если выиграют белые, **Black wins** — если победят черные, и **Draw** — если будет ничья. Партия заканчивается ничьей, если ни один из соперников не может победить. Партия закончится победой одной из сторон, когда охранник “съедает” охранника соперника. Изначально охранники могут занимать абсолютно любую позицию на поле, за исключением того, что не могут находиться на одной клетке.

Пример

ВВОД	ВЫВОД
g6 g7 B	Black wins