Задача A. Сколько лестниц?

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

Назовём *лестницей длины* L *и высоты* H последовательность чисел a_0, a_1, a_2, $\ldots$, a_L такую, что

- $a_0 = 0$, $a_L = H$;
- Для любого $0 < i \leqslant L$ либо $a_i = a_{i-1} + 1$, либо $a_i = a_{i-1} + 2$.

По заданным L и H найдите количество различных лестниц длины L и высоты H. Лестницы считаются различными, если соответствующие им последовательности различаются хотя бы в одном элементе.

Формат входного файла

В первой строке входного файла заданы два числа L и H через пробел ($1 \leqslant L, H \leqslant 15$).

Формат выходного файла

Выведите в выходной файл одно число — количество лестниц длины L и высоты H.

Примеры

stairs.in	stairs.out
1 1	1
2 3	2

Задача B. Обход поля

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

На бесконечном клетчатом поле в клетке с координатами (0, 0) стоит робот, лицевой стороной обращённый к клетке (0, 1). Робот начинает обходить поле по следующему алгоритму:

1. Если на клетке перед лицевой стороной робота он уже бывал, повернуть на 90 градусов влево и вернуться к шагу 1, иначе перейти к шагу 2.

2. Перейти на клетку перед лицевой стороной робота, повернуть на 90 градусов вправо и вернуться к шагу 1.

Таким образом, первые несколько клеток, посещённых роботом, будут иметь координаты (0, 0), (0, 1), (1, 1), (1, 0), (1, −1), ...

Некоторые клетки на поле отмечены. Робот завершает обход, когда оказывается на отмеченной клетке. Выведите количество клеток, посещённых роботом при обходе.

Формат входного файла

В первой строке входного файла задано число N ($1 \leqslant N \leqslant 100$). Следующие N строк содежат по два числа X_i и Y_i, разделённых пробелом — координаты отмеченных клеток. Все координаты не превосходят по модулю 100.

Формат выходного файла

Выведите в выходной файл одно число — количество посещённых роботом клеток, после которого от впервые окажется в одной из отмеченных клеток поля. Если робот никогда не попадёт ни в одну из отмеченных клеток, выведите −1.

Примеры

walk.in	walk.out
2 1 1 -1 -1	3
1 2 2	13
1 0 0	1

Задача C. Простые числа

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

Как известно, *простым числом* называется натуральное число, которое делится нацело только на 1 и на себя. Так, первые пять простых чисел — это 2, 3, 5, 7 и 11; единица простым числом не считается.

Недавно, изучая свойства простых чисел, Петя обнаружил, что если написать в строчку N цифр и читать их справа налево или слева направо, то обычно некоторые подстроки оказываются простыми числами. Так, например, если написать строку 133, то при чтении слева направо в ней встречаются простые числа 13 (один раз) и 3 (два раза), а при чтении справа налево — простые числа 3 (ещё два раза), 331 и 31 — по одному разу.

Теперь Петя хочет выписать строчку, в которой при чтении справа налево и слева направо суммарно встречается в качестве подстрок как можно больше простых чисел. Так, для строки 133 количество подстрок, являющихся простыми числами, с учётом кратности равно 7, но Петя надеется, что существует другая строка из трёх цифр, в которой количество простых чисел будет ещё больше.

Для данного N выведите строку из N цифр, в которой будет содержаться в качестве подстрок как можно больше простых чисел.

Формат входного файла

В первой строке входного файла задано единственное число N ($1 \leqslant N \leqslant 10$).

Формат выходного файла

В первую строку выходного файла выведите N десятичных цифр без пробелов — строку, которая при чтении справа налево и слева направо суммарно содержит в качестве подстрок как можно больше простых чисел с учётом кратности.

Система оценки

Для каждого N от 1 до 10 сначала вычисляется S_N — максимальное количество подстрок, являющихся простыми числами, полученное всеми участниками и жюри для этого N. Если в ответе участника содержится столько же простых чисел, за это начисляется 20 баллов; в противном случае количество баллов для этого N равно $10 \cdot \frac{C_N}{S_N}$, где C_N — количество простых чисел в ответе участника.

Примеры

primes.in	primes.out
2	20
3	133

Указанные ответы не обязательно являются оптимальными. В примере с $N = 2$ количество простых чисел — 2 (двойка, прочитанная слева направо и справа налево); обратите внимание на то, что подстрока 02 не считается корректной записью простого числа, так как содержит ведущие нули.