

1. Заячьи петли (haak)

1 сек

20 очков

Зайчонок Юська учится убегать от хищников, и мама дала ему задание — прыгать по полю, стараясь при этом как можно больше петлять. Так как мама не может сейчас проследить за тренировкой, она попросила Юську записать траекторию своих прыжков на смартфон, чтобы потом их проанализировать.

Каждый раз, когда Юська приземляется, телефон сохраняет координаты (X, Y) соответствующей точки. Будем называть *прыжком* движение Юськи из каждой такой точки в следующую. *Петляющая серия прыжков с запада на восток* — это такая последовательность двух или более прыжков подряд, в которой чередуются прыжки на *северо-восток* и *юго-восток* через один. Первый прыжок в серии может быть в любом из этих двух направлений, но в последующих прыжках направление должно всегда чередоваться. Прыжок считается сделанным в направлении на северо-восток, если в результате как X -, так и Y -координата зайчонка увеличились. При прыжке на юго-восток координата X увеличивается, а координата Y — уменьшается.

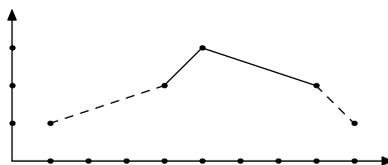
Необходимо написать программу, которая анализирует последовательность точек, записанных на смартфон Юськи, и находит в ней самую длинную *петляющую серию прыжков с запада на восток*.

Входные данные. На первой строке текстового файла `haaksis.txt` дано число сохранённых точек N ($1 \leq N \leq 50\,000$). На каждой из следующих N строк даны два разделённых пробелом целых числа: координаты соответствующей точки X_i и Y_i . Абсолютные значения координат не превышают $1\,000\,000$.

Выходные данные. На единственной строке текстового файла `haakval.txt` вывести количество прыжков в самой длинной петляющей серии прыжков с запада на восток. Если Юська не напрыгал ни одной такой серии, вывели текст `EI OLE`.

Пример.

	<code>haaksis.txt</code>	<code>haakval.txt</code>
	5	2
	1 1	
	4 2	
	5 3	
	8 2	
	9 1	



Пример.

	<code>haaksis.txt</code>	<code>haakval.txt</code>
	4	EI OLE
	2 1	
	6 2	
	4 1	
	8 2	

