

**1. СВЯЗАННЫЙ ГРАФ**

10 очков

5 секунд

Ориентированный граф называется сильно связанным, если для любых двух его вершин  $a$  и  $b$  найдется путь либо из  $a$  в  $b$ , либо из  $b$  в  $a$  (либо оба). Написать программу, которая проверяет, является ли данный ориентированный граф сильно связанным.

Входные данные. В первой строке текстового файла `SIDUS.SIS` находятся число вершин графа  $N$  ( $2 \leq N \leq 100$ ) и число дуг графа  $M$  ( $1 \leq M \leq 1000$ ). Вершины графа пронумерованы  $1..N$ . В каждой из следующих  $M$  строк файла находятся начальная и конечная вершины одной дуги.

Выходные данные. В первую строку текстового файла `SIDUS.VAL` вывести `JAH`, если заданный граф является сильно связанным, или `EI`, если не является, и в этом случае также вывести во вторую строку файла одну такую пару вершин  $a$   $b$ , что не найдется пути ни из  $a$  в  $b$ , ни из  $b$  в  $a$ .

<u>Пример.</u>	<code>SIDUS.SIS</code>	<code>SIDUS.VAL</code>
	3 2	EI
	1 2	2 3
	1 3	

Оценивание. В этом задании за тесты с `JAH`-ответами очки получают только те программы, которые правильно решают все тесты с `EI`-ответами.

**2. ИГРА "ЗМЕЙКА"**

30 очков

5 секунд

На плоскости задано некоторое множество отрезков, параллельных осям координат. Написать программу, которая проверяет, существует ли непрерывная кривая, которая пересекается с каждым отрезком ровно один раз.

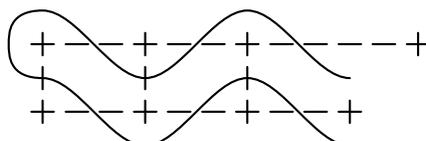
Начальной и конечной точками искомой кривой могут быть любые точки плоскости, кроме точек, лежащих на отрезках. Пересекать каждый отрезок кривая может в любой внутренней точке, то есть в любой точке, кроме концов отрезка. Сами отрезки между собой могут пересекаться только в своих конечных точках.

Входные данные. В текстовом файле `USS.SIS` задано расположение отрезков на плоскости в псевдографическом виде. В первой строке файла находится высота "рисунка" – количество строк  $N$  ( $1 \leq N \leq 50$ ). В каждой из следующих  $N$  строк не более чем 50 символов " ", "|", "-" и "+", где "|" и "-" означают соответственно внутренние, а "+" – конечные точки отрезков. У каждого отрезка есть по крайней мере одна внутренняя точка (то есть на "рисунке" ни в одном ряду и ни в одном столбце нет двух знаков "+" рядом).

Выходные данные. Если требуемая кривая существует, вывести в единственную строку текстового файла `USS.VAL` слово `JAH`, иначе `EI`.

<u>Пример.</u>	<code>USS.SIS</code>	<code>USS.VAL</code>
	3	JAH
	+ - + - + - - - +	
	+ - + - + - - +	

Одна из множества возможных кривых:



Оценивание. В этом задании тесты разделены на группы, и участник получит очки за группу тестов, только если его программа правильно решит все тесты из группы.

**3. ХОККЕЙ**

30 очков

5 секунд

На прямоугольном хоккейном поле находятся движущаяся шайба и игрок. Написать программу, которая находит минимальное время, за которое игрок может добраться до шайбы.

Входные данные. В первой строке текстового файла `НОКИ.SIS` находятся размеры поля  $X$  и  $Y$  ( $0 < X \leq Y \leq 1000$ ), во второй строке – начальные координаты шайбы  $X_1$  и  $Y_1$  ( $0 \leq X_1 \leq X$ ,  $0 \leq Y_1 \leq Y$ ), в третьей строке – проекции скорости шайбы  $v_x$  и  $v_y$  ( $|v_x| \leq 1000$ ,  $|v_y| \leq 1000$ ), в четвертой строке – начальные координаты игрока  $X_m$  и  $Y_m$  ( $0 \leq X_m \leq X$ ,  $0 \leq Y_m \leq Y$ ), и в пятой строке – максимальная скорость игрока  $v$  ( $0 < v \leq 1000$ ). Все данные – действительные числа, заданные в системе СИ (длины и координаты в метрах, скорость в м/с).

Выходные данные. В единственную строку текстового файла `НОКИ.VAL` вывести действительное число  $T$  – минимальное время (в секундах), по истечении которого возможно, что игрок и шайба будут находиться в одной точке. Предполагается, что трение отсутствует, отражения шайбы от бортов поля идеальные, и игрок может достичь своей максимальной скорости мгновенно. Выводимый результат не должен отличаться от точного значения более чем на 0,01%.

<u>Пример.</u>	<code>НОКИ.SIS</code>	<code>НОКИ.VAL</code>
	30 60	3
	15 20	
	0 20	
	15 10	
	10	

**4. ГРАВИТРИПС**

30 очков

2 секунды

Гравитрипс – это настольная игра для двух игроков, в которую играют черными и белыми фишками на прямоугольном поле. Поле состоит из 6×6 ячеек и располагается вертикально (ребром). Ячейки образуют 6 столбцов, в каждом 6 ячеек. В начале игры все ячейки пустые, и игроки начинают размещать свои фишки в ячейках с целью собрать 4 подряд фишки в одном столбце, ряду или диагонали.

За каждый ход игрок бросает одну свою фишку в какой-либо столбец. Фишка падает по столбцу вниз и оказывается в самой нижней свободной клетке этого столбца.

Придумать и реализовать стратегию для этой игры. Программа должна прочитать из входного файла состояние поля и сделать один ход. Исключение составляет начало игры, когда программа должна сообщить судье имя автора. Заданное для задачи временное ограничение распространяется на совершение одного хода.

Входные данные. В первой строке текстового файла GRAVI.SIS находится число 0, 1 или 2. Если 0, программа должна вывести в выходной файл имя своего автора. Если 1 или 2, в каждой из следующих 6 строк файла ровно 6 чисел: состояние игрового поля, где 0 означает пустую клетку, 1 – черную и 2 – белую фишку. Программа должна ходить фишкой того цвета, который указан числом, находящимся в первой строке входного файла.

Выходные данные. Если в первой строке входного файла был 0, вывести в единственную строку текстового файла GRAVI.VAL имя автора программы. Если 1 или 2, вывести в выходной файл ровно 6 строк, в каждой из них ровно 6 чисел – состояние игрового поля после совершения хода.

<u>Пример.</u>	GRAVI.SIS	GRAVI.VAL
	2	000000
	000000	000000
	000000	020000
	000000	112000
	112000	121000
	121000	221000
	221000	

Оценивание. Решения этого задания оцениваются в ходе проведения турнира, в котором каждая программа играет с каждой другой программой две партии: одну черными фишками и одну белыми. Проигрыш дает 0, ничья – 1 и победа – 3 очка. Программа, нарушившая временное ограничение или сделавшая ход не по правилам, сразу считается проигравшей партию. Каждый участник получает количество очков, пропорциональное сумме очков, набранной его программой в турнире.

Замечание. На дискете в дополнение к примерам входных и выходных файлов дан также графический интерфейс для тестирования своего решения (позволяет играть против своего решения) и примеры программы, которые иллюстрируют общение с судьей (советуем использовать пример программы в качестве основы для своей программы; свою программу тем не менее следует представить в файле со своим именем).