



- 1 означает, что Ёжик позвал Лошадь, и она действительно оказалась в той же клетке, что и он. В этом случае другие два числа равны 0, и программа-решение должна закончить свою работу.

Программа-решение не должна делать более 10 000 ходов.

### Пример взаимодействия

| стандартный ввод | стандартный вывод |
|------------------|-------------------|
| 2 3              |                   |
| 1 2              | 0 0 1             |
| 0 1 0            | 1 -1 0            |
| 0 0 0            | 1 0 1             |
| 1 0 0            |                   |

### Комментарий

Ёжик находился в клетке (1, 2). Сначала он попробовал позвать Лошадь в той же клетке (вывод: 0 0 1), но Лошади там не оказалось, и она сместилась вправо (ввод: 0 1 0). Ёжик сместился по диагонали, но Лошадь звать не стал (вывод: 1 -1 0), а Лошадь осталась на месте (ввод: 0 0 0). Ёжик сместился вправо и позвал Лошадь (вывод: 1 0 1). Лошадь оказалась в той же клетке и отозвалась (ввод: 1 0 0). Значит, изначально Лошадь находилась в клетке (2, 1), а встретились они в клетке (3, 1). Ёжик при этом сделал три хода и дважды запросил местоположение Лошади.

### Подзадачи и система оценки

В данной задаче две подзадачи. Каждый тест в обеих подзадачах оценивается отдельно. Оценка за тест вычисляется по формуле  $\min\{10, \text{round}(10 \times (J/S)^2)\}$ , где 10 – оценка в баллах за тест,  $S$  – количество ходов, которое потребовалось программе-решению, чтобы обнаружить Лошадь,  $J$  – количество ходов, которое требуется заданному эталонному решению при том же начальном положении Ёжика. Округление ведется по правилам математики.

#### Подзадача 1 (оценивается из 40 баллов)

$$2 \leq N, M \leq 10.$$

#### Подзадача 2 (оценивается из 60 баллов)

$2 \leq N, M \leq 30$ . В этой подзадаче количество запросов о том, есть ли Лошадь в текущей клетке, не должно превышать  $N \times M$ .

### Вспомогательная программа

Для тестирования своего решения вы можете использовать вспомогательную программу «runpair», которая находится у вас в каталоге «с:\work\runpair».

Она позволяет запустить две программы и перенаправить стандартный поток вывода первой программы на стандартный поток ввода второй программы и наоборот.

Для тестирования с ее помощью программы-решения вам придется помимо программы-решения написать программу, моделирующую поведение лошади. Тогда можно запустить одновременно программу, моделирующую поведение лошади, и программу-решение, с помощью команды

```
runpair «исполнимый файл лошади» «исполнимый файл ёжика»
```

и на экране будет отображен их диалог.