1. Больше разнообразных примеров тут: <http://bit.ly/tetris-tests>

## Ограничения

Ширина и высота стакана не превышают 1 000 000. Количество ячеек в фигуре не превышает 20. Количество фигур не превышает 1 000 000. Количество команд не превышает 100 000 000. Естественно, тесты будут разумными, например, мы не будем заставлять печатать состояние поля размером миллион на миллион.

3 0

.\*.

.\*.

#..

#..

9 1

\*\*.

...

...

#..

11 -9

.\*.

.\*.

...

...

## Оформление решения

1. Всё решение должно содержаться в одном cs-файле с кодом на C#, имя которого составлено по шаблону:**<фамилия>\_<имя>.cs** где **фамилия** и **имя** — твои фамилия и имя латиницей.
2. Решение не должно делать потенциально вредоносных действий — обращаться к сети, к файловой системе (естественно, кроме чтения указанного в аргументах входного файла), запускать другие приложения, и т.п.
3. Не должно использовать сторонние библиотеки, кроме указанных в задании выше.
4. Должно в точности соблюдать формат ввода и вывода. Проверь точное совпадение вывода твоего решения с эталонным выводом на предоставленных примерах.

# Задача 2. Описание решения

В файле **<фамилия>\_<имя>.txt** опиши текстом, как устроено решение, на каких исходных данных оно работает особенно эффективно, а на каких ведет себя наихудшим образом. Мы ожидаем 100-150 слов.

# Полезные ссылки

Не уверен, что стоит считать плохим кодом, а что нет? Мы сделали небольшую игру-тренажер, которая научит базовым навыкам приведения кода в порядок. [**http://cleancodegame.github.io**](http://bit.ly/1C9zofM)

А лучше прочитать книгу ["Чистый код" Роберта Мартина](http://bit.ly/1Dwre5P) или хотя бы пробежаться по её [краткому конспекту](http://slidesha.re/1DTGjwg).

Короткий онлайн практикум по языку запросов LINQ: <http://bit.ly/1CL8XT8>

Концепция неизменяемых классов неплохо описана в серии статей тут: <http://weblogs.asp.net/bleroy/immutability-in-c>

Библиотека неизменяемых коллекций: [https://www.nuget.org/packages/System.Collections.Immutable](https://www.nuget.org/packages/System.Collections.Immutable/)

Библиотека для чтения формата JSON: <https://www.nuget.org/packages/Newtonsoft.Json/>

# Вопросы по формулировке

Вопросы по формулировке можно задавать комментариями в этом документе или письмом на почту [kontur-student@kontur.ru](mailto:kontur-student@kontur.ru)