

## Подробное описание правил игры «Squares 2014»

Если ты читаешь этот документ, то, видимо, у тебя появилась неоправданная уверенность в том, что тебе по силам хотя бы осилить правила, описанные в этом документе. Что ж, посмотрим, оправдаешь ли ты свои собственные ожидания.

Правила этой игры достаточно сложные. Это я к тому, что я пойму тебя, если после их прочтения ты разноешься и пойдешь делать унылые задания по продукции мелккомышек.

Игра получила название «Squares 2014». Чуть ниже ты прочитаешь правила этой игры. Сразу скажу, что твоей задачей будет написать консольное приложение, которое будет выполнять роль игрока. Если после этих слов ты еще не испугался, то, наверное, пора читать сами правила игры.

В игре будут использоваться квадраты пяти разных цветов: красный, зеленый, желтый, синий и оранжевый. Изначально у каждого из игроков есть поле, состоящее из 10 пустых стопок для квадратов. У каждого из игроков, помимо этого, изначально имеется 0 очков.

Далее каждый из игроков заполняет стопки своего соперника квадратами выбранных им цветов так, чтобы высота каждой из стопок составляла 15 квадратов. Затем игроки по очереди совершают ходы. За один ход любой из игроков может сделать ровно одно из следующих действий:

- Произвольным образом перемешать все цвета в *одной* стопке своего поля (в таком случае количество квадратов каждого цвета должно остаться неизменным);
- Произвольным образом перемешать все цвета на *одном* уровне (на одном и том же расстоянии от низа стопки) во всех стопках своего поля (в таком случае количество квадратов каждого цвета так же должно остаться неизменным);
- Выбрать некоторый квадрат, после чего все квадраты одного цвета, которые находятся с ним в одной компоненте четырехсвязности (в том числе и он сам) исчезают. В таком случае игроку дается  $K^2$  очков, где  $K$  — количество исчезнувших квадратов. Если под каким-либо квадратом оказывается свободное место, то он тут же занимает это место. Таким образом, любой квадрат, который не является самым нижним в своей стопке, всегда окажется стоящим на другом квадрате.

Также в свой ход игроку необходимо дополнить (в случае необходимости) все стопки соперника до 15 квадратов. Квадратами какого цвета

дополнять каждую из стопок остается на усмотрение заполняющего игрока.

Игра длится фиксированное четное число ходов  $T$ . Игрок, который на момент окончания игры набрал больше очков, считается победителем.

Компонентой четырехсвязности называется множество квадратов, в котором каждый из квадратов имеет общую сторону хотя бы с одним другим квадратом этого множества (в таком случае квадраты называются соседними), причем от любого квадрата этого множества можно за некоторое количество переходов к соседнему квадрату этого же множества достичь любого другого квадрата этого множества. Пардон муа за тавтологию, я — не Паустовский. Как смог, так и объяснил.

Не трудно догадаться, что пока в этом документе не появятся картинки, все это так и будет оставаться для тебя египетской тьмой. Чтобы не перегрузить твое сознание слишком большим количеством разноцветной информации для примера будет использоваться игровое поле, имеющее 5 стопок, по 3 квадрата каждая.

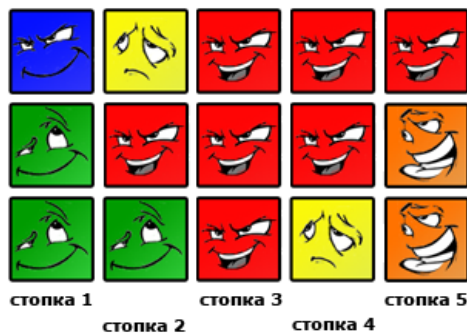


Рис. 1: Пример уменьшенной копии игрового поля

На изображении можно увидеть компоненту четырехсвязности красного цвета размера 7, зеленого цвета — размера 3, оранжевого — размера 2 и синего цвета — размера 1. Еще на этом изображении можно увидеть две компоненты четырехсвязности желтого цвета размера 1.

## Приложение, которое требуется реализовать

После того, как ты прочитал много букв и посмотрел много картинок, тебе, как мне кажется, хочется узнать, что же от тебя требуется из-за всего этого безобразия. А требуется от тебя, отец русского кодотворения, написать консольное приложение, которое бы выполняло роль игрока в описанную выше игру. Далее будет описано все, что тебе необходимо знать для написания этого самого приложения.

Программа должна считывать из файла `input.txt` сначала 15 строк по 10 символов, которые задают собственное поле. Затем программа должна считывать из файла `input.txt` 15 строк по 10 символов, которые задают поле соперника. Каждая из строк, задает один уровень высоты всех стопок слева направо.

Символ `«.»` соответствует пустому месту (такое возможно на поле соперника, если он последним ходом удалил некоторые квадраты), символ `«R»` — красному квадрату, символ `«G»` соответствует зеленому квадрату, символ `«B»` — синему квадрату, символ `«Y»` — желтому, а символ `«O»` — оранжевому. Клетки нумеруются по строкам сверху вниз и по столбцам слева направо.

Программа должна выводить данные в файл `output.txt`. Первые 15 строк по 10 символов должны содержать описание заполненного (дополненного) квадратами поля соперника в формате, аналогичном входному. Если твой невероятный бот планирует совершить ход третьего типа (удаление компоненты четырехсвязности), то требуется далее вывести два целых числа — номер строки и столбца любой клетки выбранной компоненты четырехсвязности. Если же бот настолько разумен и велик, что попытается совершить ход любого другого типа, то необходимо вывести описание своего поля после совершенного хода. Формат вывода так же аналогичен предыдущим — 15 строк по 10 символов. Поле должно отличаться от твоего исходного поля одной строкой или одним столбцом.

Если программа совершает некорректный ход, то считается, что ход пропущен.

На каждый ход программе отводится две секунды реального времени и 128 мегабайт оперативной памяти.

## Как, после прочитанного, жить дальше?

Техническая часть задачи осталась позади. Можно перейти к описанию процессов разработки и тестирования. Всем участникам (через некоторое время) будет предоставлена программа, которая позволяет «сравливать» два написанных бота. Программа будет моделировать игру в течение  $T = 30$  ходов каждого из игроков — в сумме оба игрока совершат  $T \times 2 = 60$  ходов.

Для того, чтобы игроки имели представление о степени «интеллектуальности» своего решения, будут проведены два промежуточных турнира. В первом промежуточном турнире будет использоваться значение  $T = 50$ , во втором —  $T = 65$ .

Финальный турнир будет проведен для значения  $T = 100$ .

За каждую игру в турнире в случае победы команде начисляется 3 турнирных очка, в случае ничейного исхода — 1 турнирное очко, в случае поражения — 0 турнирных очков. В итоговой таблице выше будет команда, которая набрала больше всех турнирных очков. При равенстве количества турнирных очков, выше будет команда, у которой суммарная разница очков (не турнирных) по всем играм максимальна. Если и эти показатели равны, то считается, что команды разделили одно и то же место.

О датах проведения всех турниров будет сказано в отдельном документе. Первый из них, ориентировочно, состоится после конференц-недели.

## Что такое «хорошо», а что такое «плохо»?

Угощаю парой слов о том, что запрещается делать командам и их игровым приложениям:

- Обсуждать идеи своих решений с участниками из других команд;
- Делиться исходными кодами или скомпилированными приложениями любых (даже не окончательных) своих решений с другими командами;
- Читать данные из файлов кроме `input.txt`;
- Выводить данные в файлы кроме `output.txt`;

Стоит сделать акцент на втором пункте. Не делитесь своими решениями ни в каком виде, имейте чувство ответственности перед командой и не позволяйте кому-то с вашей же помощью оказаться выше вас в итоговой таблице.

Сделаю также, акцент на том, что решения, которые будут отправлены на проверку позже указанного в соответствующем документе дедлайна не принимаются.

Удачи в придумывании и написании решений!