

Цифровая олимпиада «Волга-IT'XXI»

simtech group

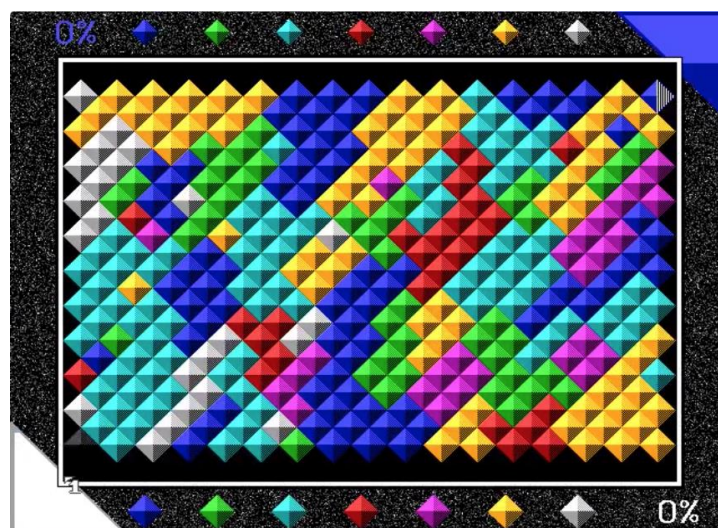
Дисциплина «Веб-разработка (PHP)»

Задание финального этапа

Игра «Филлер»

«Филлер» — игра для двух игроков.

Игра проходит на прямоугольном поле, состоящем из ромбических клеток разных цветов. В начале игры клетки раскрашены случайным образом (как правило, объединены в кластеры по несколько). Каждый игрок начинает игру со своей стартовой клетки, находящейся в левом нижнем или правом верхнем углу поля.



На каждом ходу игрок изменяет цвет своей стартовой клетки на любой другой — при этом все клетки, примыкающие к стартовой области по стороне и окрашенные в тот же цвет, также перекрашиваются в выбранный цвет. Таким образом игрок «захватывает» соседние клетки, перекрашивая свою область в цвет этих клеток.

Игрок не может выбрать на новом ходу свой текущий цвет и цвет, которым на этом ходу окрашена область противника.

Цель игры состоит в захвате большего количества клеток, чем противник.

Геймплей игры: <https://www.youtube.com/watch?v=88ghb75VSrs>

Участнику олимпиады в рамках финального этапа предлагается реализовать клиент для сервера API, который проходит игру «Филлер».

Клиент представляет из себя искусственный интеллект (ИИ) одного из игроков в заранее созданной игре.

Клиент — это отдельный скрипт, который запускается следующей командой:

```
$ php bin/filler_ai.php --gameServer=... --gameId=... --playerId=...
```

- `gameServer` — адрес сервера, на котором ведется игра.

Пример: `https://example.com`

- `gameId` — идентификатор игры.

Пример: `d290f1ee-6c54-4b01-90e6-d701748f0851`

- `playerId` — номер игрока, за которого играет ИИ.

Пример: `2`.

На каждый запуск скрипта ИИ должен сделать один ход в игре, то есть выполнить следующие действия:

- Сделать запрос к API для получения текущего состояние игрового поля.
- Принять решение, нужно ли вообще делать ход.
- Если ход не нужно делать — закончить свою работу с кодом выхода, равным номеру игрока-победителя:

```
exit(1); // Player 1 is the winner
```

ИЛИ

```
exit(2); // Player 2 is the winner
```

- Если ход нужно делать — решить, какой ход сделать следующим.
- Сделать запрос к API на совершение хода.
- Закончить свою работу с кодом выхода 0:

```
exit(0); // The game goes on
```

Требования к реализации

1. Решение должно использовать фреймворк Laravel или библиотеку symfony/console;
2. Скрипт для запуска ИИ должен лежать в файле bin/filler_ai.php в папке проекта;
3. Ход ИИ не должен занимать больше 2 секунд (без учета сетевых запросов);
4. Решение должно быть опубликовано в публичном репозитории на GitHub;
5. В корне репозитория должен лежать файл README.md, в котором описано, как проект запустить локально (установка зависимостей, подготовка окружения и т.д.);
6. Код покрыт тестами (дополнительные баллы).