

Тестовое задание на должность Unity программиста

Предлагаем выполнить тестовое задание на должность Unity программиста. Результат выполнения должен быть представлен в виде репозитория с Unity 2021.1+ проектом на GitHub. Мы не ставим временных сроков для выполнения тестового задания, но просим указать, сколько времени у вас ушло на его решение.

Отправьте, пожалуйста, на Email a.khavanskaya@kefirgames.com или в [Телеграм](#):

- фамилию и имя
- ссылку на резюме
- ссылку на репозиторий GitHub с решением
- сколько времени у вас ушло на решение

Задание:

Необходимо разработать 2D клон оригинальной игры [Asteroids](#).

Цель игры – получить как можно больше очков, расстреливая астероиды и летающие тарелки, избегая при этом столкновения с ними.

Игрок управляет космическим кораблём, который может крутиться влево и вправо, двигаться только вперед и стрелять. Движение корабля должно быть с ускорением и инерцией. Экран не ограничивает передвижения, а является порталом, т.е. если вы упираетесь в верхнюю границу, то появляетесь с нижней.

У корабля есть два вида оружия:

- пули при попадании в астероид разбивают его на обломки меньшего размера, обладающие большей скоростью; попадание пуль в обломки или летающую тарелку приводит к их уничтожению;
- лазер уничтожает все объекты, которые пересекает. Игрок имеет ограниченное количество выстрелов лазером. Выстрелы пополняются со временем.

При столкновении космического корабля с астероидом, обломком или летающей тарелкой выводится сообщение о проигрыше со счетом и приглашением начать игру заново.

После старта игры периодически появляются астероиды и летающие тарелки. Астероиды двигаются в случайном направлении, а летающие тарелки преследуют игрока. Астероиды и летающие тарелки между собой не сталкиваются.

Необходимо добавить UI, на котором будут отображаться показатели корабля:

- координаты
- угол поворота
- мгновенная скорость
- число зарядов лазера
- время отката лазера

Требования к реализации:

- язык программирования: C#
- разделить логику игры и представление. Классы с логикой не должны наследоваться от MonoBehaviour
- использовать preview или experimental версии Unity пакетов нельзя
- физику Unity для передвижения объектов использовать нельзя
- необходимо использовать [Assembly Definitions](#)
- для управления использовать [Input System](#)
- ассеты можно использовать любые, качество их не учитывается
- не использовать сторонние фреймворки

При оценке тестового задания будут учитываться знания объектно-ориентированного программирования, уместное использование паттернов проектирования и стиль написания программы.