

1) Реализуйте простой экран торговца, используя Unity UI. Экран состоит из 3х частей: слева сетка предметов игрока, справа сетка предметов торговца, посередине отображено количество золота игрока.

Требования:

- предметы имеют заданную стоимость
- предметы можно покупать и продавать Drag'n'Drop
- виджет количества золота должен быть обособлен
- количество золота изменяется при покупке и продаже
- чтобы купить предмет необходимо иметь достаточно золота
- купленные предметы появляются у игрока
- проданные предметы появляются у торговца
- купленные предметы продаются дешевле цены покупки

Критерии оценки:

- способ хранения данных
- архитектура экрана
- способ взаимодействия между виджетом золота и экраном
- следование принципу SOLID
- читаемость и поддерживаемость кода

2) Необходимо реализовать сервис для поиска пути через заданные ребра пересечений прямоугольников с наименьшим количеством поворотов, который имеет следующий интерфейс:

```
struct Rectangle
```

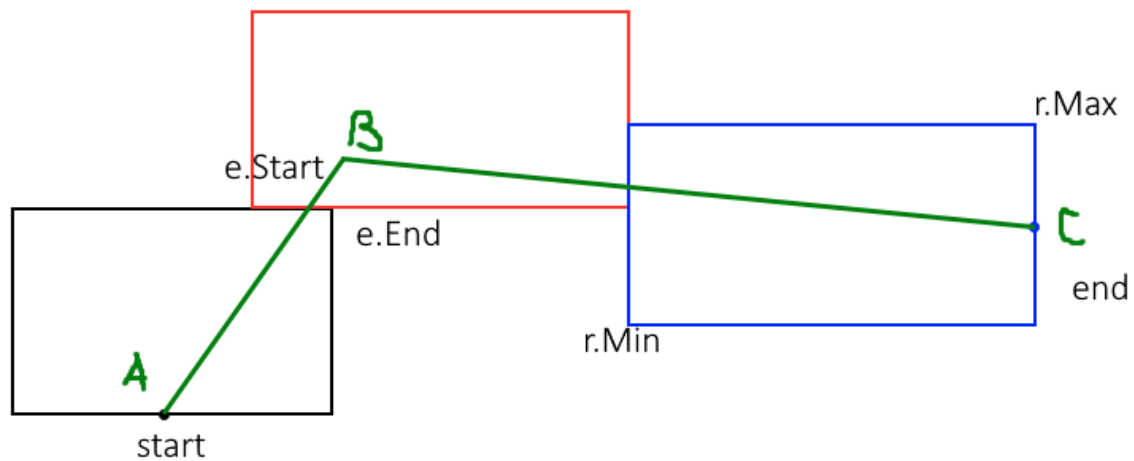
```
{  
    public Vector2 Min;  
    public Vector2 Max;  
}
```

```
struct Edge
```

```
{  
    public Rectangle First;  
    public Rectangle Second;  
    public Vector3 Start;  
    public Vector3 End;  
}
```

```
interface IPathFinder
```

```
{  
    IEnumerable<Vector2> GetPath(Vector2 A, Vector2 C, IEnumerable<Edge> edges);  
}
```



ABC = Path

Требования:

- класс реализации должен принимать входные данные и выдавать результат
- необходимо предусмотреть ошибочные входные данные
- в случае невозможности найти результат возвращать пустой лист

Критерии оценки:

- производительность
- точность найденного пути

Все задания необходимо реализовать в Unity (2019 LTS или 2020 LTS) одним проектом в 2х сценах.