



## Тест<sup>1</sup> для соискателей вакансии Программист игровой логики

Уважаемый соискатель, перед Вами тестовое задание на вакансию «Программист игровой логики» в компании Saber Interactive.

После выполнения задания в сообщении укажите, пожалуйста:

*Фамилию, имя, отчество*

*Дату выполнения*

*Примерное количество времени, затраченного на выполнение теста*

**Само решение пришлите 2 файлами: один с кодом, второй со схемой.**

По срокам выполнения задания, мы не ограничиваем. Напишите сколько вам потребуется времени для его выполнения и когда ожидать решение, после ознакомления с заданием.

### **Техническое задание:**

1. Реализуйте функции сериализации и десериализации двусвязного списка, заданного следующим образом:

```
class ListNode
{
    public ListNode Prev;
    public ListNode Next;
    public ListNode Rand; // произвольный элемент внутри списка
    public string Data;
}

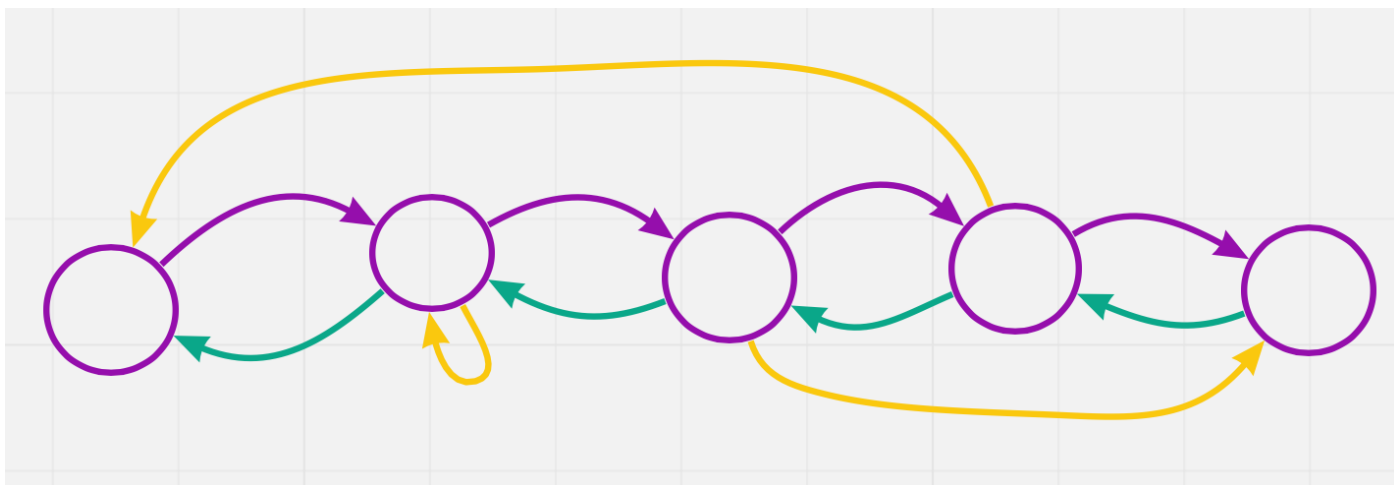
class ListRand
{
    public ListNode Head;
    public ListNode Tail;
    public int Count;

    public void Serialize(FileStream s)
    {
    }

    public void Deserialize(FileStream s)
    {
    }
}
```

<sup>1</sup> Данное задание разработано специально для тестирования кандидатов и не имеет отношения к какому-либо конкретному проекту, поэтому мы гарантируем, что ваш **концепт арт, рисунок, код, модель и т.д.**, выполненный в качестве теста не будет использоваться в коммерческой разработке. Конечную работу вы можете использовать в своем портфолио и размещать на любых ресурсах.

**Примечание:** сериализация подразумевает сохранение и восстановление полной структуры списка, включая взаимное соотношение его элементов между собой — в том числе ссылок на Rand элементы.



пример ListRand, желтые стрелки — ссылки на rand элементы

**Алгоритмическая сложность решения должна быть меньше квадратичной.**

**Нельзя добавлять новые поля в исходные классы ListNode, ListRand**

Для выполнения задания можно использовать любой общеиспользуемый язык.

Тест нужно выполнить без использования библиотек/стандартных средств сериализации.

**2. Напишите ИИ для противника используя BvTree (достаточно нарисовать схему, реализация в каком-либо из движков не требуется).**

Напишите ИИ для противника используя BvTree (достаточно нарисовать схему, реализация в каком-либо из движков не требуется).

Солдат - сущность, которая может стрелять, отправиться в указанную точку и ждать.

Солдат проводит патруль по зацикленному маршруту по точкам А и Б

По прибытии на точку солдат останавливается на 5 минут в ожидании врага. Если за 5 минут враг не появился, солдат идет на следующую точку.

Если враг обнаружен, солдат стреляет в него с паузой в 1 секунду между каждым выстрелом