**Богословский Никита. Тест-опрос.**

**1.Слушали ли вы курсы по ИИ (не обязательно в контексте игр)?**

Да. В рамках учебной программы мехмата Интеллектуальные системы (4 курс бакалавриата) и Машинное обучение (1 курс магистратуры). Самостоятельно изучал Стэнфордский курс по сверточным нейронным сетям и компьютерному зрению (cs231n), а также курс от Яндекса по глубокому обучению – Practical DL. Самостоятельное обучение было больше сфокусировано на глубоком обучении и компьютерном зрении, также частично на NLP (рекуррентные сети, трансформеры, обработка речи).

**2.Что такое FSM (конечный автомат), реализовывали ли вы его когда-либо?**

Грубо говоря, это четверка S, S0, F, t, где S – это множество состояний, S0 – стартовое состояние, F – множество конечных состояний, t – функция переходов между состояниями. Автомат детерминированный, если из одного состояния существует не больше одного перехода, и недетерминированный, если функция переходов допускает несколько переходов из одного состояния.

Реализовывал, в рамках учебной программы и работы.

**3.Опишите простейший FSM для какой-либо игры.**

Пусть у нас есть поле размера n x m, в одной клетке стоит игрок, и ему нужно дойти до какой-то другой искомой клетки. Тогда множество состояний S будет включать в себя все возможные позиции на поле, стартовым состоянием S0 будет клетка, где стоит игрок, множество конечных состояний F будет состоять из одного состояния, которое соответствует искомой клетке, а функция переходов t будет задавать возможные переходы для каждой позиции (если мы стоим у стенки, например, то не все ходы нам доступны, в стену ходить нельзя). Автомат, к слову, получится недетерминированным.

**4.Что такое эвристика, есть ли к ней какие-либо требования?**

В целом, эвристика – это некое правило, в соответствии с которым алгоритм принимает решения в различных задачах. В рамках алгоритма поиска пути А\*, эвристика h – это оценка пути из текущей вершины в конечную. Для эвристики из А\* есть два требования – допустимость и преемственность. Допустимость означает, что функция h должна для любой вершины возвращать оценку, меньшую или равную кратчайшему пути до конечной вершины (то есть оценка должна быть оптимистичной). Преемственность означает, что разница значений функции h для вершины и ее потомка не должна превышать весового значения ребра между этими вершинами, также значение функции h для конечной вершины должно быть нулевое.

**5.Общая идея алгоритма A\*, IDA\*, чем отличаются?**

Алгоритм А\* позволяет найти кратчайший путь на графе из стартовой вершины в конечную, оценивая каждую вершину как сумму уже пройденного пути g и оценки оставшегося в соответствии с эвристикой h, то есть f = g + h. А\* похож на алгоритм Дейкстры и использует очередь с приоритетом, но ищет путь только до одной конкретной вершины, а не до всех, и учитывает для каждой вершины не только пройденный путь, но еще и оставшийся с помощью эвристики.

IDA\* - это A\* с итеративным углублением. Грубо говоря, есть алгоритм поиска в глубину. Его можно сделать итеративным следующим образом: мы ставим некоторый порог по глубине, производим поиск только до этого порога, и если мы не смогли найти нужную нам вершину, то производим повторный поиск в глубину с уже большим порогом – так мы повышаем порог и итеративно ищем, пока не найдем искомую вершину. IDA\* работает похожим образом, только вместо глубины он использует значение f из A\* (f = g + h), и при повышении порога он берет наименьшее значение f, которое превышало порог на предыдущей итерации.

IDA\* отличается меньшими затратами по памяти в отличие от А\*.

**6.Работали ли вы когда-нибудь в Unity3D или Unreal Engine?**

Немного в Unity3D в рамках проектной деятельности магистратуры.

**7.Реализовывали ли какие-либо игры с элементами ИИ (любые, от логических до ААА-проектов)?**

Только в рамках учебной программы курса Интеллектуальные системы (пятнашки с помощью A\* и IDА\*, инфекция с помощью альфа/бета отсечения).

**8.Ваши любимые игры, в которых вы достаточно хорошо разбираетесь (2-3 проекта).**

*Rogue Adventure: Roguelike RPG* – карточный рогалик, где основная механика – построение колоды.

*Vector* – мобильный 2D-платформер с элементами паркура.

А вообще я больше стримы смотрю. На них мне понравились *Cyberpunk 2077* и *Atomic Heart*. Возможно, если бы было больше времени, поиграл бы в них сам.