

## СЛОЖНОСТИ ОРГАНИЗАЦИИ ОБУЧЕНИЯ В СФЕРЕ ИНФОРМАЦИОННЫХ ТЕХНОЛОГИЙ В УСЛОВИЯХ ДОСТУПНОСТИ НЕЙРОСЕТЕЙ И ВОЗМОЖНЫЕ ПУТИ РЕШЕНИЯ

**Наседкин Максим Евгеньевич**

[nasedkin.me511035@students.tou.edu.kz](mailto:nasedkin.me511035@students.tou.edu.kz)

Студент 2 курса образовательной программы «Computer Science»

Торайгыров университет, г.Павлодар, Республика Казахстан

Научный руководитель, ст. преподаватель – **Садыкова А.О.**

Согласно статистике, на данный момент генеративным искусственным интеллектом пользуется свыше 400 миллионов человек в неделю [1]. По статистике, приведенной компанией «Антиплагиат-Казахстан», в 2024 году из 95 тысяч академических работ студентов 8 тысяч имели следы искусственной генерации текста [2]. То, что студенты пользуются нейросетями в обучении – это хорошо, генеративный искусственный интеллект помогает ускорить обучение, если используется в качестве личного преподавателя, который укажет на ошибки, поможет найти более эффективное решение, или объяснит непонятную тему. Однако, когда студенты используют ИИ, чтобы он решал за них задания, создавал готовые презентации и академические работы, то это превращается в большую проблему.

Суть образования в том, чтобы дать человеку знания и навыки, которые пригодятся в работе. Студент изучает темы в течение курса, выполняет практические задания, накапливает знания и опыт, которые будет использовать в будущем. На факультетах, связанных с разработкой приложений, сайтов и написанием кода, важно понимание алгоритмов и конструкций, необходимых для создания и защиты создаваемых продуктов. Получить эти знания можно только путем проверки на практике, чем и должны заниматься обучающиеся. До появления генеративного ИИ в открытом доступе, студентам приходилось искать чужой код, пробовать брать из него фрагменты и запускать у себя, меняя что-либо. Так появлялось понимание того, как устроен алгоритм, за что отвечают функции. Вместе с тем, как генеративный ИИ стал доступен каждому, кто-то стал пропускать этот метод проб и ошибок, который и приносит основной багаж полезных знаний. Ведь можно просто попросить нейросеть решить задачу, и код с большой долей вероятности запустится, не выдав ошибку. А если код все же не запустился, то не составит труда попросить ИИ-ассистента исправить ошибку. Таким образом получается программист, который совершенно не понимает, что происходит в его коде. Согласно исследованию, которое провел центр безопасности и новых технологий при Джорджтаунском университете, опубликованному в ноябре 2024 года, примерно 48% кода от генеративного ИИ содержит уязвимости [3]. Поэтому важно заботиться о качестве обучения будущих разработчиков.

На начальных этапах обучения, когда студента знакомят с профессией, обучают алгоритмам, дают начальные знания по созданию кода, можно воспользоваться

инструментами, к которым нейросеть не сможет приложить руку. Одним из таких инструментов можно считать бесплатную платформу Scratch.

Scratch - это визуально-блочная среда программирования, в которой как такового кода нет. Приложение, а чаще всего игра или анимация, создается с помощью блоков. Есть разные блоки: отслеживающие выполнение определенных действий пользователем, логические операторы, математические действия и многие другие. Главная особенность, которая нас интересует – тут нет встроенных нейросетей, которые сами соединят нам блоки, а также нельзя вставить обычный код. Студенту придется самому создавать логику разрабатываемого проекта. У этой платформы есть разные альтернативы, например «Snap!» - расширенная модификация Scratch с данными классов, процедурами, рекурсией и возможностью создавать собственные блоки. Однако, в статье рассматриваются именно возможности создания студенческих проектов на оригинальном сайте Scratch.



Рисунок 1 – Платформа Scratch

Чтобы развивать логическое мышление и научиться создавать алгоритмы, можно в качестве практического задания дать обучающемуся выбрать проект, который он реализует на данной платформе. Это может быть уже существующая небольшая 2D игра, собственная разработка студента, или задание, которое придумает сам преподаватель. Главные пункты разрабатываемого проекта – это уникальность на данной платформе, сам проект не должен быть маленьким и простым, а также алгоритмы должны быть придуманы, учитывая особенности этой платформы.

В качестве примера такого проекта возьмем игру «Neighbours from Hell», созданную в 2003 году студией «JoWood Entertainment» (рисунок 2). Было решено перенести частичный функционал одного уровня, главные элементы меню, нужные для его запуска и часть игровых механик, необходимых для игры и победы. Этот проект создан в рамках курса «Введение в специальность», когда нужно было создать проект на платформе Scratch. Вся логика кода и вычислений была придумана самостоятельно в процессе разработки.



Рисунок 2 – Игра, выбранная для проекта

Разберем основные моменты, которые были использованы при создании этого проекта. Поскольку в Scratch есть некоторые ограничения, например в разрешении экрана, а также в перемещении объектов за его пределы, то нужно адаптироваться под них. Из-за его нестандартного соотношения сторон, а также отсутствия инструмента камеры, нужно придумывать логику перемещения объектов на экране. Для решения проблемы с ограничением перемещения объектов за пределами экрана было придумано простое решение – разделить огромный фон игрового уровня пополам. Для реализации переключения между разными разделами меню и запуска уровня была использована несложная система (рисунок 3).

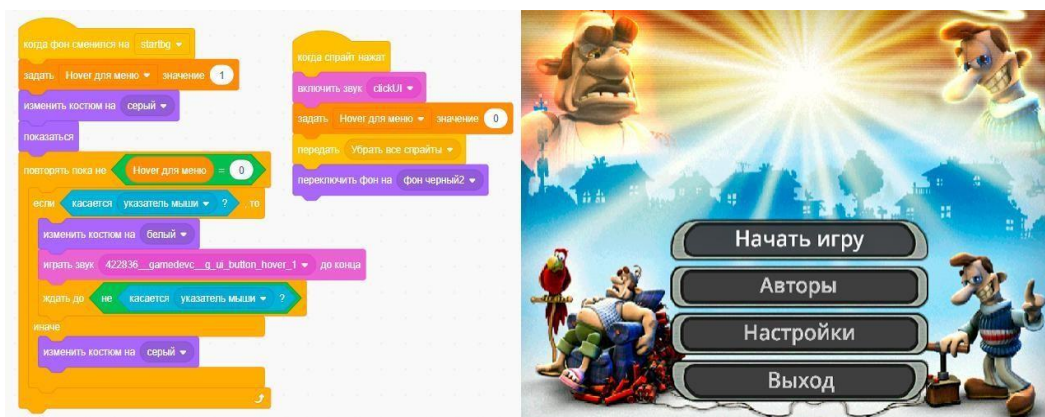


Рисунок 3 – Реализация переключения меню и уровней.

На рисунке 3 видно два блока кода кнопки «Начать игру». Левый блок кода ожидает, когда фон сменится на «startbg», после чего меняет вид кнопки на серый, включает отслеживание наведения на кнопку, чтобы при наведении на нее курсора срабатывал звук, а также текст окрашивался в более яркий цвет. Правый же блок кода при нажатии на кнопку скроет все элементы с экрана, и переключит фон на другой, после чего другие элементы поймают событие переключения и появятся на экране, так появится меню выбора уровня. По такому же алгоритму при запуске уровня скроются все элементы с экрана и фон сменится на первый уровень, после чего появятся персонажи, интерфейс, все объекты и начнется сам уровень. В оригинальной игре можно передвигать камеру, если перемещать курсор к краям экрана. Для реализации этого была создана логика расчета координат, которая в дальнейшем будет использоваться для всех игровых объектов на уровне (рисунок 4).

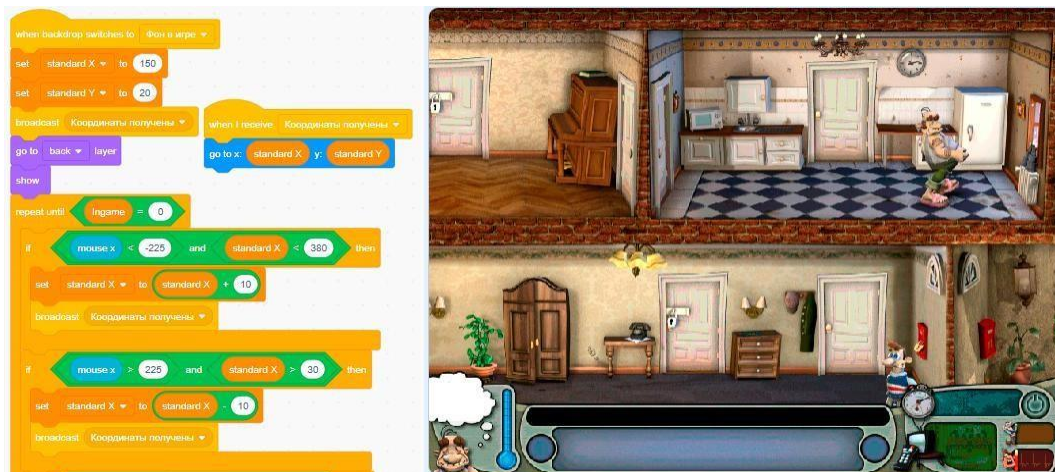


Рисунок 4 – Логика передвижения камеры и перемещения объектов

На рисунке 4, когда фон сменяется на «Фон в игре», глобальным переменным «standard X» и «standard Y» присваиваются значения, вымеренные для конкретно этого объекта. В данном случае мы видим код и числовые значения для правой половины фона уровня. Эти глобальные переменные используются во всех объектах уровня для расчета их расположения. Затем фон переносится на задний план и происходит самое главное: запускается цикл, в котором с помощью встроенной функции «mouse x» программа передает координаты курсора мыши, и проверяет, находится ли он у края экрана, а также учитывает, выйдет ли фон за дозволенные рамки, если переместить его в эту сторону. Само перемещение происходит после того, как подсчитывается новое значение для ранее указанных переменных, и передается команда «Координаты получены». Данная команда обрабатывается одновременно во всех элементах уровня, благодаря чему и происходит имитация перемещения камеры в пространстве.

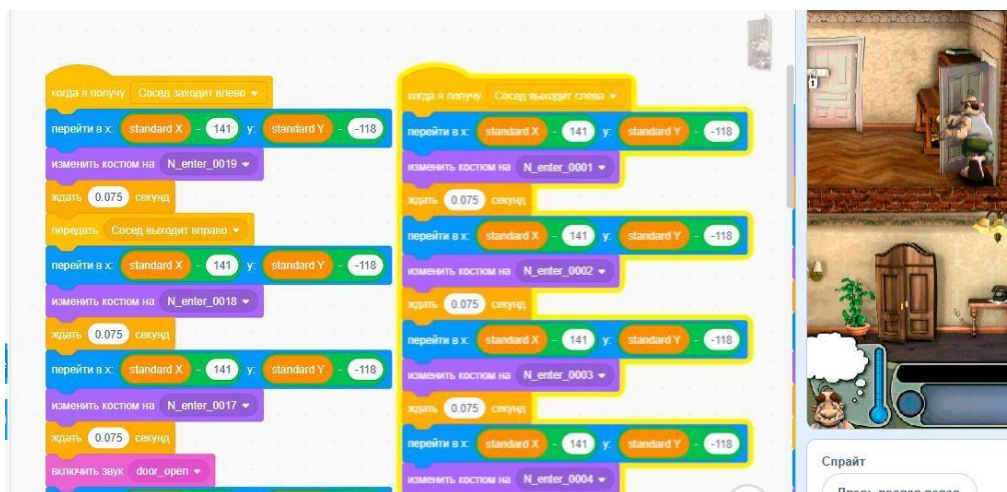


Рисунок 5 – Анимация открытия двери

На рисунке 5 видно, как устроена анимация открытия двери. При получении события «Сосед заходит», или «Сосед выходит», дверь начинает последовательно заменять свой «костюм» на кадры анимации из оригинальной игры с небольшой межкадровой задержкой. Костюмы – это изображения, загруженные как альтернативный вид объекта. В блоках, связанных с перемещением объекта, вписаны значения, которые позиционируют этот объект относительно фона. В некоторых случаях для разных кадров нужно вымерять разные значения, поскольку точка начала координат у объекта рассчитывается платформой в зависимости от формы, из-за чего происходят рывки во время переключения костюмов.

Были разобраны основные моменты, которые присутствуют в конечном проекте. Логика переключения между разделами меню и запуска уровней может быть использована во многих проектах, которые способны выполнить студенты. Механики, связанные с перемещением персонажей и использованием объектов имеют объемные системы блоков, а также специфическую логику, которая подойдет малому числу проектов. Весь проект для изучения доступен всем на платформе Scratch [4].

Когда генеративный искусственный интеллект может самостоятельно создать логику и функционал программы при создании кода, такие инструменты как Scratch могут способствовать честному выполнению заданий студентами. Главное, чтобы создаваемый проект был уникальным на данной платформе во избежание копирования чужих разработок.



Рисунок 6 – Известная игра «Getting over it», воссозданная в Scratch [5]

Подобный подход к обучению на ранних стадиях будет способствовать развитию у студентов навыков по созданию алгоритмов, адаптации к необычным условиям разработки, а также увеличивать интерес к специальности, поскольку тут не нужны знания кода и особенностей языков программирования для создания интересных проектов.

Когда студент получил начальный опыт разработки, получил знания о структурах данных, изучил основы некоторых языков программирования, можно включить в программу обучения открытое программное обеспечение для создания визуальных программ с конструкторами. Конструкторы ограничивают возможности нейросетей, генерирующих код. Просто попросить написать код не получится, ведь программа конструктора контролирует содержание файла. Для примера возьмем такой фреймворк для создания графических интерфейсов как «wxWidgets». В среде разработки со встроенным конструктором DialogBlocks. Здесь нужно построить визуальную часть приложения в конструкторе, настроить элементы в нем же, а затем в созданном программой коде добавлять код функциям. Интерфейс программы видно на рисунке 7.

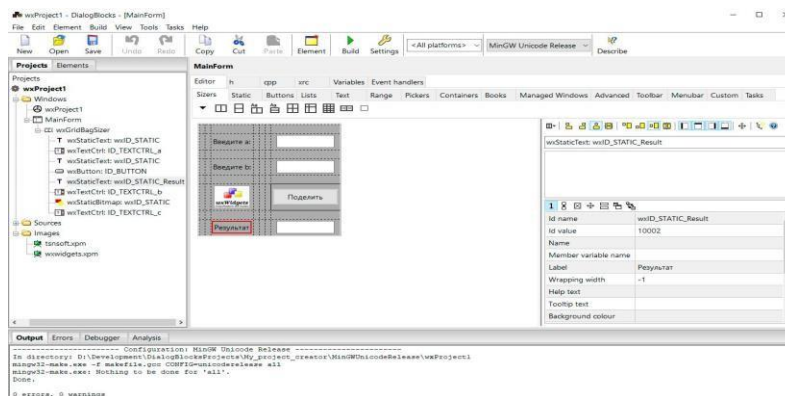


Рисунок 7 – Интерфейс программы DialogBlocks со встроенным конструктором

Подобный подход к обучению будет полезен уже тем, кто имеет опыт программирования, и должен более углубленно погружаться в разработку программных продуктов. А также это полезно для составления учебных программ, ведь текстовая нейросеть не может взаимодействовать с визуальным конструктором.

В статье были разобраны проблемы в современном обучении по направлению информационных технологий в условиях доступности генеративного искусственного интеллекта, а также разобраны несколько вариантов решения для разных этапов обучения студентов. Рассмотренные ранее инструменты разработки, которые усложняют использование нейросетей будут полезны в организации обучения. Их внедрение не составит большого труда для учебных заведений, ведь инструменты не требуют затрат, и имеют низкий порог вхождения. Главное, что нужно развивать в студентах – это понимание того, что они создают и какими методами нужно это делать. Нейросети должны выступать вспомогательным инструментом, а не заменять разработчика. Если каждый обучающийся разовьет алгоритмическое мышление, научится анализировать то, что ему предлагает ИИ, и извлекать из этого пользу, то развитие цифрового сектора в стране не заставит себя долго ждать.

#### **Список использованной литературы**

1. ChatGPT Statistics - 2023 // Demand Sage. – [Электронный ресурс]. – URL: <https://www.demandsage.com/chatgpt-statistics/>
2. 8 тысяч академических работ студенты Казахстана написали с помощью ИИ // Forbes Kazakhstan. – [Электронный ресурс]. – URL: <https://forbes.kz/articles/8-tysyach-akademicheskikh-rabot-studenty-kazahstananapisali-spomoshyu-ii>
3. Cybersecurity Risks of AI-Generated Code // CSET Georgetown. – [Электронный ресурс]. – URL: <https://cset.georgetown.edu/publication/cybersecurity-risks-of-ai-generated-code/>
4. Личный проект "Neighbours from hell" // Scratch. – [Электронный ресурс]. – URL: <https://scratch.mit.edu/projects/1069633190/>
5. Проект "Getting over it" // Scratch. – [Электронный ресурс]. – URL: <https://scratch.mit.edu/projects/389464290/>
6. wxWidgets Documentation // wxWidgets Official Site. [Электронный ресурс]. – URL: <https://www.wxwidgets.org/>