

ИСПОЛЬЗОВАНИЕ МЕЖПРЕДМЕТНЫХ СВЯЗЕЙ ПРИ ОБУЧЕНИИ ПРОГРАММИРОВАНИЮ В СРЕДЕ SCRATCH

Бабаева А.Б., Ораева О.О.,

студентки 3 курса ВГУ имени П.М. Машерова, г. Витебск, Республика Беларусь,

Научный руководитель – Булгакова Н.В., ст. преп.

Освоение разделов «Основы алгоритмизации и программирования» школьного курса информатики вызывает трудности у многих учащихся. Среди причин этого выделяют неуспешность детей или боязнь потерпеть неудачу; акцент на изучение информационных технологий, то есть формирование компьютерной грамотности (чаще всего под этим понимаются общие навыки работы на компьютере, использование средств вычислительной техники). Для того, чтобы свести к минимуму негативный эффект указанных причин и помочь учащимся успешно освоить программирование, важно создать условия, в которых ученики будут иметь возможности самореализации и самоутверждения. Использование в обучении среды программирования Scratch даже скучное обучение может превратить в интересное и увлекательное занятие, а также повысить интерес школьников к изучению сложного материала. В связи с этим актуальной является возможность объединить учебу и игру с использованием специально подобранных задач. Это позволяет открывать детям новые способы для самовыражения и поможет учиться, создавая свои игры и анимационные истории. Такой подход ускоряет и углубляет понимание учащимися структуры знаний, а также способствует поэтапному формированию практических навыков программирования.

Среда программирования Scratch представляет собой визуальную объектно-ориентированную среду, созданную для детей, которая имеет простой и удобный интерфейс. Программа в этой среде собирается, как конструктор, мозаика или пазлы [1]. Используя среду Scratch, можно не только программировать, но и создавать сложную модель, в которой взаимодействует несколько объектов.

Целью настоящей работы является рассмотрение возможностей использования специально подобранных задач при обучении Scratch-программированию младших школьников.

Материал и методы. Материал исследования – Scratch как язык и среда программирования; межпредметные связи информатики с математикой (геометрические фигуры) и рисованием (изображение фигур, работа с цветом, правила перспективы). В работе используются методы исследования экспериментально-теоретического уровня: анализ и синтез, изучение и обобщение, формализация.

Результаты и их обсуждение. Среда программирования Scratch представляет собой среду разработки программ простыми, интуитивно понятными и легко усваиваемыми средствами для детей, не имеющих опыта в программировании. Такое представление информации повышает степень вовлеченности учащегося в процесс получения знаний, информация быстрее воспринимается и легче запоминается. Особую значимость в обучении среда Scratch приобретает благодаря возможности использовать графику, создавать объекты и анимировать их.

Девиз Scratch – «Придумывай, создай, программируй!». Это следует принципу создания проекта: Сначала приходит идея («придумай»), потом программирование идеи в Scratch («создай»), а затем опубликование ее в Scratch-сообществе («программируй») [2].

Использование Scratch в обучении программированию требует от педагога продуманного и тщательно выбранного практического материала, использование которого допускает неоднозначное творческое решение. Мы рассматриваем возможность использования на уроках в начальной школе специально подобранных задач, опирающихся на межпредметные связи информатики с математикой и рисованием.

Сначала ученик должен выбрать геометрические фигуры, относящиеся к данной задаче, рассмотреть способы их объединения в объект. Вообще говоря, это творческий процесс, и одну и ту же задачу разные ученики могут решить по-разному. Например, ученикам предлагается набор геометрических фигур: круги, квадраты, прямоугольники, треугольники. Используя только эти фигуры (их размеры можно менять), нужно собрать объект, который затем будет использован как персонаж (спрайт). Раскрасить фигуры учащийся может самостоятельно. Для решения этой задачи в среде Scratch имеется встроенный графический редактор, который допускает работу с векторной и растровой графикой. На этом этапе навыков программирования

не потребуется. Например, из геометрических фигур учащимся нужно создать рисунок зайчика или человечка.

Сложность программирования растёт по мере увеличения сложности поставленной задачи. На втором этапе ученикам предлагается подготовить несколько костюмов для персонажа и создать анимацию. Например, зайчик, которого создали в предыдущем задании, должен выполнить прыжок, а человечек должен выполнить шаг.

На третьем этапе задача усложняется – нужно «научить» объект выполнять какое-то более сложное действие. Например, зайчик должен выполнить три прыжка, перемещаясь слева направо, а человечек должен выполнить несколько шагов справа налево (или в другом направлении).

Такой подход допускает дальнейшее усложнение задания. Например, усложнить задачу можно добавлением еще одного персонажа. Персонажи должны взаимодействовать. Дальнейшее усложнение задачи, например, появление какого-то события (поменялся цвет светофора, раздался какой-то звук, пошел снег, дождь, появилась радуга и т.д.), мотивирует учеников к более глубокому изучению возможностей среды Scratch.

Заключение. Решая подобные задачи, дети учатся придумывать и реализовывать различные объекты (персонажи), определять, как они выглядят в разных условиях, перемещать их по экрану, устанавливать способы взаимодействия между ними. При этом младшие школьники могут создавать проекты, не требующие сложной алгоритмической структуры. Свои творческие способности они могут выразить через составление сценария проекта, использование графических и звуковых компонент. Появляется мотивация к изучению новых возможностей как самой среды Scratch, так и новых, более сложных, алгоритмических конструкций. Их освоение может осуществляться учениками самостоятельно, без участия педагога.

1. Горбунова Т.В., Леонова Т.А. Педагогические технологии в обучении студентов – будущих учителей информатики программированию в среде Scratch // Вестник Шадринского государственного педагогического университета. – № 3 (43), 2019. – С. 105–110.
2. Scratch.mit.edu [Электронный ресурс]: образоват. портал. – Режим доступа: <https://scratch.mit.edu/>.

ПРОГРАММЫ РАСЧЕТА ЭЛЕКТРОННО-ОПТИЧЕСКИХ СИСТЕМ И АНАЛИЗ ИХ ПРИМЕНЕНИЯ

Бирюкова Д.В.,

магистрант ВГУ имени П.М. Машерова, г. Витебск, Республика Беларусь

Научный руководитель – Антонович Д.А., канд. техн. наук, доцент

В настоящее время, для развития и быстрого роста множества отраслей промышленности и науки требуются высокоэффективные способы обработки и получения материалов с новыми свойствами. Одной из прогрессивных технологий применяющейся для этих целей является электронно-лучевая обработка материалов.

Данная технология позволяет реализовать широкий спектр воздействий на материалы, например, закалка металлов и нанесение на них защитного покрытия, сварка, и ряд других, что в совокупности с высоким КПД преобразования энергии позволило найти применение электронно-лучевым технологиям в различных направлениях науки и техники: химической промышленности, металлургии, электронной промышленности, машино- и приборостроении, медицине [1].

Целью исследования является анализ существующих пакетов прикладных программ для моделирования электронно-оптических систем, выявление их преимуществ и недостатков.

Материал и методы. Материалом для исследования послужили программные обеспечения для математического моделирования электронно-оптических систем. При проведении исследований применялись общепризнанные методы научного познания.

Результаты и их обсуждение. Основой любого электронно-лучевого устройства является электронно-оптическая система (ЭОС), использующаяся для формирования электронного пучка с переделёнными параметрами. Работа ЭОС строится на таких физических явлениях как эмиссия электронов и движение заряженных частиц в электромагнитных полях.

Для уменьшения финансовых и временных затрат на разработку новых электронно-лучевых устройств применяются технологии компьютерного моделирования, в частности моделирования ЭОС для создания необходимой конфигурации луча.