



Визуальное программирование в KODU: первый шаг к ИТ-образованию

Актуальность. В настоящее время область информационных технологий стала важнейшим сектором экономики, во многом определяющим темпы научно-технического прогресса. Мировая конкуренция между странами определяется темпами внедрения этих технологий, поскольку именно внедрение инноваций и новых технологий обеспечивает в экономически развитых странах 90% ежегодного прироста внутреннего валового продукта, а знания ИТ-специалистов оцениваются как интеллектуальный ресурс развития общества. В связи с этим, задача подготовки высокопрофессиональных кадров, способных развивать новые информационные технологии и эффективно использовать их на практике становится стратегически важной для прогресса общества.

Однако в последнее время в компьютерной и образовательной среде все чаще поднимаются вопросы, связанные с проблемами подготовки ИТ-специалистов. Очевиден факт: программы обучения в школах и вузах не согласованы с потребностями рынка, в то время как, по мнению аналитиков ведущих кадровых агентств, Россия вошла в число стран с устойчивым развитием ИТ-рынка. О кризисе системы ИТ-образования говорят повсеместно: проблема настолько остра, что становится помехой развитию отрасли в целом.

Осознание этой проблемы способствует выработке конструктивных путей ее решения. И первый шаг в процессе создания системы отечественного ИТ-образования должен быть сделан образовательными учреждениями общего образования. Речь необходимо вести **о введении специализированных пропедевтических курсов в области ИТ-технологий**, которые ориентировали бы обучающихся на выбор наукоемких профессий, связанных с ИТ-индустрией.

В указанном контексте очень важна целенаправленная профориентационная деятельность, которая должна носить практикоориентированный, комплексный и многоступенчатый характер. Начинать профориентационную работу в области ИТ-образования необходимо уже **на основной (и даже начальной!) ступени общего образования**.

Что можно сказать о детях, родившихся в XXI веке? Они достаточно уверенно чувствуют себя в цифровом мире, быстро привыкают пользоваться (приобретают, владеют) различными технологиями, программными продуктами, гаджетами... Взрослые удивляются тому, что для детей совершенно естественно. Но эта привычка может оказаться пагубной. Ведь редко кто из них задумывается о том, **кем и как** были созданы или разработаны такие привычные для них вещи (те же компьютерные игры!). И задача взрослых – разбудить в них желание творить! Идею, сформулированную Ф. Ницше «Только для созидания должны вы учиться!», необходимо донести до каждого сидящего за школьной партой. Вот смысл учения, вот та благородная миссия, к которой должен быть готов обучающийся, – вырасти и изменить мир, сделать его лучше.

При этом **нужно создать условия, в которых бы обучающиеся смогли реализовать свои потенциальные возможности.**

Одним из реальных направлений такой практикоориентированной профориентационной работы на начальной и основной ступенях общего образования является внедрение **пропедевтического курса изучения визуального языка программирования Kodu, предназначенного для создания компьютерных игр.**

Общая характеристика программного продукта.¹ Kodu - это визуальная среда для создания **казуальных** игр без программирования для персональных компьютеров, ориентированная на детскую (наиболее успешно применяют Kodu дети от 8 лет) и подростковую аудиторию.

Для справки (по материалам Википедии):

Казуальная игра — компьютерная игра, предназначенная для широкого круга пользователей. Сам термин «казуальная» происходит от лат. *casualis*, что означает «случайный». Таким образом, казуальная игра — это игра, в которую играют от случая к случаю, между делом, чаще всего — чтобы как-то «убить» время. Ввиду своего предназначения такая игра, как правило, обладает достаточно простыми правилами и не требует от пользователя хорошего владения компьютером. Многие подобные игры обладают также яркой привлекательной графикой и минимумом текста.

Kodu — многофункциональный инструмент для **нарративного** творчества, который за счет дружелюбного интерфейса мотивирует к конструированию различных миров: выбору объектов и среды их обитания, моделированию поведения объектов, условий действий, отношений между разными объектами и т.п. Kodu демонстрирует творческий аспект программирования.

Для справки (по материалам Википедии):

Нарратив (англ. и фр. — *narrative*) — исторически и культурно обоснованная интерпретация некоторого аспекта мира с определенной позиции. В литературе нарратив — линейное изложение фактов и событий в литературном произведении, то есть то, как оно было написано автором. Синонимами сравнительно нового для русского языка термина «нарратив» являются более традиционные «**повествование**» и «**рассказ**».

Drag-and-drop программирование используется с так называемым языком высокого уровня Kodu Language. С его помощью можно управлять параметрами и настройками игрового мира, коллизиями, цветом и визуализацией.

Графический интерфейс обладает высокой степенью прозрачности, за счет чего пользователь легко манипулирует исполнителями на экране, создает поведение игрока, управляет визуальным рядом, звуками и сценарием.

Для создания миров используется библиотека, включающая более 200 стандартных игровых сценариев и базовых игровых элементов, редактор игрового поведения, 20 различных персонажей с различными способностями.

Манипулируя исполнителями в их виртуальных мирах, обучающиеся получают первоначальный опыт работы с базовыми алгоритмическими структурами: при программировании в Kodu выбираются визуальные фрагменты для условия (WHEN) и действия (DO), т.е. осуществляется пропедевтика базовых алгоритмических конструкций.

¹ Kodu Game Lab. [Электронный ресурс] – Режим доступа: <http://gcup.ru/load/kodu/2-1-0-1504>

Преимуществом среды Kodu является возможность визуализации написанного сценария, что позволяет установить соответствие между планируемыми действиями исполнителя и его реальным поведением.

Условия реализации пропедевтического курса визуального программирования в Kodu. Организовать полноценное раннее обучение в ИТ-сфере в рамках обязательного школьного образования в соответствии с государственными образовательными стандартами общего образования **первого поколения** не представлялось возможным из-за жесткого распределения часов в Базисных учебных планах. Это являлось одной из проблем школьного технологического образования.

Реализация Федеральных государственных образовательных стандартов **второго поколения** открыла новые возможности для развития ИТ-образования. В соответствии с ФГОС определенное количество часов (например, для 5 классов – 7 часов в неделю) учебной нагрузки школьника отводится **на внеурочную деятельность**. Основная идея такого подхода заключается в создании условий для реализации потенциальных возможностей обучающихся, выбора каждым из них индивидуального образовательного маршрута в соответствии с личностными характеристиками и потребностями (мотивом, особенностями, способностями и т.п.).

Именно поэтому, во внеурочной деятельности видится необходимым введение подобных пропедевтических курсов в рамках функционирования **клубов или кружков**, связанных с изучением основ программирования, поскольку эта область ИТ-сферы является базовой.

Не менее актуальной формой реализации программы является организация в образовательных учреждениях общего образования **платных дополнительных образовательных услуг** – кружков, клубов, и т.п., ориентированных на организацию интеллектуального досуга обучающихся, связанного с приобретением первоначальных навыков программирования.

Кроме того, среда визуального программирования Kodu может быть рекомендована к внедрению **в учреждениях дополнительного образования**.

Подобные мероприятия имеют высокую профориентационную значимость и направлены на удовлетворение образовательных потребностей определенной целевой группы обучающихся и их родителей.

Кто должен осуществлять такую профориентационную работу? Взрослые.... Учителя и родители в первую очередь. Но их тоже этому надо научить. Надо показать возможности реализации таких пропедевтических курсов в области ИТ-технологий. И здесь особая миссия делегируется студентам – волонтерам, которые, с одной стороны, быстро адаптируются к цифровым продуктам, с другой – уже готовы вести некую просветительскую деятельность.

Планируемые результаты. Целью предлагаемого пропедевтического курса обучения программированию является формирование интереса обучающихся к инженерно-технологическому образованию в области ИТ-сферы (в частности, программирования) че-

рез организацию проектной деятельности, ориентированной на создание интеллектуального продукта.

Как отмечается в Проекте примерной программы по информатике для основной школы, «сегодня человеческая деятельность в технологическом плане меняется очень быстро, на смену существующим технологиям и их конкретным техническим воплощениям быстро приходят новые, которые специалисту приходится осваивать заново. В этих условиях велика роль **фундаментального образования**, обеспечивающего **профессиональную мобильность человека**, готовность его к освоению новых технологий, в том числе информационных. Поэтому **в содержании курса информатики основной школы целесообразно сделать акцент на изучении фундаментальных основ информатики, выработке навыков алгоритмизации, реализовать в полной мере общеобразовательный потенциал этого курса**»¹.

Использование среды визуального программирования Kodu может способствовать реализации следующих целей изучения учебного предмета «Информатика», обозначенных в Примерной программе по информатике для основной ступени общего образования в качестве приоритетных:

- формирование информационной и **алгоритмической культуры**;
- формирование представления об основных изучаемых понятиях: алгоритм, модель и их свойствах;
- развитие алгоритмического мышления, необходимого для профессиональной деятельности в современном обществе; развитие умений составлять алгоритм для конкретного исполнителя; формирование знаний об алгоритмических конструкциях, знакомство с основными алгоритмическими структурами — линейной, условной и циклической и др.

Основные **предметные результаты обучающихся**, формируемые в процессе проектирования в среде Kodu, включают:

- умение использовать термины «алгоритм», «программа»; понимание различий между употреблением этих терминов в бытовой речи и в информатике;
- умение составлять неветвящиеся (линейные) алгоритмы управления исполнителями;
- умение формально выполнять алгоритмы, описанные с использованием конструкций ветвления (условные операторы) и повторения (циклы), вспомогательных алгоритмов;
- умение создавать и выполнять программы для решения несложных алгоритмических задач в выбранной среде программирования;
- умение использовать готовые прикладные компьютерные программы и сервисы в выбранной специализации, умение работать с описаниями программ и сервисами и др.

Учитель получает уникальную среду, через призму деятельности в которой, обучающийся имеет возможность осознать сущность и природу таких базовых понятий информатики как «алгоритм», «исполнитель», «программа», «подпрограмма», «модель» и др. Обучающиеся в игровой форме постигают суть **объектно-ориентированного программирования**, знакомясь с понятиями «объект», «класс», «родитель», что является теоретическим фундаментом для освоения базовых понятий «инкапсуляция», «полиморфизм», «наследование».

В частности, обучающиеся осваивают процесс управления исполнителем на основе системы команд этого исполнителя, получают опыт моделирования среды и деятельность исполнителя

¹ Примерная программа по информатике для основной школы. [Электронный ресурс] – Режим доступа: <http://standart.edu.ru/catalog.aspx?CatalogId=8421>

в зависимости от условий, отношений между объектами; осваивают один из основных подходов оптимизации структуры программы – **написание и вызов подпрограмм** (вспомогательных алгоритмов).

Моделирование в этом случае выступает **средством познания**: обучающийся «выступает в роли создателя, разработчика моделей, которые в силу этого отражают личностные факторы, особенности ассоциативного мышления обучаемого, его опыт, мотивы и предпочтения»¹.

С другой стороны модель может выступать и как «**объект изучения**», поскольку любая модель может рассматриваться как новый конструктивный объект, обладающий своими свойствами и характеристиками. Для разных моделей можно выделить их инвариантные свойства, особенности, накладываемые выбранным способом представления объекта моделирования»².

Рассматриваемые вопросы составляют инвариант как непрофильного, так и профильного высшего профессионального ИТ-образования.

Указанные аспекты реализации дидактического потенциала визуальной среды программирования Kodu позволяют сделать выводы, что внедрение данной среды в образовательный процесс создает объективные условия для ранней профилизации обучающихся (уже на ступени начального и общего образования), формирования мотивации у обучающихся к получению ИТ-образования **через получение практического опыта**.

Кроме того, априори можно утверждать, что обучение визуальному программированию в среде Kodu является естественной средой для развития широкого спектра метапредметных результатов школьника, что может быть отражено в Программе формирования универсальных учебных действий образовательного учреждения.

Так, в ходе разработки сценария игры и его практической реализации обучающиеся учатся управлять своей деятельностью; контролировать ее и вносить свои коррективы; проявлять инициативность и самостоятельность; создаются естественные условия для формирования умений понимать причины успеха/неуспеха учебной деятельности и способности конструктивно действовать даже в ситуациях неуспеха.

Именно работа над созданием продукта способствует формированию навыков решения проблем творческого и поискового характера, планирования учебных действий в соответствии с поставленной задачей. И именно в ходе программирования обучающийся учится оценивать эффективность способов достижения результата, выбирать оптимальный вариант и аргументировать свой выбор.

Само планирование проектной деятельности ориентировано на вовлечение учащихся в деятельность по постановке целей, анализу и управлению процессом обучения во время проведения учебного проекта и саморефлексии после его завершения. Освоение начальных форм познавательной и личной рефлексии - одна из основных задач, сформулированных в Федеральном государственном образовательном стандарте начального и основного общего образования.

Таким образом, процесс программирования, включая процесс разработки сценария, создания миров и моделирования поведения объектов, естественным образом способствует формированию следующих регулятивных универсальных учебных действий:

¹ Бешенков С.А., Ракитина Е.А. Моделирование и формализация. М.: Лаборатория Базовых знаний, 2002. — 336 с.

² Там же

- умение ставить и формулировать для себя новые задачи; развивать мотивы и интересы своей познавательной деятельности;
- умение самостоятельно планировать пути достижения целей, в том числе альтернативные, осознанно выбирать наиболее эффективные способы решения задач;
- умение соотносить свои действия с планируемыми результатами, осуществлять контроль своей деятельности в процессе достижения результата, определять способы действий в рамках предложенных условий и требований, корректировать свои действия в соответствии с изменяющейся ситуацией;
- умение оценивать правильность выполнения учебной задачи, собственные возможности её решения;
- владение основами самоконтроля, самооценки, принятия решений и осуществления осознанного выбора в соответствии с поставленными целями.

Кроме того, отличительной особенностью организации работы над проектами при обучении программированию является ориентация на кооперированную деятельность школьников, на развитие его коммуникативных универсальных учебных действий:

- умения организовывать продуктивное сотрудничество и совместную деятельность с учителем и сверстниками;
- работать индивидуально и в группе: находить общее решение и разрешать конфликты на основе согласования позиций и учёта интересов;
- формулировать, аргументировать и отстаивать своё мнение и т.п.

Обучающиеся учатся работать в команде, выполнять разные роли и обязанности. Все это становится частью их жизни. Приобретаемый ими социальный опыт и навыки оказывается практически значимыми.

Априори можно утверждать, что данный пропедевтический курс обучения визуальному программированию ориентирован на формирование планируемых в ФГОС образовательных результатов обучающихся.

2013, О.Ф. Брыксина, к.п.н., доцент, эксперт образовательных программ компаний Intel, IBM, SMART, зав. кафедрой ИКТО Поволжской государственной социально-гуманитарной академии. Автор 15 учебно-методических пособий и 58 научных статей по проблемам реализации дидактических функций средств ИКТ в образовательном процессе, формирования ИКТ-компетентности педагога и создания информационной среды образовательного учреждения, а также авторских программ курсов повышения квалификации: «Реализация интеллектуального потенциала и формирование общей культуры школьников средствами информационно-коммуникационных технологий», «Технологии подготовки учащихся к виртуальным олимпиадам и проектам», «ИКТ-компетентность педагога как условие подготовки социально-адаптируемой личности школьника», «Информатика и ИКТ в начальной школе» и др., которые реализуются в рамках Приоритетного национального проекта «Образование».