

УДК 372.800.4

ПРОЕКТИРОВАНИЕ МОДЕЛИ ОБУЧЕНИЯ ПРОГРАММИРОВАНИЮ В ОСНОВНОЙ ШКОЛЕ НА ОСНОВЕ СЕМИОТИЧЕСКОГО ПОДХОДА

Гребнева Д.М.

ФГАОУ ВПО «Российский государственный профессионально-педагогический университет»
(филиал), Нижний Тагил, e-mail: grebdash@gmail.com

В статье рассматривается процесс проектирования модели обучения программированию в основной школе. Предложенная модель включает в себя четыре компонента: целевой, программно-технический, структурно-содержательный и критериальный. В целевом компоненте выделены личностные, метапредметные и предметные цели обучения. В программно-техническом компоненте рассмотрены элементы робототехники в качестве эффективного средства обучения программированию школьников. В структурно-содержательном компоненте предложено содержание раздела «Алгоритмы и элементы программирования» с элементами робототехники. В критериальном компоненте определены и описаны уровни владения учащимися программированием. Модель методики обучения программированию может быть использована для 7–9 классов общеобразовательных школ.

Ключевые слова: модель обучения программированию, семиотический подход, робототехника, знаково-символические универсальные учебные действия, содержание обучения программированию в основной школе

THE DESIGN OF THE SEMIOTIC APPROACH BASED MODEL OF TEACHING PROGRAMMING IN SECONDARY SCHOOLS

Grebneva D.M.

FGAOU VPO «The branch of the Russian State Vocational Pedagogical University in Nizhniy Tagil city», Nizhniy Tagil, e-mail: grebdash@gmail.com

The article discusses the process of designing the model of teaching programming in secondary schools. The proposed model consists of four components which are target, software and hardware, structural-content and criteria components. In the target component personal, metasubject and subject learning aims are marked. In the software and hardware components elements of robotics as effective means of teaching students programming are considered. In the structural-substantial component content of the section «Algorithms and programming elements» with elements of robotics are suggested. In the criteria component the levels of knowing programming are defined and described. Model of teaching programming can be used for 7–9 classes of secondary schools.

Keywords: model of teaching programming, semiotic approach, robotics, signs and symbols universal educational activities, the content of teaching programming in secondary schools

Применение семиотического подхода в информатике рассматривается О.Ф. Брыкиной, Н.А. Кургановой, В.И. Фоминым и др. По отношению к обучению программированию семиотический подход можно представить себе как позицию обучающегося, знакомого с некоторыми знаковыми системами (русским языком, дорожными знаками, позиционными системами счисления) и желающего узнать, чем выделяются такие знаковые системы, как языки программирования [5].

При построении методики обучения информатике на основе семиотического подхода необходимо учитывать, что фиксированными являются цели и содержание обучения. Данные компоненты изначально заложены в нормативных документах образования: федеральном государственном образовательном стандарте, фундаментальном ядре общего образования и др.

Проектирование модели методики обучения информатике на основе семиоти-

ческого подхода, включает в себя решение следующих задач:

- конкретизация целей обучения (целевой компонент);
- определение программных и технических средств, способствующих достижению поставленных целей (программно-технический компонент);
- определение содержания обучения, конструирование блоков учебной информации и их расположение в логической последовательности (структурно-содержательный компонент);
- определение требований к уровню освоения содержания обучения (критериальный компонент).

Целевой компонент

В настоящее время, согласно ФГОС основной школы цели обучения имеют многоуровневый характер: предметные, метапредметные и личностные цели.

В концепции ФГОС под личностными результатами понимается «сформировавшаяся в образовательном процессе система ценностных отношений обучающихся к себе, другим участникам образовательного процесса, самому образовательному процессу и результатам» [1].

К *личностным целям* обучения информатике можно отнести:

- развитие мотивации к изучению информатики;
- развитие навыков саморегуляции и планирования деятельности.

Под *метапредметными результатами* понимаются «освоенные обучающимися на базе одного, нескольких или всех учебных предметов способы деятельности, применимые как в рамках образовательного процесса, так и при решении проблем в реальных жизненных ситуациях» [1].

Следует отметить, что практически все задачи по программированию носят метапредметный характер и требуют привлечения знаний из других областей (математики, физики, технологии и др.). Работа с разными знаковыми системами способствует развитию умений учеников создавать, преобразовывать знаки и символы для решения учебных и познавательных задач.

Таким образом, к *метапредметным целям* обучения программированию мы относим:

- научить применять полученные знания для решения практических задач как из области информатики так и других областей;
- развивать знаково-символическую деятельность, которая позволит учащимся легко ориентироваться в большом объеме информации, выделять главное и второстепенное;
- развивать коммуникабельность, которая проявляется в умении учащихся письменно и устно излагать проблемы, идеи решения задач, анализировать и обсуждать полученные результаты.

Под предметными результатами образовательной деятельности понимается «усвоение обучающимися конкретных элементов социального опыта, изучаемого в рамках отдельного учебного предмета, - знаний, умений и навыков, опыта решения проблем, опыта творческой деятельности» [1].

К предметным целям обучения информатике, раздел «Алгоритмы и элементы программирования», согласно примерной программе по информатике для 7–9 классов, относятся:

- умение составлять линейные алгоритмы управления исполнителями и записывать их на выбранном языке программирования (знание синтаксиса языка программирования);

- умение формально выполнять алгоритмы, описанные с использованием конструкций ветвления и повторения, вспомогательных алгоритмов, простых и табличных величин (знание семантики языка программирования);

- умение создавать и выполнять программы для решения несложных алгоритмических задач в выбранной среде программирования (прагматический аспект программирования).

Следует отметить, что предметные цели обучения разделу «Алгоритмы и элементы программирования» соотносятся с семиотическим подходом.

Таким образом, к *предметным целям* обучения информатике (на примере программирования) мы относим:

- научить обучающихся записывать основные алгоритмические конструкции (согласно стандарту это линейный алгоритм, оператор ветвления, циклы) на выбранном языке программирования;

- научить обучающихся понимать смысл алгоритмических конструкций, «читать» и формально исполнять структуры, записанные на языке программирования;

- научить решать практически значимые задачи с использованием языка программирования;

- научить оценивать эффективность используемых алгоритмов.

Достижение вышеперечисленных предметных, метапредметных и личностных целей будет иметь результатом развитие предметной учебной успешности учащихся.

Программно-технический компонент

Согласно федеральному закону об образовании учитель свободен в выборе педагогически обоснованных средств обучения [2, ст. 47 п. 3.2]. В процессе обучения немаловажную роль играют технические средства обучения, с помощью которых учитель имеет возможность показать развитие явлений, их динамику, сообщать учебную информацию определенными блоками и управлять индивидуальным процессом усвоения знаний. В качестве рекомендуемой системы средств обучения информатике в школе можно, согласно М.П. Лапчику [6], можно рассматривать:

- компьютеры и компьютерные классы, периферийное оборудование;
- программно-методическое обеспечение курса информатики;
- объектно-ориентированные программные системы;
- учебное, демонстрационное оборудование;

– телекоммуникационное оборудование, обеспечивающее доступ в Интернет и другие локальные сети.

Например, для содержательной линии «Алгоритмы и элементы программирования» система средств обучения может быть следующей:

– компьютеры, компьютерный класс, драйвер и кабель для подключения учебного робота к компьютеру;

– примерная программа по информатике, практикум по основам робототехники, разработанный учителем план, конспект урока, раздаточный материал для обучающихся;

– среды Microsoft Robotics Developer Studio, Arduino.

– учебные робототехнические наборы (Lego NXT, Pot-bot Parallax).

С нашей точки зрения, при изучении учебного материала содержательной линии «Алгоритмы и элементы программирования» эффективным является использование элементов образовательной робототехники. Рассмотрим возможности параллельного изучения программирования и робототехники в 7–9 классах. Как показывает школьная практика, программирование является объективно сложным предметом и учащиеся слабо мотивированы на его изучение. Введение элементов робототехники при изучении программирования позволит заинтересовать учащихся, разнообразить учебную деятельность, использовать групповые активные методы обучения. Следует отметить, что совместное изучение программирования и робототехники на западе приобретает все большую популярность и дает положительные результаты, о чем свидетельствует ряд исследований.

Робототехника является интересной для учащихся с точки зрения новизны, актуаль-

ности содержания, способствует развитию алгоритмического мышления, умению применять свои навыки для решения проблем реального мира. Использование элементов робототехники при обучении программированию способствует повышению уровня мотивации учащихся к предмету, более легкому пониманию принципов действия алгоритмических конструкций, содействует развитию умений самостоятельно и творчески думать.

Структурно-содержательный компонент

Содержание обучения информатике определяется ФГОС и примерной программой по информатике. Например, на изучение раздела «Алгоритмы и элементы программирования» отводится 22 часа. Выделяют четыре модуля: «Базовые понятия», «Утверждения. Логические значения», «Основные конструкции алгоритмических языков», «Решение задач на составление алгоритмов и программ».

Как было отмечено выше, введение элементов робототехники позволяет реализовать принцип изучения знаковых систем по возрастающим степеням абстрактности и способствует более эффективному усвоению знаний учащимися с разными ведущими каналами восприятия, развивать мотивацию к изучению предмета, тем самым создавая хорошие условия для развития предметной учебной успешности. Согласно принципу дискретности осмысления, учебная информация должна излагаться учащимся порционно и в определенной логической последовательности.

Учитывая целесообразность параллельного изучения программирования с элементами робототехники [3], рассмотрим возможное содержание (таблица) раздела «Алгоритмы и элементы программирования».

Содержание раздела «Алгоритмы и элементы программирования»

Содержание	Количество часов	Возможности использования робототехники
Повторение. Понятие алгоритма и исполнителя.	1	Знакомство с учебными роботами, их системой команд
Линейные алгоритмические конструкции	2	Изучение алгоритма движения робот по прямой на заданное расстояние (время)
Ветвление	4	Получение роботом информации. Сенсоры. Выполнение действия роботом в зависимости от условия
Циклы	4	Использование циклов в решении типовых задач робототехники
Вспомогательные алгоритмы	4	Встроенные и пользовательские функции в управлении поведением робота
Массивы	4	Массивы в управлении движением робота. Задача поиска кратчайшего пути
Работа над групповым проектом	3	Составление программ с помощью изучаемого языка программирования и реализация их для роботов-симуляторов (например, LegoNXT, iRoombCreate, Voe-Bot) или реальных роботов
Итого	22	

Блок 1. Повторение. На данном этапе с учащимися повторяются основные понятия: алгоритм, исполнитель, система команд. Рассматривается робот как исполнитель, его система команд и конструктивные особенности.

Блок 2. Линейные алгоритмические конструкции.

2.1. Демонстрация линейных программ управления роботом. Организация линейного алгоритма на языке VPL. Компоненты «Data», «Simple Dialog». Сервис «Generic DifferentialDrive». Сравнение диаграммы VPL и блок-схем.

2.2. Определение линейного алгоритма. Примеры. Запись линейного алгоритма на языке блок-схем. Среда Visual C++. Библиотека «stdio.h». Ввод и вывод данных: scanf(), printf(). Значение знака «&» в операторе scanf(). Выполнение арифметических операций.

2.3. Линейный алгоритм в управлении поведением робота. Написание программы управлением робота на языке Си.

Блок 3. Ветвление.

3.1. Понятие о ручном и программном управлении роботом. Управление роботом с помощью пульта.

3.2. Организация ветвления на языке VPL. Блок If. Входные и выходные данные блока If. Сервис «Contact». Организация и настройка связей. Разбор задачи «Стоп по условию».

3.3. Синтаксис и семантика структуры if на языке Си. Логические условия. Сложные условия. Операции И(&&), ИЛИ (||), НЕ (!). Задачи на выбор условия.

3.4. Составление программы управления движением роботом на языке Си.

Блок 4. Циклы.

4.1. Демонстрация циклических программ управления роботом. Организация циклов на языке VPL. Определение переменной. Организация счетчика.

4.2. Разбор задачи на использование циклов в управлении движением робота, например, «Объезд препятствий подсчет попыток».

4.3. Синтаксис и семантика структур for и while. Повторение действий. Примеры задач: организация счетчика, подсчет суммы.

4.4. Написание программы на языке Си.

Блок 5. Вспомогательные алгоритмы.

5.1. Демонстрация программ управления роботом с использованием встроенных и пользовательских функций

языка VPL. Создание пользовательского блока на языке VPL. Условие движения робота по окружности.

5.2. Разбор задачи «Движение робота по окружности» на языке VPL для робота-симулятора.

5.3. Синтаксис и семантика структур void() и function(). Создание простейших функций: сравнение двух чисел, обмен значениями. Графические процедуры: рисование треугольника.

5.4. Написание программы управления поведением робота на языке Си.

Блок 6. Массивы.

6.1. Демонстрация программ управления поведением робота с использованием циклов. Работа с массивами на языке VPL. Компонент ByteArray: ввод, вывод элементов массива. Применение массивов в решении задач.

6.2. Разбор задачи «поиск кратчайшего пути».

6.3. Массивы в языке Си. Синтаксис и семантика структуры array().

6.4. Написание программы управления поведением робота на языке Си.

Блок 7. Работа над групповым проектом.

Итоговый проект может представлять собой задания на конструирование робота, на ручное и программное управление.

Критериальный компонент

Критический уровень: Обучающийся понимает значение отдельных символов, однако составление конструкций вызывает у него сложности. Не умеет читать схемы, графики, коды программ.

Низкий уровень. Обучающийся успешно может выполнять замещение, то есть переносить функции некоторого объекта или процесса на знаково-символическое средство. В изучении программирования это означает, что обучающиеся знают синтаксис и семантику алгоритмических конструкций, но не могут соотнести их с условием решаемых задач.

Средний уровень. Обучающийся знает синтаксис и семантику алгоритмических конструкций, применяет их для решения аналогичных или тривиальных задач программирования. Может проверять листинг программы на синтаксические и семантические ошибки, однако не может анализировать созданную программу на эффективность.

Желаемый уровень. Знает синтаксис, семантику и прагматику языка программирования. Умеет моделировать решение задачи по программированию с использованием

знаково-символических средств. Применяет алгоритмические конструкции для решения творческих задач по программированию. Умеет анализировать листинг программы на синтаксические, семантические ошибки и оценивать эффективность алгоритма.

Обобщая вышесказанное отметим, что предложенная модель обучения программированию в основной школе предполагает включает в себя четыре основных компонента (целевой, программно-технический, структурно-содержательный и критериальный) и предполагает использование средств робототехники. Данная модель может быть использована для 7–9 классов общеобразовательных школ.

Список литературы

1. Брыксина О.Ф. Формирование профессиональной готовности студентов педагогического колледжа к реализации содержания образования на уровне учебного предмета «Информатика». – Самара, 1999. – 226 с.
2. Губанова М.М. Учитель и Ученик: грамотность, компетентность, технологичность. – Кемерово: КРИПКиПРО, 2013. – 167 с.
3. Гребнева Д.М. Обучение школьников программированию на основе семиотического подхода: авт. дис. ... канд. пед. наук. – Екатеринбург, 2014. – 23 с.
4. Закон Российской Федерации «Об образовании». – URL: <http://минобрнауки.рф/документы/2974> (дата обращения 25.12.2015)
5. Кауфман В.Ш. Языки программирования: концепции и принципы. – М.: ДМК-пресс, 2010. – 464 с.
6. Курганова Н.А. Развитие знаково-символической деятельности учащихся в процессе обучения информатике на основе семиотического подхода: автореферат дис. ... канд. пед. наук: 13.00.02. – Омск, 2006. – 25 с.
7. Лапчик М.П., Семакин И.Г., Хеннер Е.К. Методика преподавания информатики – М.: Академия, 2001. – 624 с.
8. Фомин В.И. Развитие содержания подготовки к информационно-аналитической деятельности на основе семиотического подхода: дис. ... д-ра пед. наук: 13.00.02. – Самара, 2009. – 385 с.