

МИНИСТЕРСТВО ОБРАЗОВАНИЯ И НАУКИ РФ  
федеральное государственное бюджетное образовательное учреждение высшего  
образования  
КРАСНОЯРСКИЙ ГОСУДАРСТВЕННЫЙ ПЕДАГОГИЧЕСКИЙ УНИВЕРСИТЕТ  
им.В.П.АСТАФЬЕВА  
(КГПУ им.В.П.Астафьева)

Институт/факультет Институт математики, физики и информатики  
(полное наименование института/факультета/филиала)  
Выпускающая кафедра Базовая кафедра информатики и  
информационных технологий в образовании  
(полное наименование кафедры)

КАРЕЛИН ГРИГОРИЙ СЕРГЕЕВИЧ

ВЫПУСКНАЯ КВАЛИФИКАЦИОННАЯ РАБОТА

Тема «Развитие алгоритмической культуры в процессе обучения робототехнике в  
основной школе»  
Направление подготовки 44.03.01 Педагогическое образование  
(код и наименование направления)  
Профиль Информатика  
(наименование профиля для бакалавриата)



ДОПУСКАЮ К ЗАЩИТЕ

Заведующий кафедрой  
д.п.н., профессор Пак Н.И.  
(ученая степень, ученое звание, фамилия, инициалы)

11.06.2018  
(дата, подпись)

Руководитель Т.А. Яковлева  
к.п.н., доцент Яковлева Т.А.  
(ученая степень, ученое звание, фамилия, инициалы)

Дата защиты 26.06.2018

Обучающийся Карелин Г.С.  
(фамилия, инициалы)

11.06.2018  
(дата, подпись)

Оценка отлично  
(прописью)

Красноярск 2018 г.

## Содержание

ВВЕДЕНИЕ .....	3
Глава 1. Теоретические основы развития алгоритмической культуры в процессе обучения информатики .....	6
1.1 Сущность понятия «алгоритмическая культура», его составляющие ..	6
1.2 Структура и содержание алгоритмической линии в школьном курсе информатики .....	13
1.3 Робототехника, как инструмент развития алгоритмической культуры учащихся.....	20
Выводы по 1 главе .....	26
Глава 2. Факультативный курс «Введение в робототехнику на базе LEGO Mindstorms EV3» .....	28
2.1 Технологические возможности конструктора «LEGO Mindstorms EV3» .....	28
2.2 Результативно-целевая модель курса .....	39
2.3 Содержание факультативного курса "Введение в робототехнику на базе LEGO Mindstorms EV3 " .....	44
ЗАКЛЮЧЕНИЕ .....	47
СПИСОК ИСПОЛЬЗОВАННЫХ ИСТОЧНИКОВ .....	49
ПРИЛОЖЕНИЕ А. ....	53
ПРИЛОЖЕНИЕ Б.....	70

## **ВВЕДЕНИЕ**

В век информатизации и автоматизации современные информационные технологии оказывают большое влияние на повседневную жизнь современного человека. Роль подготовки молодежи в области информатики и информационных технологий продолжает оставаться очень высокой. Информационная компонента остается ведущей составляющей технологической подготовки человека в какой бы сфере деятельности ему не пришлось работать в будущем.

Начальный уровень информационной грамотности школьника для его успешной работы в технической системе «человек-компьютер», а также успешного взаимодействия в неформальных «безмашинных» системах «учитель-ученик», «ученик-ученик» и др., обеспечивается в процессе развития алгоритмической культуры, которая создает операционное наполнение, способное обслуживать деятельность школьника в рамках учебных дисциплин, в том числе за пределами «компьютерной» обстановки [24]. Тем самым, развитие алгоритмической культуры учащихся является одной из актуальных задач школьного образования.

Формирование алгоритмической культуры начинается с самых ранних лет жизни человека. Ее дальнейшее развитие происходит при изучении всех школьных дисциплин. Однако, ведущая роль принадлежит математике и информатике, где развитие алгоритмической культуры определяется как одна из нормативных целей обучения. С точки зрения информатики, алгоритмическая культура составляет основу компьютерной грамотности и предполагает создание у учащихся полной ориентировочной основы алгоритмизации: понимание сущности и свойств алгоритма; представление о возможности автоматизировать область деятельности человека, имеющую алгоритм этой деятельности; умение описать алгоритм с помощью

определенных средств и методов; знание основных типов алгоритмических процессов и т.д.

В федеральном государственном образовательном стандарте произошло усиление и уточнение алгоритмической линии. Раздел «Алгоритмизация и элементы программирования», в процессе освоение которого развитие алгоритмической культуры происходит в большей степени, занимает центральное место в базовом курсе информатики. Однако, данный раздел информатики, характеризуется заметными трудностями для школьников, как с определением понятия алгоритма и построением алгоритмических конструкций, так и при работе с формальными языками программирования и их правильном применении. В следствии с этим, появляется потребность в изменении содержания, методов и средств, используемых при преподавании данной темы. Это становится возможным благодаря включению в учебную программу дисциплин близко связанных с информатикой и программированием.

Тем самым, внедрение в образовательную среду раздела Робототехники, предлагаемого ФГОС в качестве дополнительного, приобретает все большую значимость и актуальность в настоящее время.

Видение данной проблемы и определило выбор темы выпускной квалификационной работы: «Развитие алгоритмической культуры в процессе обучения Робототехнике»

Объект исследования: процесс развития алгоритмической культуры в школьном курсе информатики.

Предмет исследования: факультативный курс по робототехнике, как средство развития алгоритмической культуры.

Цель исследования: теоретически обосновать и разработать факультативный курс, направленный на развитие алгоритмической культуры учащихся.

Задачи исследования:

1. Рассмотреть различные подходы к содержанию понятия «алгоритмической культуры» в современном обществе;
2. Проанализировать содержание алгоритмической линии школьного курса информатики с учетом требований ФГОС ;
3. Рассмотреть робототехнику, как инструмент развития алгоритмической культуры;
4. Разработать факультативный курс «введение в робототехнику» направленный на развитие алгоритмической культуры.

Методы исследования:

1. Анализ и систематизация научно-методической литературы;
2. Анализ нормативных документов;
3. Синтез и обобщение собранного материала по теме исследования;
4. Проведение аналогии между робототехникой и информатикой;

# **Глава 1. Теоретические основы развития алгоритмической культуры в процессе обучения информатики**

## **1.1 Сущность понятия «алгоритмическая культура», его составляющие**

Понятие "культура" является сложным и многогранным, в различных областях человеческой деятельности оно имеет огромное количество значений. Как отмечается в философском энциклопедическом словаре "В широком смысле культура есть совокупность проявлений жизни, достижений и творчества народа или группы народов" [7]. Применительно к различным сферам жизни возможно рассмотрение и соответствующих культур.

В образовании, как одной из сфер жизни общества, понятие "культура" используется достаточно широко. Так, В.С. Леднев, характеризуя черты всесторонне развитой личности, выделяет такие основные стороны культуры личности, как познавательная культура, нравственная культура, трудовая культура, эстетическая культура, коммуникативная культура, физическая культура.

В начале 70-х годов 20-ого столетия в педагогическую теорию и практику М. П. Лапчиком было введено новое понятие – «Алгоритмическая культура учащихся», которая, в общем понимании, является совокупностью специфических представлений, умений и навыков, связанных с овладением общими компонентами алгоритмизации.

Существуют и другие подходы к определению данного понятия. А.А. Наумов определяет алгоритмическую культуру как специфическую подсистему культуры, которая прямо и непосредственно связана с социально-информационной деятельностью людей, информационной

культурой, культурой мышления. Она характеризует уровень решения и оценки разнообразных задач (от глобальных до частных) как обществом, так и конкретным человеком [25]. С точки зрения О.Н. Родионовой, алгоритмическая культура является частью общей культуры личности, представляет собой системное и динамическое образование, характеризующееся определенным уровнем развития алгоритмического мышления, осознанием общих компонентов алгоритмизации и проявляющееся в разнообразных формах алгоритмической деятельности, побуждаемой потребностно-мотивационной сферой [14].

Представления учащихся об алгоритме, алгоритмических процессах, а также методах их описания и представления вырабатываются при изучении многих дисциплин, изучаемых в школе, и особенно математики. Но с появлением и распространением электронно-вычислительных машин, алгоритмические представления, умения и навыки начали приобретать самостоятельное значение, и со временем были определены как новый элемент общей культуры современного человека и целенаправленный компонент общего школьного образования, получив название «алгоритмическая культура» учащихся.

При введении основ информатики и вычислительной техники в школьный курс, формирование алгоритмической культуры определялось, как одна из нормативных целей обучения данному предмету [25]. Наполнение этого понятия в тот период связывалось с изучением учащимися основ алгоритмизации и программирования, что прежде всего объяснялось существовавшим уровнем компьютерной техники и программного обеспечения. Современный же этап развития школьного курса информатики требует уточнения перечня и содержания компонентов, характеризующих понятие "алгоритмическая культура".

Существует несколько подходов к определению содержания и основных компонентов алгоритмической культуры.

Впервые основные компоненты и содержание данного понятия были раскрыты М. П. Лапчиком в собственном диссертационном исследовании:

1. Понятие алгоритма и его свойства. Понятие алгоритма является основным понятием и основным компонентом алгоритмической культуры. В обучении алгоритмизации нет необходимости использовать строгое понятия алгоритма, достаточно его толкования на понятном для себя уровне. Из свойств алгоритма, наиболее существенными являются: понятность, точность(однозначность), дискретность, конечность и массовость.

2. Понятие языка описания алгоритмов. Язык описания алгоритма представляется как средство выражения алгоритма. Выбор языка определяется областью применения алгоритма. Первооснова алгоритмизации- выполнение требования четко придерживаться границ языковых возможностей в общении с тем или иным исполнителем алгоритма.

3. Уровень формализации описания. Понятие уровня формализации описания неразрывно связано с понятием языка. Если описание составлено для автоматизированного устройства, то язык, используемый при этом, имеет строгую систему формальных правил, образующих синтаксис языка. А Сам язык становится формализованным. А если алгоритм адресуется человеку, в этом случае окончательный вариант алгоритмизации может иметь неформальное представление. Важно, чтобы алгоритм был понятен исполнителю, т.е. имел команды, входящие в систему команд данного исполнителя. Таким образом, уровни формализации представления алгоритмов, применяемые на практике, варьируются в достаточно широком диапазоне.

4. Принцип дискретности (пошаговости) описания. Построение алгоритма подразумевает выделение точной целенаправленной последовательности допустимых элементарных действий, приводящих к требуемому результату. Именно множество этих действий формирует детерминированную дискретную структуру описания алгоритма, сообщающую ему ясность и четкость. В разных языках эти отдельные

действия алгоритма представляются разными средствами. В словесном описании алгоритма – это отдельные предложения, в языке блок-схем – это отдельные блоки, в объектном языке ЭВМ – это отдельные команды, в языках программирования высокого уровня – операторы.

5. Принцип «блочности». Данный принцип подразумевает умение разделить сложную задачу на более простые части. Именно таким методом приходится пользоваться, когда задача оказывается достаточно сложной, чтобы сразу описать алгоритм ее решения на нужном языке. В данном случае задача делится на блоки, которым придается собственное значение, и после того, как была составлена первоначальной схемы, которая связывает части задачи, проводится работа по конкретизации (уточнению) этих отдельных блоков. Любой из них может быть детализирован по принципу, описанному выше. При конечном составлении алгоритма из блоков возможны два различных варианта. В первом случае, детальное представление блока располагается в определенное место алгоритма, а сам блок как бы «растворяется» в нем. Во втором случае, само содержимое блоков не встраивается в алгоритм, а в его определенных местах располагаются ссылки, указывающие на размещение отдельных блоков; Окончательный алгоритм будет состоять из главного алгоритма и всех его отдельных блоков (вспомогательных алгоритмов).

6. Принцип ветвления. Благодаря данному принципу в алгоритмических описаниях, путем наличия специальных средств, реализуются логические ситуации, требующие принятия решения исходя от заданных начальных условий. При реализации принципа ветвления, нужно осознавать следующее:

а) описанием предусматриваются все допустимые варианты начальных данных и для всякой их комбинации, описание должно быть результативным;

б) для определенных значений исходных данных исполнение алгоритма всегда проходит только по одному из возможных путей, определяемому конкретными условиями.

7. Принцип цикличности. Подразумевает возможность неоднократного повторения одних и тех же фрагментов описаний при различных значениях входных данных. Данный принцип организован с помощью применения логических средств языка, а также специальных средств организации повторяющихся алгоритмов (например, циклические операторы в языках программирования высокого уровня). Важным компонентом алгоритмической культуры человека является понимание общей схемы работы циклического процесса, а также умение выделять эту самую повторяющуюся часть цикла, при построении алгоритмов.

8. Выполнение (обоснование) алгоритма. Наряду с предыдущими принципами, выполнение алгоритма, заключающаяся умении воспринимать и исполнять разрабатываемые фрагменты описания алгоритма отвлеченно от планируемых результатов, также является важным компонентом алгоритмической грамотности

9. Организация данных. Информация или начальные данные, подлежащие обработке, являются основой для алгоритма. Организация данных подразумевает фиксирование результатов работы алгоритма (как промежуточных, так и окончательных) [6].

Математиком Т. И. Алферьевой, были выделены компоненты, из которых должна складываться алгоритмическая культура в процессе изучения математики [32]:

- понимание сущности алгоритма и его свойств;
- понимание сущности языка, как средства для записи алгоритма;
- владение приёмами и средствами для записи алгоритмов;
- понимание алгоритмического характера методов математики и их приложений;
- владение алгоритмами школьного курса математики;
- понимание элементарных основ программирования на компьютере.

Наиболее актуальные компоненты алгоритмической культуры обозначил А.А. Наумов [25]:

- мотивационно-ценностный, призванный обеспечить готовность к действию на основе понимания алгоритмизации поставленной задачи;
- когнитивный, отражающий содержание алгоритмической подготовки школьников;
- технологический – умение использовать алгоритмизацию для решения задач, развивать интерес к алгоритмической деятельности и информационным технологиям;
- коммуникативный, предполагающий конструктивное ведение диалога («человек – человек», «человек – компьютер»), обмен информацией.

Однако, все-таки, компоненты алгоритмической культуры, сформулированные М.П. Лапчиком, являются наиболее подробными и полными и могут быть положены в основу оценочной модели уровня сформированности алгоритмической культуры.

Анализируя исследования разных авторов по данному направлению можно сделать вывод, что алгоритмическая культура является важным компонентом общей культуры человека. Она направлена на формирование представлений, умений и навыков, обеспечивающих начальный уровень информационной грамотности школьника для его успешного взаимодействия в системе «человек-компьютер», а также, успешного взаимодействия в неформальных системах «ученик-ученик», «учитель-ученик» и др. Благодаря сформированной алгоритмической культуре человек способен спланировать свои действия; спрогнозировать результаты своей деятельности; находить оптимальное решение, действуя в ограниченных условиях.

Формирование алгоритмической культуры начинается с самых ранних лет жизни человека. Ее дальнейшее развитие происходит при изучении многих школьных дисциплин. Однако, ведущая роль принадлежит

математике и информатике, где развитие алгоритмической культуры определяется как одна из нормативных целей обучения. С точки зрения информатики, алгоритмическая культура составляет основу компьютерной грамотности и предполагает создание у учащихся полной ориентировочной основы алгоритмизации: понимание сущности и свойств алгоритма; представление о возможности автоматизировать область деятельности человека, имеющую алгоритм этой деятельности; умение описать алгоритм с помощью определенных средств и методов; знание основных типов алгоритмических процессов и т.д., что несомненно является необходимостью в нашей современной жизни.

## **1.2 Структура и содержание алгоритмической линии в школьном курсе информатики**

Базовый курс информатики (7-9 классы), обеспечивает обязательный общеобразовательный минимум подготовки школьников по данному предмету. Он направлен на овладение учащимися методами и средствами информационной технологии решения задач, формирование навыков сознательного и рационального использования компьютера в своей учебной, а затем профессиональной деятельности. Изучение базового курса формирует представления об общности процессов получения, преобразования, передачи и хранения информации в живой природе, обществе, технике.

Содержание базового курса отражает важнейшие аспекты общеобразовательной значимости информатики, как предмета:

- мировоззренческий аспект, связанный с формированием представлений о системно-информационном подходе к анализу окружающего мира, о роли информации в управлении, специфике самоуправляемых систем, общих закономерностях информационных процессов в системах различной природы:

- “пользовательский” аспект, связанный с формированием компьютерной грамотности, подготовкой школьников к практической деятельности в условиях широкого использования информационных технологий;

- алгоритмический (программистский) аспект, связанный, в большей мере, с развитием алгоритмической культуры школьников.

В настоящее время, именно последнему (алгоритмический) аспекту уделяется особое внимание.

Актуальность и необходимость развития алгоритмической культуры в школьном курсе информатики находит свое отражение в ФГОС. Согласно требованиям, предъявляемым федеральным стандартом, предметные результаты освоения образовательной области «Математика и информатика» должны отражать: «формирование информационной и алгоритмической культуры; формирование представления об алгоритме; развитие алгоритмического мышления, необходимого для профессиональной деятельности в современном обществе; развитие умений составить и записать алгоритм для конкретного исполнителя; формирование знаний об алгоритмических конструкциях, логических значениях и операциях; знакомство с одним из языков программирования и основными алгоритмическими структурами (линейной, условной и циклической)» [1].

В примерной программе по информатике разделу «Алгоритмы и элементы программирования» отведено одно из центральных мест. По сравнению с примерной программой государственного стандарта (ГОС 2004 г.), в примерной программе, соответствующей федеральному стандарту (ФГОС 2011г.) произошло уточнение и усиление алгоритмической линии. В данном разделе выделены и подробно раскрыты пять содержательных элементов (глав).

Обязательный минимум содержания учебного материала, который должен быть усвоен учащимися полностью, составляют следующие главы [2]:

1. «Исполнители и алгоритмы. Управление исполнителями»;

Содержание этой главы включает в себя сведения о различных исполнителях алгоритмов; обстановках, в которых они обитают; системах команд исполнителей (СКИ). Раскрывается понятие алгоритм, способы описания алгоритма (словесное, блок-схемы, формальное), происходит знакомство с формальными языками для записи алгоритмов и системами программирования. Рассматривается компьютер, как устройство-

исполнитель некоторой заранее составленной программы, принципы программного управления компьютером.

## 2. «Алгоритмические конструкции»;

В данной главе изучаются основные алгоритмические конструкции:

- Конструкция «следование» (линейный алгоритм, ограниченность линейных алгоритмов);
- Конструкция «ветвление» (полная и неполная формы, выполнение и невыполнение условия, простые и составные условия и их запись.);
- Конструкция «повторения» (циклы с заданным числом повторений, с условием выполнения, с переменного цикла).

Запись алгоритмических конструкций на определенном формальном языке.

## 3. «Разработка алгоритмов и программ»;

В содержании этой главы рассматриваются: Оператор присваивания; переменные их типы (целые, вещественные), допустимые имена и значение, табличные величины (одномерные массивы).

Содержатся примеры задач обработки данных, алгоритмы их решения и реализация в определенной среде программирования:

- нахождение минимального и максимального числа из двух, трех, четырех данных чисел;
- нахождение всех корней заданного квадратного уравнения;
- заполнение числового массива в соответствии с формулой или путем ввода чисел; нахождение суммы элементов массива; нахождение минимального (максимального) элемента массива.

Рассматривается составление алгоритмов и программ по управлению исполнителями КУМИРа (Робот, Черепашка, Чертежник и др).

Затрагиваются этапы разработки программ (составление требований, выбор алгоритма и его реализация, отладка программы, тестирование).

Простейшие приемы диалоговой отладки программ (выбор точки останова, пошаговое выполнение, просмотр значений величин, отладочный вывод).

Происходит знакомство с документированием программ.

#### 4. «Анализ Алгоритмов».

В этой главе рассматриваются сложности вычислений: количество выполненных операций, размер используемой памяти; зависимость от размера исходных данных; примеры коротких программ, выполняющих обработку большого (небольшого) объема данных.

Определение возможных результатов работы алгоритма при данном множестве входных данных; определение возможных входных данных, приводящих к данному результату. Примеры описания объектов и процессов с помощью набора числовых характеристик, а также зависимостей между этими характеристиками, выражаемыми с помощью формул.

Данные содержательные элементы входят в базовую часть учебного курса информатики и относятся к предметным результатам, которым учащиеся должны научиться.

Согласно примерной основной образовательной программы основного общего образования, в результате освоения раздела «Алгоритмы и элементы программирования» выпускник научится:

- составлять алгоритмы для решения задач;
- представлять алгоритм решения задачи разными способами (словесным, графическим, в виде блок-схемы, с помощью языков программирования), определять наиболее оптимальный способ для решения конкретной задачи.
- видеть результат выполнения заданного алгоритма или его фрагмента;
- знать термины «исполнитель», «алгоритм», «программа», и понимать разницу между употреблением этих терминов в обыденной речи и в информатике;

- выполнять «вручную» простые алгоритмы управления исполнителями и анализа числовых и текстовых данных, записанные на конкретном языке программирования с использованием основных управляющих конструкций.

- составлять несложные алгоритмы с использованием основных управляющих конструкций и записывать их в виде программ на выбранном языке программирования; выполнять эти программы на компьютере;

- использовать величины различных типов, табличные величины, а также выражения, составленные из этих величин; использовать оператор присваивания;

- анализировать предложенный алгоритм.

- использовать логические значения, операции и выражения с ними;

- записывать на выбранном языке программирования арифметические и логические выражения и вычислять их значения.

Помимо элементов содержания, составляющих базовую часть раздела «Алгоритмы и элементы программирования», в примерной программе предусмотрены дополнительные содержательные элементы, относящиеся к результатам, которым учащиеся «получают возможность научиться», и изучение которых возможно за счет вариативной части курса информатики.

Отдельно стоит обратить внимание на целую главу раздела – робототехника, содержание которой также очень подробно раскрыто авторами примерной основной образовательной программы, но пока не является обязательным для изучения [2].

В содержании данной главы рассматривается робототехника, как наука о разработке и использовании автоматизированных технических систем, автономные роботы и автоматизированные комплексы, примеры роботизированных систем.

Учащиеся знакомятся с понятиями: микроконтроллер, сигнал. обратная связь: получение сигналов от цифровых датчиков (касания, расстояния, света, звука и др.

Предусмотрено конструирование и моделирование роботов, знакомство с ручным и программным управлением роботами.

Приводятся примеры учебной среды разработки программ управления движущимися роботами, алгоритмы управления движущимися роботами, реализация алгоритмов "движение до препятствия", "следование вдоль линии" и т.п. Анализ алгоритмов действий роботов, испытание механизма робота, отладка программы управления роботом, влияние ошибок измерений и вычислений на выполнение алгоритмов управления роботом.

Включая дополнительные элементы содержания в школьный курс, учащиеся получают возможность [2]:

- познакомиться с использованием в программах строковых величин и с операциями со строковыми величинами;
- создавать программы для решения задач, возникающих в процессе учебы и вне ее;
- познакомиться с задачами обработки данных и алгоритмами их решения;
- познакомиться с понятием «управление», с примерами того, как компьютер управляет различными системами (роботы, летательные и космические аппараты, станки, оросительные системы, движущиеся модели и др.);
- познакомиться с учебной средой составления программ управления автономными роботами и разобрать примеры алгоритмов управления, разработанными в этой среде.

В данной главе были проанализированы государственный образовательный стандарт ООО (ГОС 2004 г.), федеральный государственный образовательный стандарт ООО (ФГОС 2011 г.) и

примерные основные образовательные программы, соответствующие стандартам ГОС и ФГОС, с целью выявления содержания алгоритмической линии и соответствия предметным результатам, предъявляемым к выпускнику. Исходя из анализа, содержательной линии «Алгоритмы и элементы программирования» (ФГОС) видно, что, в настоящее время, алгоритмическая линия является ведущей. Подробно раскрыто ее содержание и уточнены требования к предметным результатам учащихся, предъявляемым федеральным стандартом.

В примерной программе, соответствующей ФГОС, появляется новый раздел- робототехника, который пока является не обязательным для изучения и относится к дополнительным предметным результатам. Нетрудно предположить, что вскоре, пройдя через имеющиеся трудности, «Робототехника» займет одно из ведущих мест в базовой части раздела «Алгоритмы и элементы программирования».

### **1.3 Робототехника, как инструмент развития алгоритмической культуры учащихся**

Современный этап развития общества характеризуется внедрением информационных технологий во все сферы деятельности человека. Компьютеры и ЭВМ уже стали значительной частью нашей жизни. Любая человеческая деятельность, связанная с процессами управления в различных системах, сводится к реализации определенных алгоритмов. Поэтому, в настоящее время, формирование алгоритмической культуры обучающихся является одной из важных образовательных задач, решением которой занимается информатика.

«Алгоритмы и элементы программирования» - ведущий раздел курса информатики в основной школе, так как способность составлять и реализовывать алгоритмы занимает одно из центральных мест при обработке информации и решении поставленных задач. Данный раздел традиционно характеризуется заметными трудностями для школьников, как с определением понятия алгоритма и построением алгоритмических конструкций, так и при работе с формальными языками программирования и их правильном применении. В результате, учитель сталкивается с проблемой мотивации к изучению программирования и самостоятельной работе [19].

С развитием визуального объектно-ориентированного программирования, изучение формальных языков постепенно теряет свою значимость. Написание программы для исполнителя становится более удобным и простым. Программист работает с визуальными объектами, может изменять их внешний вид и, что не менее важно, сразу же наблюдать результат выполненных изменений на реальном исполнителе. Однако определенные фундаментальные знания и понимание основ алгоритмизации

и программирования остаются необходимыми для правильной работы программы.

Исходя из этого, появляется потребность в изменении содержания, методов и средств, используемых при преподавании данной темы. Это становится возможным благодаря включению в учебную программу дисциплин близко связанных с информатикой и программированием [17].

В качестве таких методов и средств ФГОС предлагает включение в содержание раздела алгоритмизации образовательной робототехники.

В период автоматизации и информатизации работы активно применяются на производстве и в быту. Программное управление системами робототехники является одним из активно развивающихся направлений в программировании, поэтому внедрение робототехники в учебный процесс, не только на профильном, но и базовом уровне имеет значимость и актуальность.

Образовательная робототехника – это направление в обучении, которое объединяет знания физики, информационного моделирования, кибернетики и программирования, и позволяет наглядно представить работу алгоритмических конструкций. Использование её методик и средств обучения способствует существенному развитию алгоритмического мышления, способностей и алгоритмической культуры в общем [9].

На занятиях робототехникой, используются конструкторы - это набор сопрягаемых деталей и электронных блоков для создания программируемого робота. Конструкторы предназначены, как для индивидуальной, так и для групповой работы учащихся.

На сегодняшний день существует большое количество разнообразных учебных конструкторов для обучения школьников навыкам программирования робототехнических платформ. Обучающие платформы можно разделить [8]:

- по сложности и количеству элементов для сборки;

- по типу программного обеспечения контроллера, управляющего роботом;
- по возрасту пользователей;

Наиболее популярными платформами, используемыми во всем мире, являются конструкторы фирм: LEGO, RoboRobo, Robotis, «КиберТех».

Занятия по робототехнике значительно отличаются от традиционных занятий по любому другому предмету. С одной стороны, важно добиваться того, чтобы созданные модели работали, и отвечали соответствующим требованиям, но при этом учащиеся получают возможность проявлять творческий подход при решении поставленной задачи и учиться на собственном опыте, а любая ошибка должна поощряться, ибо только через ошибку можно прийти к результату. Основным принципом обучения «шаг за шагом» обеспечивает учащимся возможность работать в собственном темпе, а применение робототехники становится реальным на любом этапе обучения, будь это начальная школа, средние или старшие классы [21].

Программирование робота на определенные действия в графической среде, не является нерешаемой задачей для учащихся, а наоборот оказывает существенное влияние на современное понимание идей программирования и позволяет открыть на эту область в «новом свете».

Конструкторы позволяют привлечь и удерживать внимание учащихся, организовать учебную деятельность, применяя различные предметы, и проводить интегрированные занятия, позволяя решать задачи по физики, математики и т.д. [20].

При работе с конструктором формируется умение определять способы решения учебной задачи на основе заданных алгоритмов, объединять известные алгоритмы деятельности в ситуациях, которые не предполагают стандартного применения одного алгоритма, отказываться от образца деятельности и искать оригинальные решения [16].

Эффективность обучения основам робототехники напрямую зависит от организации занятий, проводимых с применением определенных методов. Наиболее эффективными методами, которые используются в процессе реализации курса робототехники, являются, метод портфолио, метод взаимообучения, модульный метод, метод проблемного обучения, но главным методом является метод проектов [15].

Метод портфолио предназначен для того, чтобы систематизировать накапливаемый опыт и знания. В процессе обучения робототехники, создание портфолио достаточно важно, так как во время его разработки учащийся осмысливает свои достижения, осознает возможности и формирует собственное отношение к полученным результатам.

Метод взаимообучения представляет собой коллективный способ обучения. В его основе лежит общение обучающихся и обучаемых. Способы обучения основываются на применении различных видов общения. На занятиях робототехники метод взаимообучения реализуется учениками самостоятельно, иногда даже без участия учителя. Разобравшись в решении какой-либо конструкторской задачи или задачи программирования роботов, учащиеся с удовольствием помогают тем, кто испытывает проблемы при решении подобных задач. Таким образом, вполне нормально, если сложится ситуация, в которой ученики обучают и самого учителя, что положительно влияет на их самооценку и способствует укреплению отношений с учителем.

Сущность метода модульного обучения состоит в том, что обучающийся получает возможность работать более самостоятельно, самопланируя и самоорганизовывая свою деятельность в соответствии с предложенной ему индивидуальной программой, которая включает в себя целевой план действий, банк информации и методическое руководство по достижению поставленных дидактических целей. В основе индивидуальных программ, лежат самостоятельные организационно-методические блоки-модули.

Содержание курса по робототехнике целесообразно разбить на следующие модули:

- основы конструирования;
- программирование;
- решение прикладных задач.

Формирование структуры модулей может носить циклический характер, т.е. тематика модулей повторяется через некоторые промежутки времени. В темах конструирования и программирования одного временного периода удобно рассматривать задачи единых проектов, чтобы у учащихся сформировалось целостное представление о реализации той или иной модели робота.

Метод проблемного обучения состоит в создании для учащихся проблемных ситуаций и обучению умения находить оптимальное решение для выхода из данных ситуация. В процессе использования данного метода, учебный материала должен быть представлен как цепь проблемных ситуаций, учитель оказывает ученикам лишь необходимую помощь в решении проблем, проверяет правильность решений и осуществляет общее руководство процессом систематизации и закрепления приобретенных знаний. Данный метод способствует активизации самостоятельной деятельности учащихся, направленной на разрешение проблемной ситуации. В результате этого происходит творческое овладение знаниями, навыками, умениями и развитие мыслительных способностей. Практически любую задачу, которая решается в процессе конструирования и программирования роботов, можно представить в качестве проблемной ситуации. Активизируя творческое и критическое мышление, учащиеся способны оптимизировать собственное решение задачи.

Под метод проектов понимают способ достижения дидактической цели через подробную разработку проблемы, в результате которого получится определенный, вполне реальный, осязаемым, результат, оформленным тем

или иным образом. Использование данного метода при занятиях робототехникой позволяет развивать познавательные и творческие навыки учащихся. Самостоятельная работа над проектом дисциплинирует ребят, заставляет мыслить критически.

Основные этапы разработки проекта:

- Обозначение темы проекта.
- Цель и задачи представляемого проекта.
- Гипотеза.
- Разработка механизма на основе конструктора.
- Составление программы для работы механизма.
- Тестирование модели, устранение дефектов и неисправностей.

При разработке и отладке групповых проектов, учащиеся имеют возможность самостоятельно выбирать этап, над которым будут работать (модульное обучение), а также, в процессе работы, делятся опытом друг с другом (метод взаимообучения), что очень эффективно влияет на развитие познавательных, творческих навыков, а также самостоятельность школьников.

На практике, в процессе реализации курса по робототехнике, наиболее продуктивным является применение совокупности нескольких методов обучения из вышеописанных [26].

Подводя итог, можно сделать вывод, что введение робототехники в образовательный процесс является целесообразным. При правильно организованном обучении, она способствует повышению уровня мотивации учащихся к изучению программирования и более легкому пониманию принципов действия алгоритмических конструкций, за счет наглядного представления их работы, а также пониманию использования основ алгоритмизации и программирования для решения проблем реального мира. В учебной деятельности полученные знания смогут помочь учащимся

сформировать представление об особенностях составления программ управления, автоматизации механизмов, моделировании работы систем, тем самым способствовать дальнейшему самообразованию в области алгоритмизации и программирования и развитию алгоритмической культуры.

Исходя из возможностей и эффективности использования робототехники, был разработан факультативный курс, направленный на развитие алгоритмической культуры учащихся (Приложение А) и разработана оценочная модель уровня сформированности основных компонентов алгоритмической культуры (Приложение Б).

## **Выводы по 1 главе**

Проанализировав библиографические источники по теме исследования, можно сделать выводы:

Алгоритмическая культура является важным компонент общей культуры человека и направлена на формирование представлений, умений и навыков, обеспечивающих начальный уровень информационной грамотности школьника. На формирование алгоритмической культуры направлены большинство школьных дисциплин, однако, ведущая роль принадлежит математике и информатике. С точки зрения информатики, алгоритмическая культура составляет основу компьютерной грамотности и предполагает создание у учащихся полной ориентировочной основы алгоритмизации.

Исходя из анализа, содержательной линии «Алгоритмы и элементы программирования» (ФГОС) видно, что, в настоящее время, алгоритмическая линия является ведущей. Подробно раскрыто ее содержание и уточнены требования к предметным результатам учащихся, предъявляемым

федеральным стандартом. В примерной программе, соответствующей ФГОС, появляется новый раздел- робототехника, который пока является не обязательным для изучения и относится к дополнительным предметным результатам.

При правильно организованном обучении, введение робототехники в образовательный процесс является целесообразным, она способствует более легкому пониманию принципов действия алгоритмических конструкций и повышению уровня мотивации учащихся к изучению программирования. Знания, полученные в процессе занятий по робототехнике, смогут помочь учащимся сформировать представление об особенностях составления программ управления и автоматизации механизмов, тем самым будут способствовать дальнейшему самообразованию в области алгоритмизации и программирования и развитию алгоритмической культуры.

## **Глава 2. Факультативный курс «Введение в робототехнику на базе LEGO Mindstorms EV3»**

### **2.1 Технологические возможности конструктора «LEGO Mindstorms EV3»**

При всем разнообразии образовательных конструкторов, мы остановились на конструкторе популярной компании LEGO, которая уже более 20 лет занимается разработкой наборов для конструирования и является одним из лидеров данной отрасли в мировом рейтинге.

Для изучения алгоритмизации с использованием основ робототехники выбрана следующая материально-техническая база:

1. Робототехнический набор для конструирования Lego Mindstorms EV3;
2. Программное обеспечение Lego Mindstorms Education EV3.

Lego Mindstorms Education Evolution 3 – это набор для конструирования программируемой робототехники, дающий возможность создания и управления собственными роботами LEGO. Набор совместим с деталями Lego Technic. Это означает, что платформу можно использовать для создания самых разнообразных и даже невероятных конструкций, от простых «машинок» и «роборук», до сложных конвейеров или даже «решателей» кубика Рубика. Фактически любой набор Lego Technic может стать источником деталей для проектов, также никаких проблем не будет с заменой пострадавших запасных частей [4].

Конструктор разработан для двух целевых аудиторий: для домашнего пользования (дети) – Home Edition EV3 и для образовательных учреждений (ученики и преподаватели) – Education EV3.

Версия Home Edition EV3 выпускается в картонной коробке, содержит блок EV3, два больших и один средний сервомоторы, датчики: инфракрасный (с маячком), цвета и касания, детали для конструирования.

В комплект образовательной версии Lego Mindstorms EV3 входят [3]:

- 1 модуль EV3;
- аккумуляторная батарея;
- 2 больших мотора
- 1 средний мотор;
- 2 датчика касания;
- 1 датчик цвета;
- 1 ультразвуковой датчик;
- 1 гироскоп;
- 540 строительных элементов;
- набор RJ12 кабелей;

1) Модуль EV3 (рис. 1) – программируемый интеллектуальный модуль, контролирующий работу моторов и датчиков. Модуль имеет возможность поддержки беспроводных протоколов связи Wi-Fi и Bluetooth.



Рис. 1. Модуль EV3

- Модуль EV3 обладает следующими характеристиками:

- процессор — ARM9: тактовая частота – 300 МГц;
- оперативная память — 64 мегабайт;
- FLASH память — 16 мегабайт;
- разрешение экрана модуля – 178x128/ черно-белый;
- слот расширения SD – максимально 32 Гб;
- USB 2.0 с поддержкой подключения Wi-Fi;
- Bluetooth 2.1;
- автономное питание от шести батарей типа АА, либо с использованием аккумулятора постоянного тока EV3 2050 мАч;
- операционная система Linux.

Модуль EV3 имеет девять основных портов. На одной стороне микрокомпьютера порты вывода (A-D) для подключения моторов к модулю, а также мини-USB-порт для связи модуля с компьютером, на другой стороне порты ввода (1-4) для подключения датчиков к модулю.

Дисплей на верхней части микрокомпьютера имеет возможность показывать тексты, рисунки, выводить данные с датчиков, а также результаты математических вычислений, а динамик, расположенный сбоку, может проигрывать как просто звуки, так и записанные заранее звуковые файлы.

Также модуль имеет слот USB-порта, с помощью которого можно подключать USB-адаптер Wi-Fi, а также использовать его для соединения вместе до четырех модулей EV3 (подключение шлейфом).

Интерфейс модуля содержит четыре основных окна, несущие в себе как простые функции, например, запуск и остановка программы, так и сложные, например, написание самой программы.

1. Запустить последнюю (рис. 2). В этом окне будут отображаться программы, которые запускали последними.



Рис. 2. Окно «Запустить последнюю»

2. Выбор файла (рис. 3). Из этого окна осуществляется доступ и управление файлами в модуле, а также файлы, которые хранятся на SD-карте. Файлы в окне организованы по папкам проектов, в которых, помимо программ, содержатся звуки, изображения, используемые в проекте. Есть возможность перемещать или удалять файлы с помощью навигатора по файлам.



Рис. 3. Окно выбора файла

3. Приложения модуля (рис. 4). Модуль поставляется с пятью заранее установленными приложениями модуля. Также есть возможность создавать собственные приложения в программном обеспечении EV3.



Рис. 4. Окно приложений модуля EV3

4. Настройки (рис. 5). Окно, позволяющее просматривать и корректировать различные общие настройки модуля, такие как:

- громкость звука, исходящего из динамика модуля;
- спящий режим, которое должно пройти до момента, когда модуль перейдет в спящий режим;
- bluetooth (позволяет осуществлять включение/выключение Bluetooth на модуле. Также можно установить конкретные параметры конфиденциальности, настройки AppleiOS, осуществить подключение в другим устройства по Bluetooth);
- Wi-Fi (осуществление включение связи Wi-Fi на модуле, установление соединения с беспроводной сетью);
- Информация о модуле включает в себя сведения о текущих технических характеристиках модуля (аппаратные средства, версия встроенного ПО, версия ОС модуля).



Рис. 5. Экран настройки параметров модуля EV3

2) Два больших серво мотора (рис. 6) созданы для работы с микрокомпьютером EV3 и имеют датчик вращения с точностью измерений до 1 градуса. С помощью датчика вращения мотор можно присоединить к другим моторам, что позволит роботу передвигаться с постоянной скоростью. Датчик вращения может использоваться также и для проведения различных экспериментов для точного считывания данных о расстоянии и скорости.



Рис. 6. Большой мотор комплекта Lego Mindstorms EV3

3) Средний серво мотор (рис. 7) предназначен для задач, когда скорость и быстрота отклика, а также размер робота важнее его грузоподъемности. Также как у большого серво мотора, средний серво мотор имеет точность измерения встроенным датчиком вращения – 1 градус.



Рис. 7. Средний мотор комплекта Lego Mindstorms EV3

4) Датчик света – это цифровой датчик, имеющий возможность определять цвет или яркость света, поступающего в окошко, расположенное в лицевой части датчика (рис. 8).



Рис. 8. Датчик света комплекта Lego Mindstorms EV3

Датчик может работать в трех разных режимах:

- а. Режим «Цвет». Датчик может распознавать семь цветов: черный, синий, зеленый, желтый, красный, белый и коричневый, а также отсутствие цвета.
- б. Режим «Яркость отраженного света». В этом режиме датчик определяет яркость света, отраженного от лампы, излучающий красный свет, используя шкалу от 0, что означает «очень темный», до 100, что означает «очень светлый».
- с. Режим «Яркость внешнего освещения». В этом режиме датчик определяет силу света, входящего в окошко из окружающей среды, используя аналогичную шкалу от 0 до 100.

5) Ультразвуковой датчик – это цифровой датчик, который определяет расстояние до находящегося перед ним объекта (рис. 9).



Рис. 9. Ультразвуковой датчик комплекта Lego Mindstorms EV3

Принцип работы датчика заключается в посыле звуковых волн высокой частоты и измерении времени, за которое звук отразится назад к датчику. Расстояние до объекта может измеряться как в сантиметрах, так и в дюймах.

Если в качестве единицы измерения используются сантиметры, то расстояние обнаружения составляет от 3 до 250 см, с точностью до 1 сантиметра. А если в качестве измерения используются дюймы, то расстояние обнаружения составляет от 1 до 99 дюймов, с точностью до 0,394 дюйма. Если значение датчика составит 255 сантиметров или 100 дюймов, то это будет означать, что датчик не может обнаружить никакой объект перед собой.

6) Датчик касания – это аналоговый датчик, с возможностью определения, когда красная кнопка датчика нажата, а когда отпущена (рис. 10).



Рис. 10. Датчик касания комплекта Lego Mindstorms EV3

7) Гироскопический датчик – это цифровой датчик, обнаруживающий вращательное движение по одной оси (рис. 11).



Рис. 11. Гироскопический датчик комплекта Lego Mindstorms EV3

При вращении гироскопического датчика в направлении стрелки на корпусе, датчик может определить скорость вращения в градусах в секунду. Максимальная скорость, которую датчик может измерить, составляет 440 градусов в секунду. Так же гироскопический датчик может отслеживать угол вращения в градусах, с точностью 3 градуса при повороте на 90 градусов. При подключении гироскопического датчика в модуль EV3 датчик должен быть полностью неподвижен.

Помимо достаточно насыщенной аппаратной базы, Mindstorms EV3 обладает интуитивно понятным для множества платформ (MAC, ПК, мобильные устройства) программным обеспечением.

Среда программирования EV3 разработана компанией National Instruments на основе мощной и дружелюбной в освоении и использовании инженерного программирования LabVIEW [10]. Программное обеспечение EV3 Edu Edition является бесплатным и поставляется со специальным инструментом по обучению *roboteducator*, который включает 48 пошаговых мультимедийных обучающих уроков, созданных для учеников и преподавателей в помощь обучению основам робототехники.

Визуализация алгоритмов в родном ПО LEGO MINDSTORMS Education EV3 находится на высоком уровне (рис.12). Программирование осуществляется путем перетаскивания иконок в рабочее окно с целью формирования команд. Достаточно, буквально за несколько минут, усвоить основные виды взаимодействия логических блоков (условия перехода, цикл и т.д.) и в дальнейшем постепенно наращивать сложность программ. Разумеется, есть и готовые обучающие проекты для десятков разнообразных моделей роботов.

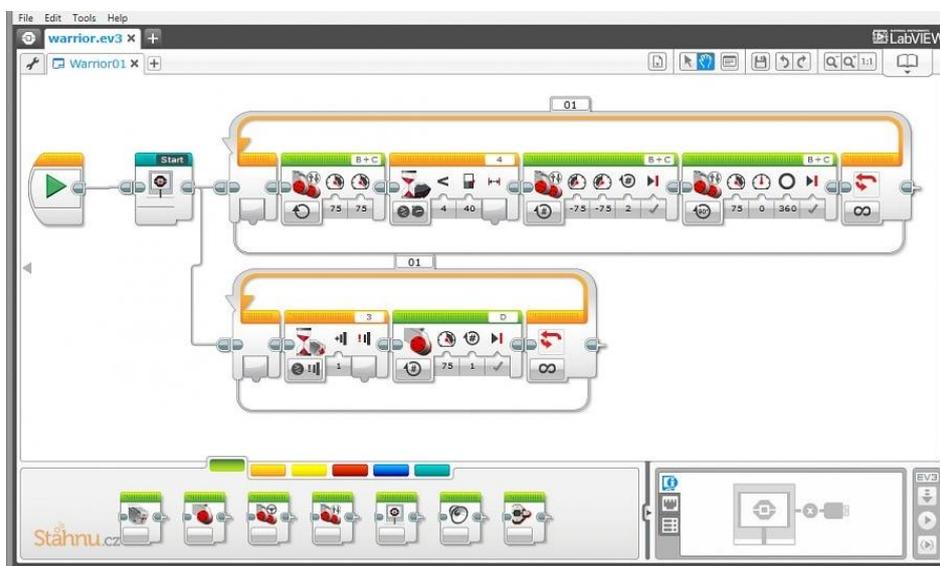


Рис.12. Окно программы

Среда программирования EV3 Edu Edition состоит из следующих основных областей:

- 1) Область программирования или рабочая область – здесь будет располагаться программа;

- 2) Палитры программирования – блоки для программирования:
  1. Действие (зеленая палитра) – вкладка отвечает за действия моторов, экрана, звука, индикатора состояния модуля;
  2. Управление операторами (оранжевая палитра) – на вкладке расположены основные алгоритмические структуры: начало, цикл, переключатель (ветвление), прерывание цикла;
  3. Датчик (желтая палитра) – расположены блоки для управления датчиками;
  4. Операции с данными (красная палитра) – вкладка содержит блоки для взаимодействия с данными;
  5. Дополнения (синяя палитра) – вкладка содержит блоки дополнительных функций: доступ к файлу, обмен сообщениями, подключение Bluetooth, остановка программы;
  6. Мои блоки (бирюзовая палитра) – вкладка может содержать блоки, созданные пользователем.
- 3) Страница аппаратных средств – установка и управление связью с модулей EV3, просмотр состояния моторов, датчиков, загрузка программ;
- 4) Редактор контента – электронная тетрадь, встроенная в программное обеспечение. Здесь можно получить инструкции по сборке робота или задокументировать свой проект, используя текст, изображение или видео;
- 5) Панель инструментов программирования – панель содержит основные инструменты для работы с программой.

Существенным преимуществом MINDSTORMS Education EV3 является его совместимость с пакетами LabVIEW или RobotC, что позволяет программировать на более высоком уровне и использовать набор при занятиях в старших классах.

ПО для настольных компьютеров позволяет вести электронные тетради учеников, благодаря которым преподаватель может из своей версии

приложения оценивать успехи конкретного ученика и наблюдать за его прогрессом. Плюс ко всему можно использовать не только имеющиеся на борту ПО учебные материалы (коих множество), но и с помощью встроенного редактора контента создавать свои собственные [10].

По рекомендациям производителя, MINDSTORMS Education EV3 ориентирован на детей среднего школьного возраста, для которых стоит вопрос выбора будущей профессии. А учебную программу на базе EV3 можно спокойно планировать на всю среднюю школу и дальше.

Используя EV3, каждый из учеников сможет активнее раскрыть те способности, которые были в нем заложены природой, воспитанием и учебным процессом.

Прирожденный математик будет пристально следить за телеметрией датчиков, за тем, как именно фиксируется пройденное роботом расстояние, как записывается угол, на который он отклоняется, и прочее. Будущий программист, само собой, погрузится в программирование робота, разбирая алгоритмы, по которым тот движется. И непременно будет создавать свои, не предусмотренные штатной инструкцией. Ребенок, увлеченный физикой, сможет с помощью робота проводить наглядные эксперименты, благо с датчиками у наборов проблем нет, равно как и у ребенка — с фантазией.

По большому счету использование EV3 можно «привязать» к любому образовательному проекту, связанному с программированием или моделирование, поскольку возможность увидеть работу собственных алгоритмов «в железе» имеет большое значение [24].

В общем, использованием наборов MINDSTORMS EV3 в обучении позволит выделить интересы ребенка и сконцентрировать их на развитие.

## **2.2 Результативно-целевая модель курса**

### **Пояснительная записка**

Программа курса «Введение в робототехнику на базе LegoMindstorms EV3» составлена в соответствии требований Федерального государственного образовательного стандарта основного общего образования, Примерной основной образовательной программы основного общего образования. Содержанию курса соответствует содержанию дополнительного модуля «Робототехника» в главе «Алгоритмы и элементы программирования» примерной программы по информатике.

Настоящий курс предлагает использование образовательных конструкторов MINDSTORMS EV3, как инструмента для обучения школьников конструированию, моделированию и компьютерному управлению на уроках робототехники. Простота в построении модели в сочетании с большими конструктивными возможностями конструктора позволяют детям в конце урока увидеть сделанную своими руками модель, которая выполняет поставленную ими же самими задачу. При построении модели затрагивается множество проблем из разных областей знания - от теории механики до психологии. Использование компьютера, как средства управления моделью; его использование направлено на составление управляющих алгоритмов для собранных моделей. [26].

**Место курса в образовательном процессе:** Программа факультативного курса «Введение в робототехнику на базе LegoMindstorms EV3» предназначена для учащихся 6-9 классов. Программа рассчитана на 34 часа (2 часа резервные).

**Актуальность курса:** Курс направлен на формирование творческой личности, живущей в современном мире, развитие алгоритмической и информационной культуры учащихся. Работа с образовательными конструкторами LEGO позволяет школьникам в форме познавательной игры узнать многие важные идеи и развить необходимые в дальнейшей жизни навыки. Курс ориентирован на изучение основных физических принципов и базовых технических решений, лежащих в основе всех современных конструкций и устройств. В учебной деятельности полученные знания смогут помочь учащимся сформировать представление об особенностях составления программ управления, автоматизации механизмов, моделировании работы систем. Тем самым курс способствует дальнейшему самообразованию в области алгоритмизации и программирования.

### **Цели изучения курса**

#### **Образовательные:**

- Углубление знаний по основным принципам механики и основам алгоритмизации и программирования; ознакомление с основами программирования в компьютерной среде Mindstorms EV3.

#### **Развивающие:**

- Развитие алгоритмического, абстрактного и логического мышления. Развитие умения творчески подходить к решению задачи. Развитие научно-технического и творческого потенциала личности ребенка путем организации его деятельности в процессе интеграции начального инженерно-технического конструирования и основ робототехники.
- Развитие умения довести решение задачи до работающей модели. Развитие умения излагать мысли в четкой логической последовательности, отстаивать свою точку зрения,

анализировать ситуацию и самостоятельно находить ответы на вопросы.

**Воспитательные:**

- формирование творческого подхода к поставленной задаче;
- формирование представления о том, что большинство задач имеют несколько решений;
- формирование целостной картины мира;
- ориентирование на совместный труд.

В процессе освоения курса у учащихся будут формироваться следующие компетенции (таблица 1).

<b>Формируемые компетенции</b>	
<b>Ключевые общеобразовательные (общеучебные умения, способы деятельности; универсальные учебные действия)</b>	Навыки индивидуальной работы в процессе моделирования и описания алгоритма
	Проявление творческих способностей в процессе моделирования и составлении программы для решения поставленной задачи
	Умение анализировать, отбирать и структурировать необходимую информацию для создания конкретной модели
	Умение планировать свою деятельность при создании модели и написании программы для исполнителя
	Адекватная самооценка своей деятельности в процессе создания модели
	Готовность к повышению своего образовательного уровня и продолжению обучения с использованием средств и методов информатики и ИКТ
<b>Предметные</b>	Представления об алгоритме, модели — и их свойствах;

<b>компетенции (предметные умения, способы деятельности)</b>	Представление о возможностях, и инструментах LEGO Mindstorms EV3;
	Умение программировать контролер EV3, сенсорные системы;
	Умение пошагово выполнять несложные алгоритмы управления исполнителями и анализа числовых и текстовых данных;
	Владение алгоритмами и методами решения организационных и технических задач;
	Навыками использования информационных устройств: компьютера, мобильного телефона, конструктора LEGO.

Таблица 1. Формируемые компетенции

**В результате освоения курса обучающиеся должны:**

**знать:**

- правила безопасной работы с робототехническим конструктором LEGO Mindstorms EV3;
- основные компоненты конструктора LEGO Mindstorms EV3;
- конструктивные особенности различных моделей, сооружений и механизмов;
- компьютерную среду Mindstorms Education EV3, включающую в себя графический язык программирования;
- виды подвижных и неподвижных соединений в конструкторе; основные приемы конструирования роботов;
- конструктивные особенности различных роботов;
- как передавать программы;
- как использовать созданные программы;

**уметь:**

- работать с литературой, с журналами, с каталогами, в интернете (изучать и обрабатывать информацию);
- самостоятельно решать технические задачи в процессе конструирования роботов (планирование предстоящих действий, самоконтроль, применять полученные знания, приемы и опыт конструирования с использованием специальных элементов и т.д.);
- осуществлять управление имеющимся формальным исполнителем;
- разрабатывать алгоритм решения практических задач;
- реализовывать блок-схемы, понимать описания алгоритмов в программном обеспечении Lego Mindstorms Education EV3;
- подбирать алгоритмическую конструкцию, соответствующую заданной ситуации;
- составлять линейные, ветвящиеся и циклические алгоритмы управления для исполнителя Lego Mindstorms EV3;
- выполнять тестирование и отладку алгоритма для исполнителя Lego Mindstorms EV3;
- выделять подзадачи: определять и использовать вспомогательные алгоритмы;
- применять полученные знания в практической деятельности;

**Владеть навыками:**

- разработки плана действия для решения практических задач;
- работы с роботом Lego Mindstorms EV3 и средой Mindstorms Education EV3

## **2.3 Содержание факультативного курса "Введение в робототехнику на базе LEGO Mindstorms EV3 "**

Содержание курса разработано в соответствии с содержанием дополнительного модуля «Робототехника» в разделе «Алгоритмы и элементы программирования» и включает 10 тематических разделов:

### **1. Вводное занятие. Презентация программы. ТБ**

В данном тематическом разделе учащиеся знакомятся с предметом «Робототехника», понятием робот, какие бывают роботы. Рассматриваются современные тенденции робототехники, зарубежные и отечественные разработки. Знакомятся с программой курса, проводится инструктаж по технике безопасности на занятиях, правилам внутреннего распорядка и поведение в коллективе. Знакомство с конструктором и правилами работы с ним.

### **2. Конструирование.**

Во втором разделе, учащиеся начинают работу с конструктором. Учащиеся знакомятся со способами крепления деталей и различными принципами конструирования. Выполняют 2 практические работы по сооружению жесткой конструкции - «Высокая башня» и подвижной – механический манипулятор «хваталка».

### **3. Первые модели.**

В данном разделе происходит знакомство с самыми простыми моделями, при создания которых используются моторы и модуль EV3. Учащиеся выполняют практические работы по конструированию тележек, управляют сделанными моделями с помощью своих мобильных устройств. Уже на данном этапе от учащихся требуется умение следовать четкому алгоритму действий для создания определенной модели робота.

#### **4. Подключения EV3.**

Учащиеся подробно знакомятся с подключением электромоторов, датчиков; способами обмена данными между EV3 и компьютером с использованием USB-кабеля и Bluetooth. Изучают технические характеристики EV3: память, быстродействие; расположение и назначение портов, кнопки. Знакомятся с программной средой.

#### **5.Интерфейс EV3.**

В этом тематическом разделе учащиеся учатся знакомятся с интерфейсом блока EV3, учатся составлять программу движения роботом с использованием блока. Знакомятся с возможностями управления моторами, использованием датчиков для управления роботом. По завершению изучения теоретической части выполняют тест «Команды управления моторами в EV3 Program» и практическую работу: «Программируем робота без компьютера». Уже на данном этапе, в результате составленной программы, учащиеся смогут наблюдать действия реального, запрограммированного исполнителя, что является важной составляющей использование средств робототехнике в школьном курсе.

#### **6. Интерфейс программной среды LEGO Mindstorms Education EV3.**

После знакомства с возможностями программирования робота с использованием блока EV3, учащиеся знакомятся с визуальной программной средой EV3 Edu Edition и языком программирования LabVIEW. Изучают основные области программной среды: рабочее поле, палитра команд, страница аппаратных средств, редактор контента. Учатся настраивать параметры команд: мотор вперед мотор назад. По итогу изучения, выполняют две практических работы: «Плавный поворот» и «Поворот на месте».

#### **7. Программирование.**

Тематический раздел, посвященный программированию роботов, является ключевым в данном курсе. В процессе его изучения, учащиеся

учатся составлять программы, используя основные алгоритмические структуры: следование, ветвление, цикл.

Учащиеся знакомятся с типами переменных, видами разветвляющихся и циклических алгоритмов, рассматривают элемент «Переключатель» и его параметры. Выполняют тест на тему «Линейный, циклический и разветвляющийся алгоритмы» и практические работы: «Квадратик», «Движение по лабиринту», «Движение вокруг квадрата». После, рассматриваются алгоритмы управления (релейный регулятор, пропорциональный регулятор, пропорционально – дифференциальный регулятор) и выполняются еще две практические работы с использованием регуляторов: «Робот описывает восьмерку», «Змейка».

### **8. Задачи для робота**

Следующий тематический раздел, не менее важный, чем предыдущий, также является значимым. Учащиеся решают поставленные перед ними задачи, учатся составлять программы с использованием различных датчиков, выполняют несколько практических работ по определению уровня шума, движению по звуковому сигналу, поиску черной линии, измерению расстояния до объекта.

### **9. Индивидуальные работы над проектами.**

В завершении курса, учащиеся делятся на группы и выполняют итоговую проектную работу, демонстрируя свои знания и умения в конструировании и программировании роботов.

### **10. Подведение итогов года.**

На последнем занятии, организуется выставка, группы демонстрируют свои проекты, учащиеся делятся впечатлениями и подводят итоги работы.

## ЗАКЛЮЧЕНИЕ

Роль подготовки молодежи в области информатики и информационных технологий продолжает оставаться очень высокой. Информационная компонента остается ведущей составляющей технологической подготовки человека и предъявляет высокие требования к уровню алгоритмической культуры выпускников общеобразовательной школы.

В ФГОС предприняты меры по усилению алгоритмической линии. Исходя из анализа, содержательной линии «Алгоритмы и элементы программирования» видно, что, в настоящее время, алгоритмическая линия является ведущей в школьном курсе информатики. Подробно раскрыто ее содержание и уточнены требования к предметным результатам учащихся, предъявляемым федеральным стандартом.

В примерной программе, соответствующей ФГОС, появляется новый раздел- робототехника, который пока является не обязательным для изучения и относится к дополнительным предметным результатам. Нетрудно предположить, что вскоре, пройдя через имеющиеся трудности, «Робототехника» займет одно из ведущих мест в базовой части раздела «Алгоритмы и элементы программирования».

Робототехника, в свою очередь,- отличный инструмент для наглядного представления работы алгоритмических конструкций и применение ее в процессе обучения информатики поможет развить алгоритмическую культуру учащихся.

Изучив теоретическую сторону исследования, для развития алгоритмической культуры учащихся был разработан факультативный курс "Введение в робототехнику на базе Lego Mindstorms EV3".

При составлении программы факультативного курса были рассмотрены теоретические сведения и возможности робототехнического комплекса Lego

Mindstorms EV3 и программного обеспечения Lego Mindstorms EV3 Edu Edition. Разработана результативно-целевая модель курса: обоснована актуальность курса, определено место в образовательном процессе, выделены цели изучения, спланированы формируемые компетенции и планируемые результаты освоения курса. Далее, было подобрано и раскрыто содержание разделов курса, отражающее рекомендации по планированию учебного материала и методике обучения алгоритмизации с использованием основ робототехники. Каждая тема курса раскрыта с точки зрения планируемых образовательных результатов.

Включение факультативного курса "Введение в робототехнику на базе Lego Mindstorms EV3" рекомендовано параллельно с изучением раздела алгоритмизации. Материалы выпускной квалификационной работы имеют практическую значимость для возможного использования преподавателями информатики общеобразовательных учреждений.

## СПИСОК ИСПОЛЬЗОВАННЫХ ИСТОЧНИКОВ

1. Федеральный Государственный образовательный стандарт основного общего образования.
2. Примерная основная образовательная программа среднего общего образования.
3. Руководство пользователя LegoMindstorms Education EV3.
4. Вязовов С.М. Соревновательная робототехника: приемы программирования в среде EV3 / С.М. Вязовов, О.Ю. Калягина, К.А. Слезин. – М.: Перо, 2013. – 132 с.
5. Копосов Д.Г. Первый шаг в робототехнику / Д.Г. Копосов. – М.: БИНОМ; Лаборатория знаний, 2012. – 189 с.
6. Лапчик М.П. Методика преподавания информатики / М.П. Лапчик, И.Г. Семакин, Е.К. Хеннер. – М.: Изд. центр «Академия», 2014. – 624 с.
7. Лебедев С.А. Философия науки: Словарь основных терминов. — М.: Академический Проект, 2004. — 320 с.
8. Мякушко А.А. Основы образовательной робототехники / А.А. Мякушко, Н.М. Сичинская, Ю.В. Смирнова.– М.: Перо, 2014. – 80 с.
9. Образовательная робототехника на уроках информатики и физики в средней школе / В.Н. Халамов [и др.]. -2-е изд., стереотип. – Челябинск, 2016. -160 с.
10. Овсяницкая Л.Ю. Курс программирования робота Lego Mindstorms EV3: основные подходы, практические примеры, секреты мастерства/ Д.Н. Овсяницкий, А.Д. Овсяницкий. – Челябинск, 2014. – 204 с.
11. Семакин И.Г. Примерная рабочая программа информатика 7-9 классы / И.Г. Семакин, М.С. Цветкова — М.: БИНОМ. Лаборатория знаний, 2016 г.

12. Угринович Н.Д. Примерная рабочая программа информатик 7-9 / Н.Д. Угринович, Н.Н. Самылкина. — М.: БИНОМ. Лаборатория знаний, 2016.
13. Лучко Л.Г. Формирование алгоритмической культуры учащихся в процессе обучения базовому курсу информатики: диссертация на соискание ученой степени кандидата педагогических наук / Омск, 2013.
14. Родионова О.Н. Подготовка будущих специалистов дошкольного образования к формированию элементов алгоритмической культуры у детей 5-6 лет: диссертация ... кандидата педагогических наук: 13.00.08. Краснодар, 2013 г.
15. Газейкина А.И. Формирование когнитивных универсальных учебных действий при обучении робототехнике учащихся основной школы / А.И. Газейкина, С.Г. Пронин // Педагогическое образование в России. — Екатеринбург: УрГПУ, 2015. — № 7. — С. 42.
16. Гайсина И.Р. Развитие робототехники в школе / И.Р. Гайсина // Педагогическое мастерство: материалы II Междунар. науч. конф. (г. Москва, декабрь 2012 г.). — М.: Буки-Веди, 2012. — С. 105-107.
17. Драч А.Н. Специфика обучения программирования в ССУЗЕ и ВУЗЕ / А.Н. Драч // мат. Междунар. науч.- метод. конф. «Информатизация образования» – 2008 (Славянск-на-Кубани, 27-30 мая 2008 г.). - Славянск-на-Кубани: Изд-во Славянского-на-Кубани гос. пед. ин-та, 2015. - С. 36.
18. Дубова Е.В. Формирование алгоритмической культуры на уроках математики / Е.В. Дубова, М.Ю. Солощенко // Наука и образование: проблемы и тенденции развития. - 2015. - № 1 (3). - С. 15-17.
19. Дуванов А.А. Кукарача снова рулит (теория и практика вступления в программирование) / А.А. Дуванов, Н.Д. Шумилина // Информатика. - Первое сентября. - 2015. - № 9. - С. 4-21.
20. Ершов М.Г. Робототехника как объект изучения в курсе физики средней школы / М.Г. Ершов // Педагогическое образование в России. — Екатеринбург: УрГПУ, 2015. — №3. — С. 124.

21. Зуев П.В. Проблемы преемственности в изучении робототехники в школе и вузе / П.В. Зуев, Е.С. Кощеева // Педагогическое образование в России. – Екатеринбург: УрГПУ, 2014. – №8. – С. 54.

22. Кормилицына Т.В. Формирование алгоритмической культуры при изучении специализированных программных сред / Т.В. Кормилицына // Учебный эксперимент в образовании. - 2015. - № 2 (74). - С. 44-49.

23. Левитин Е.С. Почему развитие в рамках системного подхода операционно-алгоритмических навыков и логико-аналитического мышления должно в XXI веке стать фундаментом всего образования, особенно математического? / Е.С. Левитин // В сборнике: Системный анализ и информационные технологии (САИТ - 2017). Сборник трудов Седьмой Международной конференции. - 2017. - С. 98-105.

24. Литвин А.В. Педагогические и дидактические возможности образовательной робототехники / А.В. Литвин // Инновации в образовании. – М.: Изд-во СГУ, 2012. – № 5. – С. 148.

25. Наумов А.А. Алгоритмическая культура в контексте подготовки специалистов в области информатики / А.А. Наумов // Вестн. Моск. пед. университета. Сер. Информатика и информатизация образования. – 2013. – № 7. – С. 268 – 269.

26. Нетесова О.С. Методические особенности реализации элективного курса по робототехнике на базе комплекта Lego Mindstorms NXT 2.0 / О.С. Нетесова // Информатика и образование. - 2013. - № 7. - С. 74-76.

27. Остапенко С.И. Формирование алгоритмической культуры учащихся на уроках математики / С.И. Остапенко // В сборнике: Актуальные вопросы образования и науки Сборник научных трудов по материалам международной научно-практической конференции. - 2018. - С. 117-119.

28. Родыгина Т.М. Формирование информационной культуры обучающихся на уроках информатики посредством алгоритмизации деятельности / Т.М. Родыгина // В сборнике: Интернет-технологии в образовании Материалы всероссийской с международным участием научно-

практической конференции. МОО «Академия информатизации образования»  
ОО «Чувашское региональное отделение Академии информатизации  
образования»; Ответственный редактор Н. В. Софронова. - 2015. - С. 117-122.

29. Шрайнер А.А. Алгоритмическая культура - ресурс образования /  
А.А. Шрайнер // В сборнике: Технологическое образование (проблемы и  
перспективы развития) Сборник трудов Всероссийской научно-практической  
конференции. Министерство образования РФ, Научно-методический Совет  
по специальности «Технология и предпринимательство» при МО РФ,  
Новосибирский государственный педагогический университет,  
Новосибирский институт повышения квалификации и переподготовки  
работников образования. - 2013. - С. 216-222.

30. Штода В.Н. Использование комплекса виртуальных лабораторий  
на уроках информатики как средство формирования алгоритмической  
культуры и логического мышления школьников / В.Н. Штода // В сборнике:  
Информационные технологии в образовании V Всероссийская (с  
международным участием) научно-практическая конференция. - 2013. - С.  
71-73.

31. Юнева Л.С. Формирование алгоритмической культуры у учащихся  
/ Л.С. Юнева // В сборнике: Современные научные исследования:  
методология, теория, практика материалы VIII Международной научно-  
практической конференции. Центр содействия развитию научных  
исследований. - 2015. - С. 111-120.

32. Алферьева Т.И. Формирование алгоритмической культуры при  
изучении математических дисциплин / Т.И. Алферьева. – [Электронный  
ресурс]. - URL: [www.dusk12.ru/teoria/alf.doc](http://www.dusk12.ru/teoria/alf.doc).

33. Лучко Л.Г. Формирование алгоритмической культуры учащихся  
как системообразующая функция базового курса информатики / Л.Г. Лучко.  
– Электронный ресурс: <https://search.rsl.ru/ru/record/01000205609>

**Рабочая программа факультативного курса «Введение в робототехнику на базе LegoMindstorms EV3»**

**Пояснительная записка**

Программа факультативного курса «Введение в робототехнику на базе LegoMindstorms EV3» составлена в соответствии с требованиями Федерального государственного образовательного стандарта основного общего образования, Примерной основной образовательной программы основного общего образования. Содержанию курса соответствует содержанию дополнительного модуля «Робототехника» в главе «Алгоритмы и элементы программирования» примерной программы по информатике.

Настоящий курс предлагает использование образовательных конструкторов MINDSTORMS EV3, как инструмента для обучения школьников конструированию, моделированию и компьютерному управлению на уроках робототехники. Простота в построении модели в сочетании с большими конструктивными возможностями конструктора позволяют детям в конце урока увидеть сделанную своими руками модель, которая выполняет поставленную ими же самими задачу. При построении модели затрагивается множество проблем из разных областей знания - от теории механики до психологии. Использование компьютера, как средства управления моделью; его использование направлено на составление управляющих алгоритмов для собранных моделей.

***Типология курса:*** прикладной.

**Место курса в образовательном процессе:** Программа факультативного курса «Введение в робототехнику на базе LegoMindstorms EV3» предназначена для учащихся 6-9 классов. Программа рассчитана на 34 часа (2 часа резервные).

**Актуальность курса:** Курс направлен на формирование творческой личности, живущей в современном мире, развитие алгоритмической и информационной культуры учащихся. Работа с образовательными конструкторами LEGO позволяет школьникам в форме познавательной игры узнать многие важные идеи и развить необходимые в дальнейшей жизни навыки. Курс ориентирован на изучение основных физических принципов и базовых технических решений, лежащих в основе всех современных конструкций и устройств. В учебной деятельности полученные знания смогут помочь учащимся сформировать представление об особенностях составления программ управления, автоматизации механизмов, моделировании работы систем. Тем самым курс способствует дальнейшему самообразованию в области алгоритмизации и программирования.

## **1. Цели изучения курса**

### **Образовательные:**

- Углубление знаний по основным принципам механики и основам алгоритмизации и программирования; ознакомление с основами программирования в компьютерной среде Mindstorms EV3.

### **Развивающие:**

- Развитие алгоритмического, абстрактного и логического мышления. Развитие умения творчески подходить к решению задачи. Развитие научно-технического и творческого потенциала личности ребенка

путем организации его деятельности в процессе интеграции начального инженерно-технического конструирования и основ робототехники.

- Развитие умения довести решение задачи до работающей модели. Развитие умения излагать мысли в четкой логической последовательности, отстаивать свою точку зрения, анализировать ситуацию и самостоятельно находить ответы на вопросы.

**Воспитательные:**

- формирование творческого подхода к поставленной задаче;
- формирование представления о том, что большинство задач имеют несколько решений;
- формирование целостной картины мира;
- ориентирование на совместный труд.

<b>Формируемые компетенции</b>	
<b>Ключевые общеобразовательные (общеучебные умения, способы деятельности; универсальные учебные действия)</b>	Навыки индивидуальной работы в процессе моделирования и описания алгоритма
	Проявление творческих способностей в процессе моделирования и составлении программы для решения поставленной задачи
	Умение анализировать, отбирать и структурировать необходимую информацию для создания конкретной модели
	Умение планировать свою деятельность при создании модели и написании программы для исполнителя
	Адекватная самооценка своей деятельности в процессе создания модели

	<p>Готовность к повышению своего образовательного уровня и продолжению обучения с использованием средств и методов информатики и ИКТ</p>
<p><b>Предметные компетенции (предметные умения, способы деятельности)</b></p>	<p>представления об алгоритме, модели — и их свойствах;</p>
	<p>представление о возможностях, и инструментах LEGO Mindstorms EV3.</p>
	<p>умение программировать контролер и EV3. сенсорные системы;</p>
	<p>умение пошагово выполнять несложные алгоритмы управления исполнителями и анализа числовых и текстовых данных;</p>
	<p>владение алгоритмами и методами решения организационных и технических задач;</p>
	<p>навыками использования информационных устройств: компьютера, мобильного телефона, конструктора LEGO.</p>

## 2. Тематическое планирование курса

№	Тема	Всего часов
1	<b>Вводное занятие. Презентация программы. ТБ</b>	2
<b>2</b>	<b>Конструирование (2 часа)</b>	
2.1	Способы крепления деталей. Высокая башня.	1
2.2	Механический манипулятор (хваталка).	1
<b>3</b>	<b>Первые модели (4 часа)</b>	
3.1	Тележки. История колеса. Одномоторная тележка.	1
3.2	Тележка с автономным управлением.	1
3.3	Двухмоторная тележка. Полный привод.	2
<b>4</b>	<b>Подключения EV3.</b>	2
<b>5</b>	<b>Интерфейс EV3. Составление программ с использованием блока EV3.</b>	2
<b>6</b>	<b>Интерфейс программной среды LEGO Mindstorms Education EV3 (или Robolab 2.9 или Robot C)</b>	2
<b>7</b>	<b>Программирование на языке EV3(7 часов)</b>	
7.1	Следование	1
7.2	Ветвление	2
7.3	Циклы	2
7.4	Алгоритмы управления.	2
<b>8</b>	<b>Задачи для робота (8 часов)</b>	
8.1	Поворот, парковка в гараж, остановка (датчик касания).	2
8.2	Движения по звуковому сигналу, определение уровня шума (датчик звука).	2
8.3	Движение вдоль линии. Один датчик света/цвета.	2
8.4	Движение за рукой используя датчик ультразвука.	2
<b>9</b>	<b>Индивидуальные работы над проектами</b>	<b>2</b>
<b>10</b>	<b>Подведение итогов</b>	<b>1</b>
	<b>ВСЕГО:</b>	<b>34(2 резерв)</b>

### **3. Содержание разделов курса.**

#### **1. Вводное занятие.**

Введение в предмет «Робототехника». Что такое робот? Какие бывают роботы. Современные тенденции робототехники. Зарубежные и отечественные разработки. Презентация программы. Техника безопасности на занятиях. Правила внутреннего распорядка и поведение в коллективе. Знакомство с конструктором. Правила работы с конструктором.

*Тест «Робототехника. Современные тенденции робототехники.»*

#### **2. Конструирование.**

##### **2.1. Способы крепления деталей. Высокая башня.**

Различия принципов конструирования RIS и EV3. Способы крепления деталей. Жесткая конструкция. Конструирование самой высокой и устойчивой башни. Высота, устойчивость.

*Практическая работа: конструируем модель «Башня».*

##### **2.2. Механический манипулятор (хваталка).**

Подвижная конструкция. Понятие механизма.

*Практическая работа: конструируем модель «Механический манипулятор».*

*Тест «Принципы конструирования»*

#### **3. Первые модели.**

##### **3.1. Тележки. История колеса. Одномоторная тележка.**

*Практическая работа: конструируем модель «Одномоторная тележка».*

##### **3.2. Тележка с автономным управлением.**

Микроконтроллер. Автономное управление.

*Практическая работа: конструируем модель «Тележка с автономным управлением».*

3.3. Двухмоторная тележка. Полный привод.

Центр тяжести. Трехколесная тележка.

*Практическая работа: конструируем модель «Двухмоторная тележка» (приложение А).*

*Практическая работа: конструируем модель «Двухмоторный вездеход».*

#### **4. Подключения EV3.**

Подключение электромоторов, датчиков, обмен данными между EV3 и компьютером с использованием USB-кабеля и Bluetooth. Технические характеристики EV3. Память, быстродействие. Порты. Кнопки. Элементы питания. Программные среды.

*Тест «Технические характеристики EV3»*

#### **5.Интерфейс EV3.**

Составление программ с использованием блока EV3. Возможности управления моторами. Датчики. Использование датчиков для управления роботом. Команды управления моторами в EV3Program.

*Тест «Команды управления моторами в EV3Program»*

*Практическая работа: «Программируем без компьютера».*

#### **6. Интерфейс программной среды LEGO Mindstorms Education EV3.**

Язык программирования LabVIEW. Окно программы. Палитра команд. Рабочее поле программы. Командный центр. Настройка параметров команд. Мотор вперед. Мотор назад. Поворот.

*Практическая работа: «Плавный поворот», «Поворот на месте».*

#### **7. Программирование.**

7.1. Следование.

Линейный алгоритм. Повороты.

*Практические работы: «Квадратик», «Движение по заданному лабиринту».*

7.2. Ветвление.

Ветвление. Переключатели. Режимы отражения блока «Ветвление».  
Параллельные ветвление.

*Практическая работа: «Движение по лабиринту»*

7.3. Циклы.

Цикл с параметром. Цикл с постусловием. Переменные. Три типа переменных.

*Практическая работа: Программа «Вокруг квадрата».*

*Тест «Линейный, циклический и разветвляющийся алгоритмы»*

7.4. Алгоритмы управления (релейный регулятор, пропорциональный регулятор, пропорционально – дифференциальный регулятор).

*Практическая работа: Программа с использованием П-регулятора «Робот описывает восьмерку», «Змейка».*

## **8. Задачи для робота**

8.1. Поворот, парковка в гараж, остановка (датчик касания).

Параллельные процессы. Использование датчика касания. Управление моторами.

*Практическая работа: Программа «Парковка в гараж».*

8.2. Движения по звуковому сигналу, определение уровня шума (датчик звука).

Использование датчика звука. Управление моторами. Измерение уровня шума.

*Практическая работа: Программа «Активация робота звуком».*

8.3. Движение вдоль линии. Один датчик света/цвета.

Использование датчика света или цвета. Измерение уровня освещенности. Определение цвета с помощью датчика.

*Практическая работа: Программа «Движение вдоль линии» (приложение Б).*

*Практическая работа: Программа «Обнаружение черной линии».*

8.4. Движение за рукой используя датчик ультразвука.

Использование датчика ультразвука. Измерение расстояния.

*Практическая работа: Программа «Робот-прилипала».*

*Тест «Датчики EV3»*

#### **9. Индивидуальные работы над проектами.**

Подготовка к итоговой проектной работе. Итоговая проектная работа.

#### **10. Подведение итогов года.**

Выставка. Презентация проекта. Подведение итогов работы за год.

### **4. Выявление уровня освоения содержания.**

#### **Требования к знаниям, умениям и способам деятельности:**

В результате освоения курса учащийся должны:

##### **знать:**

- правила безопасной работы с робототехническим конструктором LEGO Mindstorms EV3;
- основные компоненты конструктора LEGO Mindstorms EV3;
- конструктивные особенности различных моделей, сооружений и механизмов;
- компьютерную среду Mindstorms Education EV3, включающую в себя графический язык программирования;
- виды подвижных и неподвижных соединений в конструкторе;
- основные приемы конструирования роботов;
- конструктивные особенности различных роботов;
- как передавать программы;
- как использовать созданные программы;

##### **уметь:**

- работать с литературой, с журналами, с каталогами, в интернете (изучать и обрабатывать информацию);

- самостоятельно решать технические задачи в процессе конструирования роботов (планирование предстоящих действий, самоконтроль, применять полученные знания, приемы и опыт конструирования с использованием специальных элементов и т.д.);
- осуществлять управление имеющимся формальным исполнителем;
- разрабатывать алгоритм решения практических задач;
- реализовывать блок-схемы, понимать описания алгоритмов в программном обеспечении Lego Mindstorms Education EV3;
- подбирать алгоритмическую конструкцию, соответствующую заданной ситуации;
- составлять линейные, ветвящиеся и циклические алгоритмы управления для исполнителя Lego Mindstorms EV3;
- выполнять тестирование и отладку алгоритма для исполнителя Lego Mindstorms EV3;
- выделять подзадачи: определять и использовать вспомогательные алгоритмы;
- применять полученные знания в практической деятельности;

**Владеть навыками:**

- разработки плана действия для решения практических задач;
- работы с роботом LegoMindstormsEV3 и средой Mindstorms Education EV3

**КОНТРОЛЬНЫЕ МЕРОПРИЯТИЯ**

п/п	№	Тема	Цель контроля	Вид контроля
-----	---	------	---------------	--------------

1	Способы крепления деталей. Высокая башня. Пр. р.: Конструируем модель «Башня»	Проверка степени освоения теоретического материала и умение применить его на практике.	Корректирующий
2	Механический манипулятор (хваталка). Тест «Принципы конструирования»	Контроль теоретических знаний учащихся	Текущий
3	Интерфейс EV3 Пр. р.: «Программируем без компьютера».	Проверка степени освоения теоретического материала и умение применить его на практике.	корректирующий
4	Интерфейс EV3 Тест «Команды управления моторами в EV3Program»	Контроль теоретических знаний учащихся	Текущий
5	Циклы. Ветвление. Практические работы по программированию циклов и ветвлений	Проверка степени освоения теоретического материала и умение применить его на практике.	тематический
6	Циклы. Ветвление. Тест «Циклический и разветвляющийся алгоритмы»	Контроль теоретических знаний учащихся	Текущий
7	Задачи для робота. Практические работы по использованию	Проверка степени освоения теоретического материала и умение применить	Корректирующий

	датчиков (касания, звука, света, цвета, ультразвука)	его на практике.	
8	Задачи для работа Тест «Датчики EV3»	Контроль теоретических знаний учащихся	Текущий
9	Итоговая проектная работа, защита индивидуальных или коллективных творческих проектов-роботов	Оценка личных достижений учащихся	Итоговый

## **ОБОСНОВАНИЕ СИСТЕМЫ КОНТРОЛЬНО-ОЦЕНОЧНОЙ ДЕЯТЕЛЬНОСТИ**

### **Механизм оценивания образовательных результатов.**

Для отслеживания уровня усвоения знаний и умений используются: тестовые диагностические работы; тематические проверочные работы.

Система оценивания выстраивается таким образом, чтобы учащиеся включались в контрольно-оценочную деятельность, приобретая навыки и привычку к самооценке.

Подходы к оцениванию представляются следующим образом: оценивание тестовых и практических работ по 5- бальной системе. Так же используется вербальное оценивание работ учащихся для мотивации личностного развития: поощрение, похвала, одобрение.

### **Формы подведения итогов реализации программы:**

*Текущий контроль* предполагает оценку усвоения знаний и умений по итогам выполнения заданий на практических занятиях. После изучения

каждого раздела предполагается проверка теоретических знаний в форме теста и практическое задание по комплексному применению умений.

*Итоговый контроль* – защита индивидуального или коллективного творческого проекта-робота (300 баллов)

По окончании курса выставляется зачет (учитывая текущий и итоговый контроль).

### **Контроль знаний и умений**

#### **Оценка достижений учащихся.**

1. Выполнение практических заданий – 0-50 баллов.
2. Выполнение теста – 0-10 баллов.
3. Выполнение итогового проекта – 300 баллов.

### **Критерии оценивания практических работ**

№	Критерий оценивания	Колл. баллов	самооценка	Оценка учителя	Итоговая оценка
1	Оригинальность и творческий подход	0-5			
2	Сложность конструкции	0-15			
3	Уровень программирования	0-15			
4	Соответствие техническим требованиям (изученной темы)	0-15			

Перевод в 5-ти балльную систему:

41-50 бал.-----«5»

31-40 бал.-----«4»

25-30 бал.-----«3»

менее 25 бал.-----«2»

### **Оценивание тестов**

*Используются тесты, в которых представлены задания разного типа.*

Выполнение теста оценивается по 10-балльной шкале.

**«Отлично»** - 9-10 баллов

**«Хорошо»** - 7-8 баллов

**«Удовлетворительно»** - 5-6 баллов

**«Неудовлетворительно»** - менее 5 баллов.

### **Критерии оценивания итогового творческого проекта**

Защита проекта	Критерий оценивания	Колл. баллов	Оценка учеников	Оценка учителя	Итоговая оценка
выступление	Оригинальность и творческий подход	0-30			
	Качество выступление (защиты проекта) <ul style="list-style-type: none"><li>• Наличие презентации;</li><li>• грамотная речь;</li><li>• доступность.</li><li>• артистичность</li><li>• логичность</li></ul>	0-30			
проект	Уровень	0-60			

	программирования				
	Техническая сложность конструкции	0-60			
	Подвижность и функциональность	0-60			
	Соответствие техническим требованиям (изученной темы)	0-60			

## 5. Учебно-методическое обеспечение

Для проведения занятий по программе необходимо использовать образовательные конструкторы LEGO Mindstorms EV3 и дополнительные элементы.

### Информационные ресурсы

<http://www.prorobot.ru> - EV3 инструкции

<https://geektimes.ru/post/116408/> -Lego Mindstorms EV3 — краткое знакомство

[http://robotkurs.clan.su/index/znakomimsja\\_s\\_naborom\\_lego\\_mindstorms\\_EV3\\_2\\_0/0-22](http://robotkurs.clan.su/index/znakomimsja_s_naborom_lego_mindstorms_EV3_2_0/0-22) - Знакомимся с набором Lego Mindstorms EV3

### Программно-аппаратные ресурсы

Комплекты LEGO Mindstorms Education EV3

Проектор

Интерактивная доска

Локальная сеть

Доступ к Internet

WEB-браузер (Mozilla Firefox, Opera.)

### **Дидактические средства обучения, в том числе и на основе ИКТ**

1. «Уроки Лего – конструирования в школе», Злаказов А.С., Горшков Г.А., 2011 г., БИНОМ.

2. Программное обеспечение робота Lego Mindstorms EV3. Мультимедийный CD-ROM

3. Комплект заданий «EV3 Maker для основной школы». *Дополнительный материал к EV3.* Мультимедийный CD-ROM

4. Овсяницкая Л.Ю. Курс программирования робота Lego Mindstorms EV3: основные подходы, практические примеры, секреты мастерства/ Д.Н. Овсяницкий, А.Д. Овсяницкий. – Челябинск : И.В. Мякотин, 2014. – 204 с.

5. *Комплект заданий по теме "Естественные науки и регистрация данных".* *Дополнительный материал к EV3.* Мультимедийный CD-ROM

### **Методические особенности организации учебно-познавательной деятельности**

В течение курса учащиеся посещают теоретические, практические занятия, где у них есть возможность изучить материал, а затем закрепить его, используя консультации учителя, на практических занятиях. По мере прохождения материала курса, учащиеся получают возможность создавать собственные модели, что способствует их творческому развитию.

Учащиеся могут использовать информационные ресурсы (учебные пособия, дополнительную литературу, Internet-ресурсы) для детального изучения материала и более полноценного создания той или иной модели.

Наряду с современным образовательным технологиям, отраженными в принципах, формах и методах обучения: индивидуальности, доступности, преемственности, результативности, - широко используется работа по методу творческого проекта. На занятиях предлагается выполнение мини-проектов по изучаемой теме из деталей LEGO конструктора.

Итоговым результатом прохождения курса является создание индивидуального или группового творческого проекта. Таким образом, личностно-ориентированный подход при изучении этого курса реализуется за счет индивидуального выбора тематики проекта, индивидуального темпа выполнения индивидуального проекта.

### **Особенности коммуникационного взаимодействия**

Учитель постоянно находится в диалоге с учащимися, поощряет инициативу; преобладает стиль дружеского взаимодействия (модель системы «учитель ◀+ ▶ученик»).

**Оценочная модель уровня сформированности основных  
компонентов алгоритмической культуры.**

Компоненты алгоритмической культуры (по Лапчику)	Показатели	Критерии оценивания
Понятие алгоритма и его свойства.	-Понимать определение алгоритма; -Определять основные свойства алгоритма.	-Называют определение алгоритма на понятном для себя уровне; -Называют и распознают основные свойства алгоритма: дискретность, понятность, точность, массовость, конечность.
Понятие языка описания алгоритмов.	-Понимать определения языка описания алгоритмов; -Знать основные конструкции управления языка программирования LabVIEW.	-Называют способы описания алгоритма (графический, словесный, алгоритмический); -Проводят аналогию между конструкциями блок-схемы и языком LabVIEW; -Составляют программы управления

		роботом в программной среде Education EV3.
Уровень формализации описания.	<p>-Понимать понятие формализации описания;</p> <p>-Понимать понятие «СКИ»;</p> <p>-Знать синтаксис языка LabVIEW.</p>	<p>-Понимают определение формализация описания на понятном для себя уровне;</p> <p>-Знают команды, входящие в «СКИ» исполнителей программной среды LEGO Mindstorms Education EV3;</p> <p>-Составляют программы управления роботом в программной среде Education EV3, согласно правилам синтаксиса языка.</p>
Дискретность (пошаговость) описания.	<p>-Понимать понятие «дискретность»;</p> <p>-Делить весь алгоритм на последовательность отдельных действий.</p>	<p>-Называют понятие «дискретность» на понятном для себя уровне;</p> <p>-Составляют алгоритм управления роботом, соблюдая свойство дискретности.</p>

«Блочность».	<p>-Понимать суть свойства «блочности»;</p> <p>-Понимать понятие «вспомогательный алгоритм».</p> <p>-Разделять сложную задачу на более простые части.</p>	<p>-Называют суть свойства «блочности» на понятном для себя уровне;</p> <p>-Формулируют понятие «вспомогательный алгоритм»;</p> <p>-Используют вспомогательные алгоритмы при составлении программы для робота.</p>
Ветвление.	<p>-Понимать назначение и принцип работы структуры ветвление;</p> <p>-Использовать структуру ветвления при написании алгоритма действий для исполнителя.</p>	<p>-Понимают структуру ветвления на понятном для себя уровне;</p> <p>-Используют «Переключатель» (условие) для создания алгоритмов движения робота;</p> <p>-Продумывают все допустимые варианты условия при написании программы.</p>
Цикличности.	<p>-Понимать назначение цикличности;</p> <p>-Знать виды циклов: с параметром, с постусловием,</p>	<p>-Понимают структуру цикличности на понятном для себя уровне;</p> <p>-Используют различные</p>

	<p>предусловием;</p> <p>-Использовать структуру повторения при написании алгоритма действий для исполнителя.</p>	<p>виды циклов при составлении программы для робота.</p>
<p>Выполнение (обоснование) алгоритма.</p>	<p>-Понимать важность и суть определенного алгоритма;</p> <p>-Уметь правильно обосновать и составить конкретный алгоритм действий.</p>	<p>-Понимают обоснование алгоритма на понятном для себя уровне;</p> <p>-Составляют алгоритм строго в соответствии с поставленной задачей.</p>
<p>Организация данных</p>	<p>-Понимать понятие «начальные данные»;</p> <p>-Понимать необходимость фиксации промежуточных и конечного результата;</p> <p>-Знать способы фиксирования промежуточных результатов работы алгоритма.</p>	<p>-Определяют понятие «начальные данные» на понятном для себя уровне;</p> <p>-Применяют способы фиксации промежуточных данных и конечного результата при составлении программы для робота;</p>