



МИНИСТЕРСТВО НАУКИ И ВЫСШЕГО ОБРАЗОВАНИЯ
РОССИЙСКОЙ ФЕДЕРАЦИИ

Федеральное государственное бюджетное образовательное
учреждение высшего образования

«ЮЖНО-УРАЛЬСКИЙ ГОСУДАРСТВЕННЫЙ
ГУМАНИТАРНО-ПЕДАГОГИЧЕСКИЙ
УНИВЕРСИТЕТ» (ФГБОУ ВО «ЮУрГГПУ»)

ФИЗИКО – МАТЕМАТИЧЕСКИЙ ФАКУЛЬТЕТ
КАФЕДРА ИНФОРМАТИКИ, ИНФОРМАЦИОННЫХ ТЕХНОЛОГИЙ И МЕТОДИКИ
ОБУЧЕНИЯ ИНФОРМАТИКЕ

**Методика обучения программированию в школе на основе
микроэлектроники**

**Выпускная квалификационная работа по направлению
44.04.01 Педагогическое образование**

Направленность программы магистратуры

«Информатика в образовании»

Форма обучения заочная

Проверка на объем заимствований:

81 % авторского текста
Работа рекомендована к защите
рекомендована/не рекомендована
«23» августа 2020г.
зав. кафедрой ИИТМОИ ЮУрГГПУ
Рузаков Андрей Александрович

Выполнил:

Студент группы ЗФ-313-125-2-1
Андреев Сергей Владимирович

Научный руководитель:
к.п.н., доцент кафедры ИИТМОИ
Носова Людмила Сергеевна

Челябинск
2020 год



МИНИСТЕРСТВО НАУКИ И ВЫСШЕГО ОБРАЗОВАНИЯ
РОССИЙСКОЙ ФЕДЕРАЦИИ

Федеральное государственное бюджетное образовательное
учреждение высшего образования

«ЮЖНО-УРАЛЬСКИЙ ГОСУДАРСТВЕННЫЙ
ГУМАНИТАРНО-ПЕДАГОГИЧЕСКИЙ
УНИВЕРСИТЕТ» (ФГБОУ ВО «ЮУрГГПУ»)

ФИЗИКО – МАТЕМАТИЧЕСКИЙ ФАКУЛЬТЕТ
КАФЕДРА ИНФОРМАТИКИ, ИНФОРМАЦИОННЫХ ТЕХНОЛОГИЙ И МЕТОДИКИ
ОБУЧЕНИЯ ИНФОРМАТИКЕ

**Методика обучения программированию в школе на основе
микроэлектроники**

**Выпускная квалификационная работа по направлению
44.04.01 Педагогическое образование**

Направленность программы магистратуры

«Информатика в образовании»

Форма обучения заочная

Проверка на объем заимствований:
_____ % авторского текста
Работа _____ к защите
рекомендована/не рекомендована
«___» _____ 2020г.
зав. кафедрой ИИТМОИ ЮУрГГПУ
Рузаков Андрей Александрович

Выполнил:
Студент группы ЗФ-313-125-2-1
Андреев Сергей Владимирович
Научный руководитель:
к.п.н., доцент кафедры ИИТМОИ
Носова Людмила Сергеевна

Челябинск
2020 год

ОГЛАВЛЕНИЕ

ВВЕДЕНИЕ.....	3
ГЛАВА 1. ТЕОРЕТИЧЕСКИЕ ОСОБЕННОСТИ ОБУЧЕНИЯ ПРОГРАММИРОВАНИЮ.....	6
1.1 Особенности обучения программированию в школе.....	6
1.2 Методики обучения программированию в школе.....	9
1.3 Методика «Перевернутый класс» в изучении программирования в школе.....	14
Вывод по Главе 1.....	18
ГЛАВА 2. ФАКУЛЬТАТИВНЫЙ КУРС «ПРОГРАММИРОВАНИЕ МИКРОКОНТРОЛЛЕРОВ».....	19
2.1 Особенности обучению программированию с использованием микроконтроллеров.....	19
2.2 Программа внеурочной деятельности «Программирование микроконтроллеров».....	20
Выводы по Главе 2.....	33
ГЛАВА 3. ПЕДАГОГИЧЕСКИЙ ЭКСПЕРИМЕНТ И ОЦЕНКА ЭФФЕКТИВНОСТИ ВВЕДЕНИЯ ФАКУЛЬТАТИВНОГО КУРСА «ПРОГРАММИРОВАНИЕ».....	34
3.1 Организация и проведение педагогического эксперимента.....	34
3.2 Анализ результатов тестирования по методике «Диагностика направленности мотивации изучения предмета» у обучающихся в МБОУ «Гимназия № 48 г. Челябинска».....	35
Выводы по Главе 3.....	41
ЗАКЛЮЧЕНИЕ.....	42
СПИСОК ИСПОЛЬЗОВАННЫХ ИСТОЧНИКОВ.....	43

ВВЕДЕНИЕ

Целью обучения программированию школе является освоение практического программирования на примере несложных практических задач, формирование представления о современной технологии программирования.

У обучающихся развиваются навыки применять, анализировать и преобразовывать информацию реальных объектов и процессов.

Умение разбить задачу на подзадачи, умение воспользоваться готовым алгоритмом более простой задачи при решении сложной – это общеучебные умения и навыки, которые формируются у каждого ученика на уроках информатики, благодаря чему, развивается алгоритмическое и абстрактное мышление.

Можно сделать вывод, что **актуальность** исследования задается современными тенденциями развития информатики, как науки.

Проблема исследования определяется противоречием между потребностью в развитии информационно-технологической культуры, самостоятельности обучающихся, развитием алгоритмического мышления, и недостаточной базой школьной программы по данному курсу.

Объект исследования: процесс обучения учащихся алгоритмизации и программированию в лингво-гуманитарной школе во внеурочной деятельности.

Предмет исследования: методика обучения программированию на основе микроконтроллера во внеурочной деятельности.

Цель исследования: заключается в теоретическом обосновании, экспериментальной проверке методики обучения программированию во внеурочной деятельности.

Гипотеза исследования: если осуществлять обучение программированию на основе микроэлектроники с использованием метода

«Перевернутый класс», в классах лингво-гуманитарного профиля, то уровень мотивации к изучению информатики у обучающихся повысится.

Поставленная цель и выдвинутая гипотеза предполагают решение следующих **задач**:

1. Ознакомиться с особенностями изучения программирования в школе.
2. Проанализировать существующие методики обучению программированию.
3. Проанализировать особенности обучения программированию на микроконтроллерах.
4. Разработать курс по обучению программированию на основе микроэлектроники.
5. Провести педагогический эксперимент и оценить эффективность введения курса в внеурочной деятельности.

Эмпирическая база работы – Муниципальное бюджетное образовательное учреждение «Гимназия № 48 г. Челябинска» (далее МБОУ «Гимназия №48»).

Для решения поставленных задач использовались следующие **методы исследования**: теоретический анализ литературы по проблематике исследования; сравнение и обобщение педагогических исследований, системный анализ и классификация, анкетирование, статистические методы обработки экспериментальных данных, педагогический эксперимент.

Научно-практическая ценность и значимость исследования заключаются в том, что разработана модель методики обучения программированию во внеурочной деятельности, ориентированной на результат, предложена методика ее реализации с использованием активных методов обучения.

На защиту выносятся следующие положения:

1. Развитое алгоритмическое и абстрактное мышление обучающихся является одним из ведущих показателей уровня их подготовки по информатике.

2. Обучение программированию на основе микроэлектроники с использованием метода «перевернутый класс» способствует развитию формированию навыков самостоятельной работы и активной деятельности обучающихся.

Диссертация состоит из введения, трех глав, в которых решаются поставленные задачи исследования, заключения, списка источников и литературы.

ГЛАВА 1. ТЕОРЕТИЧЕСКИЕ ОСОБЕННОСТИ ОБУЧЕНИЯ ПРОГРАММИРОВАНИЮ

1.1 Особенности обучения программированию в школе

Понятие алгоритма на современном этапе развития системы образования является одним из наиболее важных в рамках курса информатики и информационных технологий. На первый план выносятся необходимость развития у обучающихся алгоритмического мышления, которое позволяет решать задачи различного уровня сложности с помощью составления детального плана действий [1].

Использование таких формальных языков, как блок-схемы и структурное программирование, может быть применено для записи алгоритмов и формирования алгоритмического мышления [17].

Немаловажное значение при ознакомлении с языками программирования имеет информация о классификации, правилах предоставления данных и нормах записи ключевых операторов: ввода, вывода, присваивания, ветвления, цикла. Данные знания необходимы для дальнейшего изучения этапов разработки программы: алгоритмизация – кодирование – отладка – тестирование.

В примерной программе по информатике существует раздел, посвященный изучению алгоритмов и исполнителей. В рамках курса обучающиеся познакомятся с основными понятиями: исполнители (состояния, возможные обстановки и система команд исполнителя), команды-приказы и команды-запросы, отказ исполнителя. Также уделяется внимание изучению самого алгоритма как плана управления исполнителем и его записи на алгоритмическом языке [17].

Рабочая программа предполагает реализацию ряда практических работ:

1. Разработка линейного алгоритма (программы) с использованием математических функций при записи арифметического выражения.
2. Разработка алгоритма (программы), содержащей оператор ветвления.
3. Разработка алгоритма (программы), содержащей оператор цикла.
4. Разработка алгоритма (программы), содержащей подпрограмму.
5. Разработка алгоритма (программы) по обработке одномерного массива.
6. Разработка алгоритма (программы), требующего для решения поставленной задачи использования логических операций.

Выпускник получит возможность:

1. Познакомиться с использованием в программах строковых величин и с операциями со строковыми величинами.
2. Создавать программы для решения задач, возникающих в процессе учебы и вне её.
3. Познакомиться с задачами обработки данных и алгоритмами их решения.
4. Познакомиться с понятием «управление», с примерами того, как компьютер управляет различными системами.

По окончании курса информатики в рамках школьной программы обучающиеся получают определенный набор знаний об алгоритмах обработки данных, методах их представления и программной реализации, дискретизации. Предполагается изучение одного из языков программирования и основных алгоритмических структур (линейной, условной и циклической), практика в области написания и настройки программ в разных средах программирования.

Таким образом, выпускники будут иметь представление об основах алгоритмической культуры, а также обладать навыками для написания несложных программ.

Существует несколько учебных языков программирования с использованием русских ключевых слов, что широко используется для преподавания программирования в общеобразовательных школах [7].

При участии академика А.П. Ершова был разработан метод работы с изучением нескольких языков программирования – более простого для управления графическим исполнителем и продвинутого для выполнения всех остальных задач. По данному принципу были созданы языки Робик и Рапира.

Основные российские образовательные языки программирования представлены в таблице 1.

Таблица 1 – Основные российские образовательные языки программирования

Язык программирования	Краткое описание
1	2
BASIC	Язык подходит на начальном этапе изучения программирования в основной школе, развивает алгоритмическое и логическое мышление.
Pascal	В базовом курсе информатики изучение языков программирования высокого уровня должно носить ознакомительный характер, поэтому для первоначального знакомства целесообразно использовать язык Паскаль, который был разработан в 1971 году Н. Виртом специально как учебный язык.

Продолжение таблицы 1

1	2
Python	Python создавался с одной важной целью: сделать более простым написание больших программ с минимумом ошибок по правилам процедурного программирования, не добавляя на итоговый код программ лишних 22 накладных расходов для компилятора, как это всегда делают языки очень высокого уровня, такие как Бейсик. Подходит для изучения структурного программирования в базовом школьном курсе информатик
C++	Поддерживает такие парадигмы программирования как процедурное программирование, объектно-ориентированное программирование, 23 обобщённое программирование, обеспечивает модульность, отдельную компиляцию, обработку исключений, абстракцию данных, объявление типов (классов) объектов, виртуальные функции. Стандартная библиотека включает, в том числе, общеупотребительные контейнеры и алгоритмы. C++ сочетает свойства как высокоуровневых, так низкоуровневых языков. В сравнении с его предшественником — языком C, — наибольшее внимание уделено поддержке объектно-ориентированного и обобщённого программирования.

1.2 Методики обучения программированию в школе

Методика обучения – это совокупность элементов образовательных взаимосвязанных форм, средств и методов, направленные на планирование, контроль и анализ воспитательного и образовательного процесса, которые нацелены на повышение эффективности обучения учащихся [11].

При выборе методики нужно понимать, что преподавание в классах гуманитарного направления и информационно-технологического профиля различается. Преподавание информатики в классах гуманитарного профиля имеет свои особенности, и сложно оно именно потому, что основные

познавательные интересы учащихся находятся в иных областях знания - истории, литературы, общественных наук и т.д. Познавательный интерес в процессе преподавания информатики в классах гуманитарного профиля следует рассматривать как внешний стимул процесса обучения, как средство активизации познавательной деятельности учащихся. Таким образом, в процессе обучения информатике в таких классах преимущественное внимание следует уделять эмоциональной стороне интереса учащихся.

В классах информационно-технологического профиля одна из целей научить учащихся правильно формулировать задачи, связанные с реальной ситуацией. Согласно содержательно-методическому критерию, в классах технического направления акцентируется внимание на решении задач прикладного характера [5].

Методика обучения должна быть направлена на формирование умений моделировать реальные процессы, развитие графических умений, образного компонента мышления, усиление. Общим в обеих методиках является необходимость в формировании умений моделировать реальные процессы. Так как развитие информационной цифровой образовательной среды, появление новых форм обучения позволяет сделать более эффективным процесс обучения [12].

Создание собственной игры, при котором глубоко осваивается язык программирования и компетенции в области программирования, математики.

Интерес, полученный в процессе создания игры, является главным мотивационным основанием для того, чтобы уделять самостоятельному обучению больше времени [19].

Данная методика работает по принципу разнообразных психологических восприятий программирования, как отдельного предмета в области информатики. Модель заключается на переживании положительных эмоций в момент выполнения задания, состоящее из

отдельных этапов, закончив которые, учащийся видит результат. Этот показатель, как правило, очень высок.

Методика использует автономность обучаемого, так как предоставляется реальная свободная и возможность действовать по своему усмотрению в планировании и в определении ее выполнения [15].

Отрицательная составляющая заключается в том, что порог вхождения в данную модель большой, так как прежде, чем приступить к разработке игр, нужно изучить базовые основы языка программирования.

Следующая методика – применения робототехнических устройств в обучении программированию.

Инженерное творчество, включает в себя несколько ключевых составляющих и лишается отрицательного эффекта высокого порога, как при игровой методике. Проектирование, конструирование, программирование.

Модуль проектирование и конструирование, в котором изучаются базовые механические принципы технических решений. Изучение данного материала развивает моторику и творческий потенциал обучающегося, что говорит о развитии, как Soft Skills и Hard Skills навыков [15].

Модуль программирования основывается на написании линейных алгоритмических структур. Любой робототехнический проект выступает в роли исполнителя алгоритма [16].

Для каждого возраста существует своя среда и язык программирования, что позволяет легко усваивать материал любому желающему. В данной методике эффективным методом является создание собственных проектов с публичной защитой и участием в различных соревнованиях.

Преимущества заключаются в том, что организовывается деятельность учащегося, соблюдая баланс между теорией и практикой. Реализуются идеи, погружая в учебно-профессиональную деятельность, повышая тем самым уровень мотивации. Так же воспитываются качества

работы в команде, а также ораторство, что в дальнейшем позволяет гораздо проще внедряться в рабочие коллективы.

Данная методика среди ряда положительных оттенков имеет и негативные. Творческий потенциал развивается максимально эффективно, но программирование, в случае с игровым подходом, отстает.

Во-первых, основная среда – визуальная, что сразу же ограничивает понимание концепции языков программирования.

Во-вторых, максимально сжатый инструментарий языка, используются малое количество основных конструкций. После программирования роботехнических проектов любая задача по программированию дается очень тяжело, так как понимание и концепция языка раскрыто недостаточно.

Методика обучения на основе микроконтроллеров с использованием технологии «Перевернутый класс», является эффективным звеном, так как объединяют все необходимые положительные качества вышеописанных методик.

Микроконтроллер в цепочке знаний по программированию служит инструментом, который визуально демонстрирует результат [13]. Данный инструмент изучения концепции языков программирования становится эффективнее, так как рассматриваются все стороны изучаемого предмета.

Создание игровой консоли, в которой программируются игры, рассматриваются алгоритмы и математика, встречающиеся при решении олимпиадных и аттестационных задачах. Следующее – это технология интернет вещей, в которой встречается автоматное программирование и проектная деятельность. А также электроника и принципы работы ЭВМ на разных уровнях абстракции [10].

Данные этапы развивают у учащихся понимание программирования, как не что-то сложное и непонятное, а как способ проектирование и создание информационных комплексов.

После изучения, обучающимся без труда даются последующие знания предмета, а также решения различных сложностей задач.

Дополнительно, создавая собственные проекты и защищая их, дети получают возможность участвовать в практико-ориентированных деятельности, связанные с предметом, результат которых является квоты на поступление в высшие заведения на инженерные направленности.

В рамках национального проекта «Образование», который должен быть реализован до 2024 года, особое внимание следует уделить масштабированию лучших практик по развитию навыков цифровой экономики, а именно программированию, использованию актуальных цифровых технологий в процессе обучения и саморазвития и т.п., через образовательные программы всех уровней, в том числе программы внеурочной деятельности [9].

Для достижения поставленной цели необходимо повысить уровень подготовленности выпускников школ к получению дальнейшего образования, можно к этому прийти посредством обучения программированию.

У обучающегося программированию, развиваются сразу несколько типов мышления, так же навыки необходимые для дальнейшей жизни в социуме [14]:

Развивается логическое мышление. Так как программирование – это тренировка мозга. Обучающийся начинает понимать последовательность работы программы и тренируется следовать этой очередности [7].

Развивается творческое мышление. Обучающийся задействует свою фантазию и воображение, реализует придуманное. При выполнении заданий дети видят результат своего труда.

Формируются навыки решения проблем. На занятиях программированием проблем не избежать – у обучающегося будут ошибки, и ему придется постоянно что-то переделывать, чтобы программа работала

правильно. Он начинает легче воспринимать подобные трудности и проще их преодолевать.

Ребенок учится учиться. Ученики идут по программе, выполняют задания, готовят итоговый проект, защищают его [14].

Развиваются навыки коммуникации. В группе ученики общаются друг с другом, когда ищут ошибку, задают вопросы учителю, если исправить проблему не получается, обсуждают результаты занятий.

Без вышеперечисленных качеств невозможно подготовить выпускников школ, обладающих информационно-коммуникационными навыками [8].

Также при обучении программированию формируются многие универсальные учебные действия, предусматриваются новые виды учебной деятельности, которые в современных условиях становятся общеучебными, например, информационное моделирование, предполагающее умение грамотно пользоваться источниками информации, оценивать достоверность информации, правильно организовать информационный процесс, оценивать информационную безопасность [6].

1.3 Методика «Перевернутый класс» в изучении программирования в школе.

В наши дни образование и общество находятся в период перехода в информационный мир, диктуя новые цели, которые предполагают формирование активной личности, стремящуюся к саморазвитию, самостоятельному поиску информации и способов организации своей деятельности [4].

Результатом обучения школьника зависит от методов, приемов и средств, которые использует преподаватель на занятиях.

Изначально, до того, как технологии стали внедряться в нашу жизнь, количество источников для поиска информации было ограничено, в связи с

этим и не было места для самостоятельного получения знаний и умений в процессе активной мыслительной и практической деятельности.

Однако, ситуация кардинально меняется в последние годы, и среди современных следует отметить использование элементов смешанного обучения, в частности технологию «Перевернутый класс» [3].

Технология «Перевернутый класс» это особая модель совместной педагогической деятельности по проектированию, организации и проведению учебного процесса, при которой обучающиеся сначала самостоятельно знакомятся с материалами урока, а потом на занятиях под руководством педагога выполняют практические задания, закрепляя материал и учась использовать новые знания на практике. В технологии «Перевернутый класс» имеется ряд преимуществ:

- рациональное использование времени;
- повышение мотивации;
- возможность изучать новый материал в индивидуальном темпе;
- материалы доступны всем;
- обучающиеся полностью вовлекаются в деятельность во время

практических занятий.

Включение данной технологии в образовательный процесс меняет несколько процессов. В особенности влияние оказывается на цели, задачи занятия, а также методы и средства, формы и содержания, функции взаимодействия преподавателя и обучающихся.

Если при изучении программирования в рамках внеурочной деятельности и объяснения теоретического материала вынести за рамки занятий, то вероятность получения и освоение практического программирование увеличиться в разы.

Наибольший эффект удастся достичь использование авторских учебных материалов, ориентированные на методiku конкретного педагога.

Организовать информационную систему для хранения ресурсов и взаимодействия педагога с обучающимися возможно с применением технологий WEB или платформы Moodle.

При изучении программирования основным видом деятельности обучающегося является работа с отладочной платой, благодаря которой изучается несколько видов программирования. На этом этапе преподаватель также отвечает на возникающие во время самостоятельного обучения вопросы. Таким образом, существенно возрастает эффективность использования внеурочного времени.

Экспериментальное внедрение технологии «Перевернутый класс» в общеобразовательной организации показало высокую степень готовности. Занимающиеся дети утверждали, что и сами используют обучающее видео для ликвидации пробелов в знаниях, возможность изучать таким образом новый материал по заранее объявленной теме показалась им удобной. Развитие умений самостоятельной работы является практической основной целью обучения. Ведь самостоятельность – одно из главных качеств, обучающихся программированию и вообще IT-технологий. Очевидно, что чем выше уровень самостоятельности учеников, тем эффективнее их продвижение в учебной деятельности.

Блок обучения программированию рассчитан на 34 часа. Занятия проводятся в группах по 10 человек. В группы набираются как высокомотивированные успевающие, так и обучающиеся со слабой мотивацией и имеющие проблемы с обучением. В контрольной группе обучение велось по традиционной схеме. Знания дети получали, слушая, конспектируя и, после всего, приступая к практическим занятиям. Контроль знаний осуществляется в виде контрольных работ и опросов.

Экспериментальная группа получала доступ к видеороликам, источникам информации. При этом обучающимся была предоставлена возможность самостоятельно оценить степень готовности к уроку. Многие из обучающихся признавались, что им пришлось пересматривать материал

два и более раз для успешного освоения. Промежуточный и итоговый контроль показали существенный рост навыков программирования.

Технология «Перевернутый класс» позволяет эффективно выстраивать индивидуальную траекторию, формировать личность, способную к самообразованию и самостоятельную оценку своих достижений.

Вывод по Главе 1

Были рассмотрены особенности изучения программирования в школе: примерная рабочая программа, основные российские образовательные языки программирования.

Были рассмотрены различные методики обучения программированию.

Под методикой обучения понимается совокупность элементов образовательных взаимосвязанных форм, средств и методов, направленные на планирование, контроль и анализ воспитательного и образовательного процесса, которые нацелены на повышение эффективности обучения учащихся.

Чтобы достичь всех поставленных результатов будет разработан курс по обучению программированию на основе микроэлектроники с использованием метода «Перевернутый класс».

ГЛАВА 2. ФАКУЛЬТАТИВНЫЙ КУРС «ПРОГРАММИРОВАНИЕ МИКРОКОНТРОЛЛЕРОВ»

2.1 Особенности обучению программированию с использованием микроконтроллеров

В век повсеместного внедрения цифровых технологий в разные сферы жизни, необходимо иметь самостоятельность, умение ставить учебные цели и пути их исполнения. Также необходимо работать с разными источниками информации, выстраивая собственное мнение. Современный человек должен ориентироваться в информационной среде и непрерывно обучаться. Для воспитания этих качеств могут послужить микроконтроллеры.

Микроконтроллер – это микросхема, функция которой является управление различными устройствами.

В процессе обучения на микроконтроллерах, затрагиваются основы программирования и электроники, а также создания собственных проектов.

Изучаются парадигмы и стили программирования:

- императивная (прототипный, процедурный, функциональный);
- автоматное.

Знания закрепляются на созданной отладочной плате, в основе которой имеется микроконтроллер ATmega328. На плате установлен oled-дисплей с разрешением 128x64 пикселей. Учащиеся программируют полноценную игровую консоль с меню, настройками и играми. На этапах создания рассматриваются разные алгоритмы, концепции программирования. Далее, учащиеся переходят на продвинутый контроллер esp8266-12F, который имеет встроенный Wi-Fi модуль, который позволяет осуществлять клиент-серверное взаимодействие на языке программирования JavaScript. Также рассматривается парадигма декларативности на HTML с кастомизацией CSS.

Следующим этапом обучения является электроника и автоматное программирования. Возвращаемся к отладочной плате, на которой

дополнительно установлены различные модули для создания технологии «интернет вещей» [2].

2.2 Программа внеурочной деятельности «Программирование микроконтроллеров»

В стандарте основного общего образования по информатике и икт прописаны требования к уровню подготовки выпускников [17], среди внушительного списка требований, стоит обратить внимание на:

1. Знать: основные свойства алгоритма, типы алгоритмических конструкций: следование, ветвление, цикл; понятие вспомогательного алгоритма.

2. Уметь: выполнять базовые операции над объектами: цепочками символов, числами, списками, деревьями; проверять свойства этих объектов; выполнять и строить простые алгоритмы.

Это далеко не полный перечень требований к выпускнику. На изучение раздела программирование выделяется не так много часов, чтобы достигнуть необходимых результатов, поэтому на основе данного анализа был разработан курс внеурочной деятельности «Программирование микроконтроллеров», предназначенный для обучающихся средних классов общеобразовательных школ. Общее количество часов – 34.

Цели курса:

- формирование у обучающихся представление об языке программирования C++;
- изучение основ алгоритмов и структур данных;
- изучение основ электроники и микроконтроллеров.

Основные задачи:

- изучить основные операторы, алгоритмические конструкции;
- научиться разрабатывать алгоритмы;
- разобрать олимпиадное и автоматное программирование;

- научиться собирать электронные схемы из полупроводников и интегральных схем;

- научиться программировать микроконтроллер;
- научиться проектировать, трассировать печатные платы;
- создавать научно-исследовательские проекты.

В ходе изучения курса будут расширены и углублены знания учащихся, приобретенные на базовом уровне.

Данный курс способствует развитию интеллектуальных способностей, логического и абстрактного мышления.

Образовательные результаты:

Учащиеся должны знать:

- достоинства и недостатки языка программирования;
- основные алгоритмические конструкции и операторы;
- алгоритмы;
- основы электроники (ток, напряжение, сопротивление, полупроводники, логические схемы);
- основы проектирования и моделирования (Программа Autodesk Eagle, Fusion 360).

Учащиеся должны уметь:

- применять алгоритмические конструкции при решении различных задач (учебных, алгоритмических);
- создавать простейшие электронные схемы;
- разрабатывать собственные печатные платы;
- создавать автоматизированные комплексы на основе микроконтроллеров;
- разрабатывать информационные системы для устройств. (Javascript, noSQL, MySQL, PHP).

Базовый уровень формирует у учащихся необходимые знания о языке программирования C++ и о его взаимодействии с вычислительными системами.

Учебный курс «Программирование микроконтроллеров» предназначен для изучения в старших классах. Курс рассчитан на 34 часа. Тематическое планирование представлено в таблице 2.

В ходе изучения курса будут расширены и углублены знания учащихся, приобретённые на базовом уровне изучения материала.

Теоретические вопросы, рассматриваемые в рамках курса:

1. Что такое информация.
2. Язык программирования C++.
3. Цифровой и аналоговый тип сигнала.
4. Основы электроники.
5. Таймер микроконтроллера.
6. АЦП (Аналогово-цифровой преобразователь) и ЦАП (Цифро-аналоговый преобразователь).
7. ШИМ (Широтно-импульсовая модуляция).
8. Компилятор.
9. Память.
10. Типы данных.
11. Данные (интерфейс, порты ввода\вывода).
12. Логические уровни.
13. Процедуры и функции.
14. Условный и тернарный оператор.
15. Оператор выбора.
16. Структуры данных (массив, строки).
17. Основные алгоритмы, циклы.
18. Виды микроконтроллеров.
19. Клиент-серверная разработка.
20. Язык программирования JavaScript.

21. MySQL.

Практические аспекты, рассматриваемые в рамках курса:

1. Установка и подключение среды программирования Arduino IDE.
2. Написание программы «светофор».
3. Написание программы «Бегущие огоньки».
4. Написание алгоритмов обработки нажатия клавиш.
5. Автоматное программирование.
6. Программирование OLED дисплея.
7. Разработка программного меню, с применением вышеописанных алгоритмов.
8. Разработка игры «Змейка».
9. Разработка умного дома, работа с библиотеками.
10. Создание собственных проектов.

Таблица 2 – Тематическое планирование.

Тема	Всего часов	Теория	Практика
1	2	3	4
Подключение надстройки среды программирования микроконтроллера Arduino IDE	1	0	1
Написание программы для проекта «светофор»	2	1	1
Написание программы для проекта «Бегущие огоньки»	3	1	2
Написание алгоритмов обработки данных	3	1	2
Создание простого конечного автомата	3	1	2
Графическое представление информации	3	1	2
Программирование отладочной платы: Разработка системного меню	5	2	3

Продолжение таблицы 2

1	2	3	4
Программирование отладочной платы: Разработка игрового модуля	6	2	4
Программирование отладочной платы: Разработка IoT-модуля	8	3	5

Содержание курса:

Подключение надстройки среды программирования микроконтроллера Arduino IDE (*Скачивание и установка программы Arduino IDE. Установка драйверов для микроконтроллера, подключение и прошивка программы Blink*).

Написание программы «светофор» (*Изучение понятий цифрового и аналогового сигнала. Составление простейшей программы на условных операторах с использованием директив, изучение портов микроконтроллера, команд digitalWrite(), pinMode()*).

Написание программы «Бегущие огоньки» (*Изучение типов данных, процедур, функций*).

Написание алгоритмов обработки нажатия клавиш (*Изучение понятия таймер, работа с таймером, написание процедур обработки нажатия кнопок*).

Создание простого конечного автомата (*Изучение понятия детерминированного конечно автомата, знакомство со SWITCH-технологией*).

Программирование OLED-дисплея (*Знакомство с визуальным редактором, преобразование изображение в bitmap. Работа с библиотекой Adafruit_GFX.h, Adafruit_SSD1306.h. Загрузка изображения на экран дисплея*).

Разработка программного меню (*Создание динамического меню, управляющее с кнопок*).

Разработка игры «Змейка» (*Изучение оператора цикла с параметром и предусловием, изучение структур данных и создание двумерного рабочего*

пространства, программирование игры змейка, применяя пройденные алгоритмы и структуры).

Разработка умного дома, работа с библиотеками и датчиками (Изучение понятия датчик, работа с библиотеками, изучение клиент-серверного взаимодействия. Изучение нового языка программирования JavaScript. MySQL).

Изучение схемотехники. 3Д моделирование (Изучение основ схемотехники, моделирование в компас 3Д).

Создание собственных проектов (Создание собственных тематических автоматизированных комплексов, проектирование платы, моделирование оболочки, программирование).

Поурочное планирование

Урок № 1

Тема: Установка и настройка среды программирования Arduino IDE.
Знакомство с интерфейсом.

Тип урока: комбинированный.

Цели урока:

знать:

- понятие установщик программного обеспечения;
- файловая система операционной системы;
- системные характеристики;

уметь:

- устанавливать программное обеспечение;
- осуществлять поиск информации в интернете;
- устанавливать драйвера для стороннего оборудования.

Краткое содержание урока:

Скачивание и установка актуальной версии программы Arduino IDE.
Установка драйверов для микроконтроллера. Разбор интерфейса программной оболочки, рассмотрение стандартных библиотек. Прошивка

платы встроенным примером раздела «01.Basic -> Blink». Проверка работоспособности.

Контрольные вопросы:

1. Что нужно проверить, если при прошивке контроллера выходит ошибка загрузки?
2. Где находятся встроенные примеры библиотек?
3. Что такое загрузчик?

Урок № 2-3

Тема: Написание программы для проекта «Светофор».

Тип урока: комбинированный.

Цели урока:

знать:

- о существовании языка программирования C++;
- базовые операторы и типы данных языка;
- процедурное программирование;

уметь:

- программировать контроллер и загружать программа через программатор;
- разбивать задачу на подзадачи.

Краткое содержание урока:

Сборка проекта по схеме на макетной плате.

Программирование, изучая и закрепляя условные операторы (if...else, switch). Типы данных. Процедуры.

Задача. Написать программу для светофора с разными режимами работы. Режим работы зависит от времени суток.

Контрольные вопросы:

1. Что такое программирование?
2. Какие операторы использовались и чем они отличаются?
3. Что такое процедуры и в каких условиях используются?

Урок № 4-6

Тема: Написание программы для проекта «Бегущие огоньки».

Тип урока: комбинированный.

Цели урока:

знать:

- что такое циклы, области видимости;
- структуры данных;

уметь:

- применять цикличность при решении задач;
- оптимизировать задача с использованием структур данных.

Краткое содержание урока:

Сборка платы по схеме.

Изучение циклов (do - while, while, for), операторов break, continue, массивы.

Применение изученного материала на программирование бегущих огоньков, используя светодиоды.

Задача: Запрограммировать несколько разных световых эффектов, которые переключаются друг за другом.

Контрольные вопросы:

1. Виды циклов?
2. Чем отличаются цикл while от For?
3. Области видимости в циклах?

Урок № 7-9

Тема: Написание алгоритмов обработки данных.

Тип урока: комбинированный.

Цели урока:

знать:

- особенности аналогового и цифрового сигнала;
- алгоритм сортировки массива;

- виды фильтров обработки полученных данных с периферии микроконтроллера;

уметь:

- отправлять и принимать сигнал на микроконтроллере;
- обрабатывать и фильтровать данные;
- сортировать данные в массиве.

Краткое содержание урока:

Работа с аналоговыми датчиками и обработка полученных данных.

Изучение алгоритма сортировки массива методом пузырька, бинарный поиск.

Применение фильтров:

- Медианный.
- Скользящего среднего.

Контрольные вопросы:

1. Для чего нужны алгоритмы и фильтры, работая с множеством данных?
2. Что такое сигнал и какие типы бывают?

Урок № 10-12

Тема: Создание простого конечного автомата.

Тип урока: комбинированный.

Цели урока:

знать:

- что такое конечный автомат;
- что такое функция, рекурсия;

уметь:

- применять автоматное программирование при решении задач;
- применять функции и рекурсивность при обработки данных, полученные микроконтроллером.

Краткое содержание урока:

Установка и пайка кнопок на отладочную плату.

Написание программы для обработки различного нажатия, используя модель «конечный автомат»; знакомство с функциональным программированием.

Задача 1. Написать программу для обработки одинарного, двойного и длительного (3 секунды) нажатия двух кнопок микроконтроллера.

Контрольные вопросы:

1. Что такое абстрактный автомат?
2. Для чего нужны функции и отличие от процедур?

Урок № 13-15

Тема: Графическое представление информации.

Тип урока: комбинированный.

знать:

- интерфейсы передачи данных;
- тип изображения ВІМАР;
- работа с библиотеками, техническая документация;

уметь:

- устанавливать и подключать библиотеки в проект;
- конвертировать изображение из растрового или векторного форматов в ВІТМАР.

Краткое содержание урока:

Установка и пайка на плату дисплея OLED 128x64 с интерфейсом I2C.

Работа с изображением формата bitmap.

Создание бинарной матрицы и вывод отображения на экран, используя библиотеку Adafruit_SSD1306.

Контрольные вопросы:

1. Что такое bitmap?
2. Виды дисплеев?
3. Как закодировать изображение в шестнадцатеричной системе?

4. Интерфейс подключения экрана?
5. Как отрисовывается картина на дисплее?

Урок № 16-20

Тема: Программирование отладочной платы: Разработка системного меню.

Тип урока: комбинированный.

Цели урока:

знать:

- виды графики;
- древовидное представление данных;
- прерывание;

уметь:

- использовать в комплексе ранее изученный материал;
- программировать многоуровневую модель приложения.

Краткое содержание урока:

Рисование в растровом редакторе все интерпретации системного меню для дальнейшей разработки игрового и IoT модулей.

Конвертация изображения в BITMAP.

Написание программы для отображения меню и применение модели «конечный автомат» для обработки нажатий кнопок и перехода внутри меню.

Контрольные вопросы:

1. Каким образом экран переводит шестнадцатеричный формат изображения в картинку на экране?
2. Каким образом работает планировщик?
3. Чем аппаратное прерывание отличается от программного?

Урок № 21-26

Тема: Программирование отладочной платы: Разработка игрового модуля.

Тип урока: комбинированный.

Цели урока:

знать:

- что такое перечисление;
- что такое композитный тип данных;
- алгоритм работы с массивом «побитовый сдвиг»;
- что такое двумерный массив;
- для чего нужны директивы;

уметь:

- составлять алгоритм написания сценария приложений в виде блок-схем;
- разделять программу на процедуры и функции;
- использовать полученные знания на практике решения задач сложного типа.

Краткое содержание урока:

Проектирование, программирование и внедрение в микроконтроллер игры «Змейка» и «Крестики и нолики», используя новые знания:

- перечисление;
- композитный тип данных;
- директивы;
- побитовый сдвиг;
- двумерные массивы.

Контрольные вопросы:

1. Чем директивы отличаются от переменных?
2. Какой наиболее подходящий цикл использовать для работы с двумерными массивами?
3. Для чего используется композитный тип данных `struct`?

Урок № 27-34

Тема: Программирование отладочной платы: Разработка IoT-модуля.

Тип урока: комбинированный.

Цели урока:

знать:

- что такое делитель напряжения, сила тока;
- что такое Bluetooth и как настроено сопряжение между master'ом и slave'ом;

- что такое пирозлектрический датчик;

уметь:

- программировать большое количество модулей;
- правильно составлять схему взаимодействия микроконтроллера с датчиками;
- настроить взаимодействия микроконтроллера с окружающей средой.

Краткое содержание урока:

Написание программы для умного дома с применением IoT-технологии.

Контрольные вопросы:

1. Что такое «умный дом»?
2. Каким образом считываются данные с аналогового датчика?

Необходимые компоненты для курса:

- HC-06;
- DHT12;
- OLED;
- cd4026, сдвиговый регистр;
- 7-ми сегментный индикатор с общим катодом;
- пьезодинамик (buzzer);
- потенциометр;
- датчик присутствия (PIR);
- фоторезистор;
- Ардуино НАНО V3 (238 чип);

Выводы по Главе 2

Был разработан курс «Обучение программированию на основе микроэлектроники», определены структурные элементы методики обучения, представляющие собой совокупность взаимосвязанных компонентов: цели, содержание, методы, формы и средства обучения, необходимые для создания целенаправленного и строго определенного воздействия на формирование компетентной личности с заданными качествами.

Выявлены особенности преподавания информатики. Особое внимание уделяется на развитие самостоятельности на основе технологии «Перевернутый класс».

Данный метод помогает обучающимся учиться самостоятельно принимать решения, наращивать потенциал саморазвития, так как формируется умение самостоятельного поиска и усвоения знаний, убедительный отчет проделанной работы.

ГЛАВА 3. ПЕДАГОГИЧЕСКИЙ ЭКСПЕРИМЕНТ И ОЦЕНКА ЭФФЕКТИВНОСТИ ВВЕДЕНИЯ ФАКУЛЬТАТИВНОГО КУРСА «ПРОГРАММИРОВАНИЕ»

3.1 Организация и проведение педагогического эксперимента

Педагогический эксперимент – это научно обоснованная и хорошо продуманная система организации педагогического процесса, направленная на открытие нового педагогического знания, проверки и обоснования заранее разработанных научных предположений, гипотез [12]. Для того чтобы современное педагогическое исследование было успешным, необходимо применение таких исследовательских методов, которые обеспечивают проведение анализа состояния исследуемой проблемы в массовой практике и получение достоверного фактического материала на отдельных этапах.

В основе планирования педагогического эксперимента лежит подготовительная работа, в ходе которой были определены методы исследования, объекты измерения, выбор измерителей.

Главная цель постановки эксперимента определялась как разработка, обоснование и апробация программы внеурочной деятельности «Обучения программированию на основе микроэлектроники».

Основной базой для проведения педагогического эксперимента была выбрана МБОУ «Гимназия №48 г. Челябинска» с углубленным изучением французского языка.

Для проверки успешности реализации целевых ориентиров был выбран такой индикативный показатель, как уровень мотивации. Гимназия гуманитарная, у обучающихся разных классов в целом отсутствует мотивация к обучению математики и информатики. В соответствии с этим сформулированы задачи:

1. Опираясь на методику Т.Д. Дубовицкой «Диагностика направленности мотивации изучения предмета» собрать данные об уровне

мотивации обучающихся до проведения занятий факультативного курса и после него.

2. Провести апробацию курса и провести экспериментальную проверку правдоподобности гипотезы исследования методами математической статистики.

Этапы исследования. Исследование проводилось в три этапа в течение 2017-2020 гг.

На первом этапе (2017-2018 гг.) осуществлялся анализ учебно-методической литературы и нормативных документов в области обучения информатике; изучался опыт в области разработки вариативной части курса информатики, анализировались существующие подходы к проектированию компонента образовательного учреждения; обосновывались и формулировались принципы формирования курса внеурочной деятельности «Обучения программированию на основе микроэлектроники».

На втором этапе (2018-2019 гг.); разрабатывались электронная модель содержания курса внеурочной деятельности «Обучения программированию на основе микроэлектроники», рабочая программа курса, подбирались и создавались практико-ориентированные практические задания.

На третьем этапе (2019 г.) осуществлялась экспериментальная проверка правдоподобности гипотезы исследования методами математической статистики; формулировались выводы; оформлялось диссертационное исследование.

3.2 Анализ результатов тестирования по методике «Диагностика направленности мотивации изучения предмета» у обучающихся в МБОУ «Гимназия № 48 г. Челябинска»

Рассмотрим результаты тестирования обучающихся МБОУ «Гимназия 48 г. Челябинска» по методике «Диагностика направленности мотивации изучения предмета».

Инструкция к проведению опроса:

Вам предлагается принять участие в исследовании, направленном на повышение эффективности обучения. Прочитайте каждое высказывание и выразите свое отношение к изучаемому предмету, проставив напротив номера высказывания свой ответ, используя для этого следующие обозначения:

«верно» ++;

«пожалуй, верно» +;

«пожалуй, неверно» –;

«неверно» – –.

Помните, что качество наших рекомендаций будет зависеть от искренности и точности Ваших ответов.

1. Изучение данного предмета даст мне возможность узнать много важного для себя, проявить свои способности.
2. Изучаемый предмет мне интересен, и я хочу знать по данному предмету как можно больше.
3. В изучении данного предмета мне достаточно тех знаний, которые я получаю на занятиях.
4. Учебные задания по данному предмету мне неинтересны, я их выполняю, потому что этого требует учитель (преподаватель).
5. Трудности, возникающие при изучении данного предмета, делают его для меня еще более увлекательным.
6. При изучении данного предмета кроме учебников и рекомендованной литературы самостоятельно читаю дополнительную литературу.
7. Считаю, что трудные теоретические вопросы по данному предмету можно было бы не изучать.
8. Если что-то не получается по данному предмету, стараюсь разобраться и дойти до сути.

9. На занятиях по данному предмету у меня часто бывает такое состояние, когда «совсем не хочется учиться».

10. Активно работаю и выполняю задания только под контролем учителя (преподавателя).

11. Материал, изучаемый по данному предмету, с интересом обсуждаю в свободное время (на перемене, дома) со своими одноклассниками (друзьями).

12. Стараюсь самостоятельно выполнять задания по данному предмету, не люблю, когда мне подсказывают и помогают.

13. По возможности стараюсь списать у товарищей или прошу кого-то выполнить задание за меня.

14. Считаю, что все знания по данному предмету являются ценными и по возможности нужно знать по данному предмету как можно больше.

15. Оценка по этому предмету для меня важнее, чем знания.

16. Если я плохо подготовлен к уроку, то особо не расстраиваюсь и не переживаю.

17. Мои интересы и увлечения в свободное время связаны с данным предметом.

18. Данный предмет дается мне с трудом, и мне приходится заставлять себя выполнять учебные задания.

19. Если по болезни (или другим причинам) я пропускаю уроки по данному предмету, то меня это огорчает.

20. Если бы было можно, то я исключил бы данный предмет из расписания (учебного плана).

Ключ:

Подсчет показателей опросника производится в соответствии с ключом, где «Да» означает положительные ответы («верно»; «пожалуй, верно»), а «Нет» — отрицательные («пожалуй, неверно»; «неверно»).

«Да» - 1, 2, 5, 6, 8, 11, 12, 14, 17, 19

«Нет» - 3, 4, 7, 9, 10, 13, 15, 16, 18, 20

За каждое совпадение с ключом начисляется один балл.

0-10 – внешняя мотивация.

11-20 – внутренняя мотивация.

Для проверки достоверности различий по годам применим Т-критерий Вилкоксона. Критерий применяется для сопоставления показателей, измеренных в двух разных условиях на одной и той же выборке испытуемых.

Он позволяет установить не только направленность изменений, но и их выраженность. С его помощью мы определяем, является ли сдвиг показателей в каком-то одном направлении более интенсивным, чем в другом. Сформулируем гипотезы:

H_0 : Тенденция изменения уровня мотивации не является случайной, различия наблюдаются.

H_1 : Тенденция изменения значений показателя самоопределения является случайной, различия не наблюдаются.

Данные входного и выходного тестирований обучающихся представлены в таблице 3.

Таблица 3- Данные входного и выходного тестирований обучающихся

Ученики	Данные входного тестирования	Данные выходного тестирования
1	2	3
Ученик 1	10	14
Ученик 2	11	14
Ученик 3	16	20
Ученик 4	14	17
Ученик 5	12	15
Ученик 6	7	10
Ученик 7	9	12
Ученик 8	8	9
Ученик 9	6	16

Продолжение таблицы 3

1	2	3
Ученик 10	7	10
Ученик 11	5	8
Ученик 12	7	10
Ученик 13	12	15
Ученик 14	7	10
Ученик 15	9	12
Ученик 16	8	11
Ученик 17	7	11
Ученик 18	5	8
Ученик 19	7	10
Ученик 20	11	9

В ходе расчёта была составлена Excel-таблица с промежуточными вычислениями (Таблица 4).

Таблица 4 – Промежуточные вычисления. Критерий Вилкоксона

N	"До"	"После"	Сдвиг ($t_{\text{после}} - t_{\text{до}}$)	Абсолютное значение сдвига	Ранговый номер сдвига
1	2	3	4	5	6
1	10	14	4	4	18
2	11	14	3	3	9,5
3	16	20	4	4	18
4	14	17	3	3	9,5
5	12	15	3	3	9,5
6	7	10	3	3	9,5
7	9	12	3	3	9,5
8	8	9	1	1	1
9	6	16	10	10	20
10	7	10	3	3	9,5
11	5	8	3	3	9,5
12	7	10	3	3	9,5
13	12	15	3	3	9,5
14	7	10	3	3	9,5
15	9	12	3	3	9,5
16	8	11	3	3	9,5
17	7	11	4	4	18
18	5	8	3	3	9,5
19	7	10	3	3	9,5

Продолжение таблицы 4

1	2	3	4	5	6
20	11	9	-2	2	2
Сумма рангов нетипичных сдвигов:					2

Суммируем все нетипичные сдвиги и получаем показатель $T_{эмп} = 2$.

Критические значения $T_{кр}$ при $n=20$ показаны в таблице 5.

Таблица 5 – Критические значения $T_{эмп}$

n	$T_{кр}$	
	0.01	0.05
20	43	69

Ось значимости для данного анализа выглядит, как показано на рисунке 1. Необходимо отметить $T_{эмп}$ и убедиться, что данная величина попадает в зону значимости.

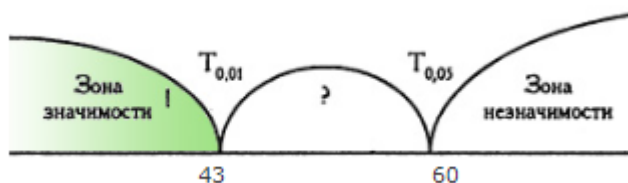


Рисунок 1 – Ось значимости

Таким образом: тенденция изменения уровня мотивации не является случайной, различия наблюдаются, гипотеза H_0 доказана.

Выводы по Главе 3

Экспериментальная работа проводилась в МБОУ «Гимназия №48 г. Челябинска» в три этапа с 2017 по 2019.

Был измерен уровень мотивации обучающихся до педагогического эксперимента и после. Анализ результатов проводился по критерию Вилкоксона.

Была подтверждена гипотеза о том, что повышение уровня мотивации к изучению информатики у обучающихся в результате освоения курса «Обучения программированию на основе микроэлектроники» является значимым.

Результаты педагогического эксперимента позволили сделать вывод, что обучение программированию на основе микроэлектроники с использованием метода «Перевернутый класс», в классах лингво-гуманитарного профиля влияет на уровень мотивации.

ЗАКЛЮЧЕНИЕ

В ходе проведения педагогического эксперимента были рассмотрены и проанализированы различные методики преподавания программирования в школе.

Выявлены особенности преподавания информатики в лингво-гуманитарных классах. Они связаны с тем, что познавательные интересы учащихся находятся в иных областях знания – истории, литературы, общественных наук и т.д. Развить познавательный интерес можно с помощью внеурочной деятельности на факультативных занятиях. Особенность таких занятий состоит в том, что они должны быть увлекательными и вызывать интерес у учащихся. Для этого применяется технология «Перевернутый класс».

В работе предложено содержание факультативного курса «Обучения программированию на основе микроэлектроники» а также методы, средства и организационные формы обучения.

Проведен педагогический эксперимент для проверки гипотезы исследования, результаты которого позволяют подтвердить эффективность разработанной методики обучения.

Результаты эксперимента подтверждают гипотезу исследования: если осуществлять обучение программированию на основе микроэлектроники с использованием метода «Перевернутый класс», в классах лингво-гуманитарного профиля, то уровень мотивации к изучению информатики у обучающихся повысится.

СПИСОК ИСПОЛЬЗОВАННЫХ ИСТОЧНИКОВ

1. Аркадьев А. Г. Сборник нормативных документов. Информатика и ИКТ: учеб. пособие / А. Г. Аркадьев – Москва : Дрофа, 2008. – 112 с. ISBN 978-5-358-02384-0.
2. Газейкина А. И. Стили мышления и обучение программированию студентов педагогического вуза [сайт] – 2014. - URL: <http://ito.edu.ru/2013/Moscow/I/1/I-1-6371.html> (дата обращения 12.11.2018)
3. Гейн А. Г. Ожидания информатики / А. Г. Гейн // Информатика в школе: прошлое, настоящее будущее. – 2014. – № 3 (71) – С. 5–10.
4. Гиппенрейтер Ю. Б., Петухова, В. В. Психология мышления. Измestьев Н. С. Объектно-ориентированное программирование как средство формирования алгоритмического мышления старших школьников // Перспективы и вызовы информационного общества – 2015. – № 4 (50) – С. 110 – 115.
5. Дьяченко В. К. Организационная структура учебного процесса и ее развитие: учебник / В. К Дьяченко. – Москва : Академия, 2006. – 192 с. ISBN 5-7155-0146-6.
6. Жемчужников, Д.Г. Формирование мотивации школьников к изучению программирования на основе межпредметной интеграции Д. Г. Жемчужников // Вестник Российского университета дружбы народов. – 2014. – № 2 (15). – С. 46–48.
7. Измestьев Н. С., Сокольская, М. А. Особенности формирования и развития алгоритмического мышления с помощью объектно-ориентированного программирования // Материалы конференции «Современные тенденции и проекты развития информационных систем и технологий. – 2015. – № 4 (13). – С. 167–172.
8. Копаев А. В. О практическом значении алгоритмического стиля мышления / А.В. Копаев // Школа и производство. – 2003. – № 6. – С. 6–11.

9. Кузнецов А. А. Школьная информатика: вчера, сегодня, завтра / А. А. Кузнецов, Т. Б. Захарова // Информатика и образование. – 2014. – № 10. – С. 3–6.
10. Кушниренко А.Г., Лебедев, Г.В. Информатика: 12 лекций о том, для чего нужен школьный курс информатики / А.Г Кушниренко // Хрестоматия. – Москва : МГУ, 1982. – 489 с. ISBN 5-222-02404-0.
11. Лапчик М. П. Методика преподавания информатики / М. П. Лапчик и др.; под общей ред. М.П. Лапчика. Москва: – Академия, 2001. – 624 с. ISBN 5-09-000598-2.
12. Малеев В. В. Общая методика преподавания информатики: учебно-методическое пособие / В. В. Малаев. – Воронеж : ВГПУ, 2005. – 271 с. ISBN 5-88519-365-7.
13. Окулов С. М. Задачи по программированию: учебно-методическое пособие / С. М. Окулов – Москва: БИНОМ. Лаборатория знаний, 2006. – 824 с. ISBN 978-5-00101-448-5.
14. Семенов А. Л., Рудченко, Т. А., Щеглова, О. В. Основы мышления и коммуникации: книга для учителя. / А. Л. Семенов – Москва : ИНТ, Просвещение, 1999. – 76 с. ISBN 978-5-09-018715-2.
15. Слостенин В. А., Исаев И. Ф., Мищенко А. И., Педагогика / учебное пособие для студентов / В.А. Слостелин – Издательство «Школа-Пресс», 1997. ISBN 5-691-00950-8.
16. Слинкина И. Н. Использование компьютерной техники в процессе развития алгоритмического мышления у младших школьников: автореф. дис. ... канд. пед. наук: 13.00.02 / Ирина Николаевна Слинкина ; науч. рук. Т. Н. Шамало ; УрГПУ. – Екатеринбург, 2000. – 22 с.
17. Федеральный государственный образовательный стандарт основного общего образования [сайт]. – URL: 2016. – URL: <https://fgos.ru>. (дата обращения 23.10.2019).

18. Хорев П. Б. Технологии объектно-ориентированного программирования : учебник / П. Б. Хорев. – Москва.: Академия, 2012. – 448 с. ISBN 978-5-7695-8091-8.

19. Черникова Н. А. Система форм организации обучения в контексте современной методики : [сайт]. – 2013 – URL: <http://www.omsk.edu/volume/2006/methodics/> – (дата обращения 13.12.2018).