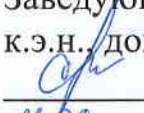



МИНИСТЕРСТВО НАУКИ И ВЫСШЕГО ОБРАЗОВАНИЯ
РОССИЙСКОЙ ФЕДЕРАЦИИ
Федеральное государственное автономное образовательное учреждение
высшего образования
«ТЮМЕНСКИЙ ГОСУДАРСТВЕННЫЙ УНИВЕРСИТЕТ»
ИНСТИТУТ МАТЕМАТИКИ И КОМПЬЮТЕРНЫХ НАУК
Кафедра алгебры и математической логики

РЕКОМЕНДОВАНО К ЗАЩИТЕ В ГЭК
Заведующий кафедрой
к.э.н., доцент
 С.В. Вершинина
 2022 г.

ВЫПУСКНАЯ КВАЛИФИКАЦИОННАЯ РАБОТА
магистерская диссертация

ПРОЕКТИРОВАНИЕ ЭЛЕКТИВНОГО КУРСА ПО ФОРМИРОВАНИЮ
УЧЕБНЫХ КОМПЕТЕНЦИЙ УЧАЩИХСЯ НА ОСНОВЕ ИНТЕГРАЦИИ
МАТЕМАТИКИ И ПРОГРАММИРОВАНИЯ НА PYTHON В СРЕДНЕЙ
ШКОЛЕ

44.04.01 Педагогическое образование
Магистерская программа «Современное математическое образование»

Выполнила работу
студентка 2 курса
очной формы обучения



Стефанько Ангелина Евгеньевна

Научный руководитель
к.н., доцент



Шармин Валентин Геннадьевич

Рецензент
к.п.н., доцент, преподаватель,
Тюменский индустриальный
университет



Осинцева Марина Александровна

Тюмень
2022

ОГЛАВЛЕНИЕ

| | |
|--|----|
| ВВЕДЕНИЕ | 4 |
| ГЛАВА 1. ТЕОРЕТИЧЕСКОЕ ОСНОВЫ ПРИМЕНЕНИЯ ИНТЕГРАЦИИ МАТЕМАТИКИ И ПРОГРАММИРОВАНИЯ НА PУТНОН ДЛЯ ПРОЕКТИРОВАНИЯ ЭЛЕКТИВНОГО КУРСА ПО ФОРМИРОВАНИЮ УЧЕБНЫХ КОМПЕТЕНЦИЙ УЧАЩИХСЯ | 10 |
| 1.1. ОПРЕДЕЛЕНИЕ ПОНЯТИЙ «КОМПЕТЕНЦИЯ» И «КОМПЕТЕНТНОСТЬ» | 10 |
| 1.2. СТРУКТУРНЫЕ КОМПОНЕНТЫ УЧЕБНЫХ КОМПЕТЕНЦИЙ ОБУЧАЮЩИХСЯ | 13 |
| 1.3. СОВРЕМЕННЫЕ ПЕДАГОГИЧЕСКИЕ ТЕХНОЛОГИИ И ПРИЕМЫ ДЛЯ ФОРМИРОВАНИЯ УЧЕБНЫХ КОМПЕТЕНЦИЙ | 17 |
| 1.4. ПРОБЛЕМА ИНТЕГРАЦИИ МАТЕМАТИКИ И ПРОГРАММИРОВАНИЯ В СРЕДНЕЙ ШКОЛЕ | 23 |
| 1.5. МОДЕЛЬ ФОРМИРОВАНИЯ УЧЕБНЫХ КОМПЕТЕНЦИЙ ОБУЧАЮЩИХСЯ НА ОСНОВЕ ИНТЕГРАЦИИ МАТЕМАТИКИ И ПРОГРАММИРОВАНИЯ | 26 |
| ВЫВОДЫ ПО ГЛАВЕ 1 | 30 |
| ГЛАВА 2. РЕАЛИЗАЦИЯ МОДЕЛИ ПРИМЕНЕНИЯ ИНТЕГРАЦИИ МАТЕМАТИКИ И ПРОГРАММИРОВАНИЯ НА PУТНОН ДЛЯ ПРОЕКТИРОВАНИЯ ЭЛЕКТИВНОГО КУРСА ПО ФОРМИРОВАНИЮ УЧЕБНЫХ КОМПЕТЕНЦИЙ УЧАЩИХСЯ | 31 |
| 2.1. МЕТОДИЧЕСКАЯ ЧАСТЬ | 34 |
| 2.2. СОДЕРЖАНИЕ ЭЛЕКТИВНОГО КУРСА | 36 |
| 2.2.1. УЧЕБНО-ТЕМАТИЧЕСКОЕ ПЛАНИРОВАНИЕ ЭЛЕКТИВНОГО КУРСА | 36 |

| | |
|---|----|
| 2.2.2. СОДЕРЖАНИЕ ТЕОРЕТИЧЕСКОЙ ЧАСТИ КУРСА | 38 |
| 2.2.3. СОДЕРЖАНИЕ ПРАКТИЧЕСКОЙ ЧАСТИ КУРСА | 52 |
| 2.2.4. СОДЕРЖАНИЕ КОНТРОЛЬНОЙ ЧАСТИ КУРСА..... | 60 |
| ВЫВОДЫ ПО ГЛАВЕ 2 | 61 |
| ЗАКЛЮЧЕНИЕ..... | 63 |
| БИБЛИОГРАФИЧЕСКИЙ СПИСОК..... | 64 |

ВВЕДЕНИЕ

За последнее время произошли значительные изменения роли и места персональных компьютеров и информационно-коммуникационных технологий (ИКТ) в обществе. Те, кто способны эффективно осваивать современные технологии и информацию имеют иной стиль мышления. На данный момент уже невозможно представить современную школу без ИКТ.

В наше время информационные технологии проникли во все сферы человеческой жизни. Поэтому важно обучать подрастающее поколение в области информатики и ИКТ. Кроме того, если человек пытается связать свою жизнь с ИКТ, то ему необходимо иметь определенные знания основ программирования для его будущей карьеры.

В связи с бурным развитием ИКТ методы преподавания информатики должны меняться с каждым годом. Однако на данный момент образовательные учреждения не уделяют этому особого внимания. В настоящее время основы программирования изучаются только на уровне введения, так как время обучения очень маленькое. В то же время олимпиады и различные соревнования дают студентам знания решения различных задач, алгоритмов и структур данных, приемов программирования. Я считаю, что наибольшего успеха в преподавании основ программирования можно добиться, начиная обучать школьников уже с 5-7 класса, т.к. в дальнейшем способность к обучению снижается. Ученики в средней школе теряют интерес к математике и математическим дисциплинам. В 11 классе когда им предоставляется выбор предметов для сдачи ЕГЭ, они больше склоняются к гуманитарным наукам. Для тех, кто выбрал для сдачи экзамена математику, целью является не получить максимальный балл, а набрать проходной. Детей учат решать задачи ЕГЭ В части и зачастую при выпуске, многие не способны даже выполнять элементарные операции с дробями.

Основной проблемой преподавания программирования в школах является отсутствие системного подхода к этой проблеме. Проблема заключается в том, что в школе не учат решать задачи с помощью программирования. Школьные уроки, по сути, сводятся к изучению языковых структур и выполнению заданий по этим структурам, но для решения проблем, которые реально возникают, даже если ребенок не станет программистом, навыки, приобретенные на занятиях по программированию, станут для него хорошим толчком в дальнейшей жизни.

На данный момент детям приходится усваивать очень большой объем информации. На данный момент детям приходится усваивать очень большой объем информации. А так как сейчас поколение Z, то у них преобладает «клиповое мышление», а это значит, что у детей снижается успеваемость по предметам, достаточно низкая концентрация внимания на какой-то определенной информации, а также неспособность к анализируванию той или иной задачи. К сожалению, все эти факторы вызывают отсутствия умения применения полученных знаний на уроках в реальной жизни.

Проблемы:

1. Ученики не могут воспринять информацию, которая дается им на слуху.
2. В математике при решении задач от учащихся требуется умение пользоваться абстрактными образами, что понижает наглядность решения данных задач.
3. При выполнении заданий на уроках математики, ребенок не всякий раз имеет возможность оценить корректность собственного решения-это делает педагог.

На сегодняшний день Python является одним из самых популярных языков. Он достаточно прост в изучении, имеет простой синтаксис и разносторонен в использовании. Так же в интернете есть очень много обучающих ресурсов для изучения языка.

Таким образом математические примеры заменяются более интересной задачей по решению множества однотипных примеров одной программой. Элективный курс по математике методом программирования на Python позволит продемонстрировать обучающимся, что математика -это не только полезно, но и интересно.

В наше время почти каждого подростка привлекают компьютерные технологии. Их больше радует прохождение какой-либо игры, чем правильный ответ в решенной ими задаче. Создание “новой жизни” в лице программы, которая сможет выполнять какое-то действие будет давать такой же эффект как и удачное прохождение уровня в игре. Но в средней школе в рабочей программе на программирование выделено очень мало часов, да и в основном первый язык программирования затрагивается только в 9 классе.

Введение элективного курса по математике на основе программирования, повысит интерес школьников к математическим дисциплинам, т.к. для роста в данном направлении нужно овладеть соответствующими математическими навыками. Математические примеры заменяются более интересной задачей на создание программы, которая будет решать эти примеры. Для решения простых примеров таких как сложение двух чисел требуются минимальные знания программирования, но сам факт создания программы, которая будет работать и выполнять нужное действие корректно, оставит яркое впечатление у ученика.

Языком программирования данного курса будет Python. На сегодняшний день Python является одним из самых популярных языков. Он достаточно прост в изучении, имеет простой синтаксис и разносторонен в использовании. Так же в интернете есть очень много обучающих ресурсов для изучения языка.

Элективный курс по математике методом программирования на Python позволит не только поднять интерес обучающихся к математике, но и поможет выработать у них аналитический склад ума.

Объект исследования: процесс обучения математике и программирования на python в средней школе.

Предмет исследования: формирование учебных компетенций обучающихся средней школы в процессе обучения математики и программирования на python .

Цель исследования: теоретическое обоснование и проектирование элективного курса по формированию учебных компетенций на основе интеграции математики и программированию на python обучающихся средней школы

Задачи исследования:

- 1) изучение определений «компетенция» и «компетентность»;
- 2) выявление основных компонентов учебных компетенций, а также разработка критериев их оценки;
- 3) анализ современных педагогических технологий на уроках математики;
- 4) изучение проблемы интеграции математики и программирования;
- 5) проектирование модели формирования учебных компетенций обучающихся в процессе обучения математике и программирования на основе их интеграции;
- 6) реализация модели применения интеграции математики и программирования на python для проектирования элективного курса по формированию учебных компетенций учащихся;
- 7) проектирование элективного курса по формированию учебных компетенций;
- 8) анализ, обработка и подведение итогов педагогического эксперимента

Гипотеза исследования: если при обучении математики и программирования в средней школе спроектировать элективный курс на

основе интеграции математики и информатики, то это позволит повысить уровень развития учебных компетенций обучающихся.

1)

Теоретико-методологической базой исследования являются:

– *компетентностный подход* (А.В. Хуторской, И.А. Зимняя, В.Д. Шадриков, А.Н.Щукин, В.А. Болотов, В.В.Сериков и др.);

– *интегративный подход* (И.Д. Зверев, В.Н. Максимова, В.С. Безруков, К.Ю. Белая, Я.С. Бродский, А.Л.Павлов и др.);

Основные этапы исследования:

– *Постановочный.* Анализ психолого-педагогической и научно-методической литературы; сформулированы проблема, цель и гипотеза; определены объект, предмет и задачи исследования.

– *Собственно-исследовательский.* Проектирование структурной модели развития учебных компетенций на основе интеграции математики и программирования; проверка уровня развития компонентов учебных компетенций обучающихся.

– *Оформительно-внедренческий.* Анализ и обработка данных контрольного этапа эксперимента; формулировка выводов; оформление диссертационной работы.

Методы исследования:

– *теоретические:* изучение и анализ психолого-педагогической и научно-методической литературы по рассматриваемой проблеме; анализ Федерального Государственного Образовательного Стандарта Основного Общего Образования (ФГОС ООО);

– *эмпирические:* наблюдение за ходом учебного процесса; анкетирование обучающихся; беседы с учащимися; педагогический эксперимент;

Научная новизна заключается в том, что:

- уточнены понятия «компетенция», «компетентность», «учебная компетенция»;
- рассмотрены педагогические технологии на уроках математики;
- спроектирована модель формирования учебных компетенций на основе интеграции математики и программирования;
- реализована модель применения интеграции математики и программирования на python для проектирования элективного курса по формированию учебных компетенций учащихся;
- спроектирована программа элективного курса на основе спроектированной модели.

Теоретическая значимость исследования состоит в том, что спроектирована модель формирования учебных компетенций на основе интеграции математики и программирования.

Практическая значимость исследования состоит в том, модель применения интеграции математики и программирования на python для проектирования элективного курса по формированию учебных компетенций учащихся может быть реализована учителями математики и информатике в практике обучения в средней школе.

По структуре работа состоит из введения, оглавления, двух глав, выводов по каждой главе, заключения и библиографического списка литературы.

**ГЛАВА 1. ТЕОРЕТИЧЕСКОЕ ОСНОВЫ ПРИМЕНЕНИЯ
ИНТЕГРАЦИИ МАТЕМАТИКИ И ПРОГРАММИРОВАНИЯ НА
PYTHON ДЛЯ ПРОЕКТИРОВАНИЯ ЭЛЕКТИВНОГО КУРСА ПО
ФОРМИРОВАНИЮ УЧЕБНЫХ КОМПЕТЕНЦИЙ УЧАЩИХСЯ**

**1.1. ОПРЕДЕЛЕНИЕ ПОНЯТИЙ «КОМПЕТЕНЦИЯ» И
«КОМПЕТЕНТНОСТЬ»**

Ориентируясь в социально-экономическом развитии на переход к постиндустриальной модели общества, российское государство предъявляет новые требования и к системе образования. Принятие ФГОС общего образования как основного документа, нормирующего деятельность этой системы, закрепляет переход от традиционной модели, ориентированной на освоение знаний, умений и навыков к компетентностной модели, позволяющей развивать свои способности людям при решении различных задач. Основные сравнительные моменты представлены в Таблице 1.

Таблица 1

Сравнение традиционного и компетентностного подходов

| Основания для сравнения | Традиционный подход | Компетентностный подход |
|------------------------------------|--|--|
| Цель обучения | Приобретение теории, которая составляет содержания образования | Ориентация на практическую часть содержания образования, которая обеспечивает успех в жизнедеятельности |
| Характер образовательного процесса | Ученики получают знания уже в «готовом виде», которые они могут усвоить, запомнить и правильно воспроизвести | Ученики не получают знания уже в «готовом виде», а наоборот они должны их добыть сами в ходе изучения проблемы |
| Преобладающий компонент процесса | Контроль за учениками | Самостоятельная работа, получение знаний на практике |
| Основная формула результата | Знаю ЧТО | Знаю КАК |

ФГОС ООО ставит требования к результатам освоения основных образовательных программ, которые подразделяются на три вида: предметные, метапредметные и личностные образовательные результаты. В разделе «Требования к результатам освоения основной образовательной программы основного общего образования» они выступают в виде концептуальной идеи общего образования.

«Стандарт устанавливает требования к результатам освоения обучающимися основной образовательной программы основного общего образования:

– *личностным*, включающим готовность и способность обучающихся к саморазвитию и личностному самоопределению, сформированность их мотивации к обучению и целенаправленной познавательной деятельности, системы значимых социальных и межличностных отношений, ценностно-смысловых установок, отражающих личностные и гражданские позиции в деятельности, социальные компетенции, правосознание, способность ставить цели и строить жизненные планы, способность к осознанию российской идентичности в поликультурном социуме;

– *метапредметным*, включающим освоенные обучающимися межпредметные понятия и универсальные учебные действия (регулятивные, познавательные, коммуникативные), способность их использования в учебной, познавательной и социальной практике, самостоятельность планирования и осуществления учебной деятельности и организации учебного сотрудничества с педагогами и сверстниками, построение индивидуальной образовательной траектории;

– *предметным*, включающим освоенные обучающимися в ходе изучения учебного предмета умения специфические для данной предметной области, виды деятельности по получению нового знания в рамках учебного предмета, его преобразованию и применению в учебных, учебно-проектны

и социально-проектных ситуациях, формирование научного типа мышления, научных представлений о ключевых теориях, типах и видах отношений, владение научной терминологией, ключевыми понятиями, методами и приемами».[37]

Одним из приоритетных направлений перехода на ФГОС ООО является понимание учителем подхода к результатам педагогической деятельности, то есть того, что все таки качество образования напрямую зависит от возможности ребенка продолжить хорошо свое обучение. Фактически, нужно «научить ребенка учиться». Все это показывает нам на необходимость в овладении ребенком ключевыми компетенциями. Он несомненно продолжит учиться с успехом, если у него будут развиты ключевые компетенции, которые смогут позволить ему решать задачи, а также возникшие трудности.

В современной педагогике нет единых определений «компетенция» и «компетентность», в связи с чем были проанализированы различные источники в психолого-педагогической литературе.

Согласно А.В. Хуторскому «компетенция, с наших позиций, это социальное требование (внешняя норма) к образовательной подготовке ученика, необходимой для его качественной продуктивной деятельности в определенной сфере. А компетентность – владение, обладание учеником соответствующей компетенцией (внутреннее качество ученика)».[40]

О.Е. Лебедев обозначил суть компетентностного подхода следующим образом: «компетентностный подход — это совокупность общих принципов определения целей образования, отбора содержания образования, организации образовательного процесса и оценки образовательных результатов. К числу таких принципов относятся следующие положения:

- смысл образования заключается в развитии у обучаемых способности самостоятельно решать проблемы в различных сферах и видах

деятельности на основе использования социального опыта, элементом которого является и собственный опыт учащихся;

- содержание образования представляет собой дидактически адаптированный социальный опыт решения познавательных, мировоззренческих, нравственных, политических и иных проблем;
- смысл организации образовательного процесса заключается в создании условий для формирования у обучаемых опыта самостоятельного решения познавательных, коммуникативных, организационных, нравственных и иных проблем, составляющих содержание образования;
- оценка образовательных результатов основывается на анализе уровней образованности, достигнутых учащимися на определенном этапе обучения»:[12]

Таким образом, рассмотрев несколько источников, нельзя говорить о том, что исследуемые определения очевидны. В рамках исследования будем полагать, что компетенции – это знания, умения, навыки и способы их применения в определенных областях, а компетентным называют человека, который в определенной сфере деятельности обладает необходимыми знаниями, умениями и навыками, которые представляют ему возможность с полной уверенностью говорить об этой деятельности и эффективно работать в ней.

1.2. СТРУКТУРНЫЕ КОМПОНЕНТЫ УЧЕБНЫХ КОМПЕТЕНЦИЙ ОБУЧАЮЩИХСЯ

Одним из направлений математического просвещения и популяризации математики является «обеспечение непрерывной поддержки и повышения уровня математических знаний для удовлетворения любознательности человека, его общекультурных потребностей, приобретение знаний и навыков, применяемых в повседневной жизни и профессиональной деятельности» .

В компетентностном подходе в образовании А.В. Хуторский выделял ряд ключевых компетенций: ценностно-смысловую, общекультурную, информационную, коммуникативную, социально-трудовую, личного самосовершенствования и учебно-познавательную (таблица 2).

Таблица 2

Описание ключевых компетенций

| Ключевая компетенция | Содержательные компоненты компетенции |
|-----------------------------------|---|
| Информационная | Поиск, отбор и анализ необходимой информации |
| Коммуникативная | Знание языков; навыки общения в коллективе; способность написать различные документы юридического характера |
| Личностного самосовершенствования | Освоение способ различных видов саморазвития |
| Общекультурная | Опыт представления картины мира, опирающийся на культурологическое понимание мира |
| Социально-трудовая | Исполнение роли гражданина, умение действовать в соответствие с общественной и личной выгодой; владение этикой трудовых и гражданских взаимоотношений |
| Учебно-познавательная | Способность самостоятельно изучения и разработки собственных материалов |

Продолжение таблицы 2

| | |
|---------------------|---|
| Ценностно-смысловая | Видение и понимание окружающего мира, осознание собственной роли и предназначения, установка целевых и смысловых установок своих действий |
|---------------------|---|

В рамках данного исследования подробнее будут рассмотрены только учебно-познавательные компетенции (УК).

А.В. Хуторский в своей работе определяет УК как совокупность компетенций «в сфере самостоятельной познавательной деятельности, включающей элементы логической, методологической, общеучебной деятельности, соотнесенной с реальными познаваемыми объектами. Сюда входят знания и умения организации целеполагания, планирования, анализа, рефлексии, самооценки учебной деятельности, а также овладение креативными навыками продуктивной деятельности: добыванием знаний непосредственно из реальности, владение приемами действий в нестандартных ситуациях, эвристическими методами решения проблем».[40]

Качество результатов обучения во многом может зависеть от степени развития УК. В рамках данного исследования будем говорить об УК как об умении учиться.

Анализируя имеющиеся источники информации по проблеме развития УК не было найдено четкой системы оценивания уровня развития данного вида компетенций у школьников 13-15 летнего возраста.

В исследовании А.З.Зака приведено множество методик по диагностике, которые построены на дополнительном обучающем материале и включают в себя различные задачи, решаемые посредством предметно-действенной, наглядно-образной или словесно-знаковой форм, с целью развития основных способов теоретического мышления (аналитического, рефлексивного и синтезирующего). [13]

О.В.Брич, опираясь на модель формирования УК С.Г.Воровщикова, выделяет «личностный (предполагающий сформированность личностных и волевых качеств), когнитивный (включающий знания о различных способах осуществления учебно-познавательной деятельности), деятельностный (состоящий из общеучебных и специальных умений, определяемых видами речевой деятельности) и оценочный (подразумевающий владение приемами самоконтроля, самооценки и т.п.)».

В связи с этим была разработана система критериев структурных компонентов УК по трем уровням, основными компонентами которых можно выделить:

1. умение ставить задачи для достижения цели;
2. умение составлять план работы;
3. умение вести наблюдение;
4. установление причинно-следственных связей;
5. формулировка выводов.

Критерии развития компонентов УК обучающихся указаны в таблице

3.

Таблица 3

Критерии оценивания уровня развития компонентов учебно-познавательных компетенций обучающихся

| Компоненты | Низкий | Средний | Высокий |
|---|--|---|---|
| Умение ставить задачи для достижения цели | Ставит задачи только с помощью учителя | Умеет ставить иерархию задач из тех, что ему предложены | Умеет самостоятельно ставить задачи для достижения цели |

Продолжение таблицы 3

| | | | |
|---|--|--|--|
| Умение составлять план работы | Составляет план работы только с помощью учителя | Умеет составлять план по наработанному алгоритму | Умеет самостоятельно составлять план работы |
| Умение вести наблюдение | Ведёт наблюдения только с помощью учителя | Ведёт наблюдение по наработанному алгоритму | Ведет наблюдение самостоятельно |
| Установление причинно-следственных связей | Устанавливает причинно-следственные связи только с помощью учителя | Устанавливает причинно-следственные связи, когда решает поставленную | Умеет устанавливать причинно-следственные связи самостоятельно |

| | | | |
|---------------------------|---|--|---|
| | | перед ним проблему | |
| Формулирование выводов | Формулирует выводы только при помощи учителя | Формулирует выводы , когда решает поставленную перед ним проблему | Формулирует выводы самостоятельно |

1.3. СОВРЕМЕННЫЕ ПЕДАГОГИЧЕСКИЕ ТЕХНОЛОГИИ И ПРИЕМЫ ДЛЯ ФОРМИРОВАНИЯ УЧЕБНЫХ КОМПЕТЕНЦИЙ

В формировании многих качеств личности немалую роль играет школьная дисциплина – математика. На уроках математики школьники тренируются рассуждать, приводить доказательства, находить рациональные пути решения задач, делать выводы. В современном мире важны такие навыки, как развитая познавательная активность, самостоятельность, умение ставить перед собой задачи и решать их. Согласно новым стандартам образования, основной целью математического образования является освоения школьниками системой математических навыков, необходимых для применения на практике, поэтому необходимо организовать процесс обучения таким образом, чтобы поставленные задачи были выполнены. В данных условиях применение только традиционных методов обучения становится невозможным.

Рассмотрим некоторые из технологии, которые я считаю наиболее современными и интересными на сегодняшний день:

1) Информационно-коммуникационные технологии.

Информационно-коммуникационные технологии (ИКТ) – совокупность методов, производственных процессов и программно-технических средств, интегрированных с целью сбора, обработки, хранения, распространения, отображения и использования информации в интересах ее пользователей .

Использование информационных технологий на уроках математики является одним из важнейших аспектов совершенствования учебного процесса, повышает его практическую направленность, развивает интеллектуальные, математические способности учащихся. Информационные технологии на уроках математики привлекательны тем, что направлены на развитие творческих способностей учащихся, делая при этом работу учителя более продуктивной. Компьютерные технологии имеют ряд преимуществ: наполнение уроков новым содержанием, яркое эмоциональное восприятие материала, создание благоприятной атмосферы, ускорение процесса обучения и рост интереса учащихся к предмету. Всё это успешно сказывается на качестве обучения и успеваемости обучающихся.

Наиболее часто используемые элементы ИКТ в учебном процессе: презентации, интерактивные доски, электронные учебники и пособия, демонстрируемые с помощью мультимедийного проектора, тренажеры и программы тестирования, образовательные ресурсы сети Интернет. Существуют различные типы уроков с применением информационных технологий: урок-лекция, урок с мультимедийной поддержкой, урок постановки и решения задачи, интегрированные уроки и т.д. Наиболее эффективно можно применять информационные технологии на уроках математики при мотивации введения нового понятия, демонстрации моделей, моделировании, отработке определенных навыков и умений, контроле знаний. Таким образом, компьютерные технологии на уроке математики экономят время, повышают заинтересованность учащихся в изучении материала и эффективность обучения, позволяют провести многостороннюю и комплексную проверку знаний, умений, усиливают интерес к уроку, к предмету, наглядно и красочно представляют материал.

2) Технология проблемного обучения

На сегодняшний день современное общество все больше предъявляет высокие требования к учащемуся как к личности, которая способна решить

все проблемы самостоятельно. У детей возникает острая необходимость в формировании активной жизненной позиции, уверенной мотивации в сфере образования, критичности мышления.

В этом плане традиционная система обучения имеет значительные недостатки по сравнению с проблемным обучением.

На данный момент проблемное обучение- организация учебных занятий, в рамках которой под руководством учителя учащиеся создают проблемный ситуации и активно самостоятельно их решают.

Технологию проблемного обучения можно использовать в рамках урока, с целью изучения нового материала или первичного закрепления материала.

Данная технология позволяет:

- усилить познавательную деятельность учащихся на уроке, что позволяет управиться с большим объемом учебного материала;
- выработать строгую учебную мотивацию;
- применять полученные знания в организации самостоятельной работы для получения новых умений в поиске из разных источников информации;
- значительно поднять самооценку учеников, т. к. при решении проблемы выслушиваются и обсуждаются любые решения.

Второй вид проблемного изложения нового материала- когда ученикам выдается индивидуальное задание, которое требует самостоятельного решения, либо когда ученикам дается вопрос, который требует сопоставления ряда изученных явлений, на который они должны предоставить собственные рассуждения и выводы.

Структура проблемного обучения выглядит так:

- 1) подготовительный этап;
- 2) этап организации проблемной ситуации;

3) понимание учениками отдельного вопроса темы в виде учебной проблемы;

4) выставление гипотезы, предположений, обоснований гипотезы;

5) собственные рассуждения, вариант решения и вывод по представленной учебной проблеме;

6) закрепление и рассуждение над полученными данными, использование этих знаний в новых ситуациях.

3)Технология критического мышления

Критическое мышление-это способность обрабатывать информацию логически и каких-либо личностных качеств, для дальнейшего использования в иных ситуациях.

Зачастую эта технология приводит к новым идеям. Основные свойства критического мышления:

Критическое мышление -это независимое мышление

Информация-это отправная, а не конечная точка критического мышления.

Критическое мышление начинается с постановки вопросов и понимания проблем, которые необходимо решить.

Социальное мышление

Благодаря этой технологии мы можем повысить интерес к обучению учеников, социальную ориентированность, а также наиболее эффективную работу с информацией.

Приемы, которые можно использовать для данной технологии- это индивидуальная работа, ролевые игры, групповые проекты. Всё это помогает в приобретении знаний и умений, помогают глубоко усвоить материал, а также повышает интерес к предмету, развивает мыслительные и креативные навыки.

ТРКМ включает в себя три стадии: вызова, осмысления и размышления.

Стадия вызова. На этом этапе нужно проделать работу, которая поднимет интерес к теме, поставить цель для изучения материала, повторение материала, и повышение интереса к теме.

Стадия осмысления нового материала (новой информации, идеи, понятия). На данном эта происходит основная работа ученика. Он изучает новый материал в книге, пытается осмыслить входные данные задачи, а также понять речь учителя.

Стадия размышления или рефлексии. На данном этапе ученику предстоит проанализировать весь изученный материал.

Всем трем стадиям обязательно надо придерживаться на уроке, так как это отображает сложный мыслительный процесс. Эта особенность данной технологии значительно расширяет границы ее применимости.

Учебная дисциплина является важной частью процесса формирования УК в связи с тем, что компетенции, которые обучающиеся развивают на одной дисциплине могут транслироваться на изучение других дисциплин. При этом у обучающихся формируется целостное информационное пространство.

УК не могут развиваться без соответствующей мотивации учащихся, которые должны понимать изучаемые объектно-субъектные отношения. Мотивацией для изучения той или иной дисциплины может выступать интерес к рассматриваемым явлениям и их связи с окружающим миром. Однако причинно-следственные связи могут быть неправильно выстроены обучающимися (или вовсе не построены) вследствие слабого развития УК, из-за чего их мотивация будет постепенно падать. Таким образом, учителю следует уделять внимание различным приемам развития УК.

На уроках учащиеся выполняют задания как практического, так и теоретического характера. Из-за того, что ни содержание стандартных школьных задач, ни процесс их решения обычно не вызывают у учащегося

познавательного интереса и желания работать, учителю необходимо разрабатывать творческие задания для стимулирования работы учеников.

Стандартные школьные задачи, содержащиеся в учебниках, хоть и подразумевают собой как теоретический, так и практический подход к их решению, не всегда побуждают к познавательной деятельности. Вследствие чего у обучающихся пропадает мотивация к их выполнению. В связи с этим актуальной проблемой является использование на уроках различных заданий по развитию УК обучающихся. Рассмотрим приемы УК на уроках:

1. Проблемные ситуации. Нахождение учителем совместно с классом проблемы при решении задачи, приводящей к нахождению учениками нового типа учебного задания, требующего особенного способа действия с ним.

2. Использование на уроках задач «без вопросов». При выполнении задач, не содержащих конкретных требований, обучающиеся замечают отсутствие вопроса. В этом случае можно прийти к выводу о построении разных типов вопросов к задаче самостоятельно, что является проявлением познавательной деятельности.

3. Исследовательская деятельность. Для того чтобы ученики могли самостоятельно проводить различные исследования, выдвигать гипотезы и подтверждать или опровергать их, создаются специальные условия. Ученики перестают получать полную информацию необходимую для достижения поставленной цели, а остаются в состоянии «частично незнания». В данных условиях все ученики находятся на равных права, имеют собственную точку зрения. Все гипотезы, выдвинутые учениками, должны проверять как возможный вариант решения поставленной задачи или достижения цели исследования. Роль учителя в данном приеме отходит с ведущей позиции. Он выступает в качестве направляющего и поддерживающего звена команды.

4. Применение ИКТ. На сегодняшний день навыки работы с ИКТ очень важны для учеников. Такие навыки как подготовка презентации для

защиты докладов, умение найти недостающую информацию в сети, работа с различными электронными платформами мотивируют учеников к познавательной деятельности и развивают нестандартное мышление учеников.

5. Самостоятельная работа - работа, при которой ученики, имея только конечную цель, самостоятельно определяют задачи, необходимые для достижения этой цели, производят необходимые умственные или физические действия и достигают этой цели без вмешательства со стороны учителя. Такая работа формирует в человеке умение самостоятельно принимать решения в различных ситуациях и находить способы решения проблем.

6. Интегрированные уроки Позволяют ученикам увидеть взаимосвязь различных образовательных предметов в рамках одной темы. У ученика создается общая картина мира. Часто интегрированные уроки позволяют взглянуть на давно известную проблему с необычной стороны, относительно предмета обучения. На таком уроке ученик использует не только знания из одного предмета, но и опыт из другого предмета, что значительно расширяет уровень познавательной деятельности.

Таким образом, каждому учителю на основе вышеперечисленных и других приемов нужно найти собственную стратегию развития УК, при которой будет проще организовать деятельность обучающихся на уроках

1.4. ПРОБЛЕМА ИНТЕГРАЦИИ МАТЕМАТИКИ И ПРОГРАММИРОВАНИЯ В СРЕДНЕЙ ШКОЛЕ

Преподавание математики с помощью компьютерного программирования — либо в рамках стандартного математического курса, либо в качестве факультатива — может придать математическим понятиям контекст и актуальность, но при этом требует такой же строгости, как и традиционное обучение математике. Например, в

следующем упражнении учащимся предлагается написать компьютерную программу для решения алгебраической задачи:

Напишите программу для магазина по продаже мороженого. Цена продажи составляет 0,39 доллара за кг. Налог с продаж составляет 8,25 процента.

В традиционном математическом классе это было бы более простой задачей: ученика просили рассчитать стоимость только одного определенного количества мороженого. Но, интегрируя компьютерное программирование, мы можем развивать навыки логического и критического мышления учащихся, развивая их способность абстрактно идентифицировать переменные компоненты, обращать внимание на точность целых и десятичных чисел в программе, разрабатывать математическую модель и создавать алгоритмы с шаблонами. Ученики также учатся критиковать рассуждения других и помогать друг другу во время разработки программы для совместного обучения. Это упражнение может помочь сделать математику более актуальной, поскольку учащиеся получают чувство выполненного долга за успешное написание компьютерной программы.

Это всего лишь один пример того, как компьютерное программирование может быть интегрировано для улучшения обучения математике.

4 причины, почему математика важна для информатики:

Но есть и другой способ определить сильную математическую подготовку: способность к абстрактному мышлению, критическому мышлению и логической дедукции — математический способ мышления. В связи с этим для достижения успеха в области компьютерных наук необходимы сильные знания математики.

1. Математика учит пониманию и общению с помощью абстрактного языка.

Компьютерное программирование имеет свои собственные языки, которые очень абстрактны. Используя синтаксис, нужно представлять определенные процессы, команды и визуальные эффекты с помощью пунктуации, символов и отдельных слов. Для человека, не имеющего опыта мышления или общения на абстрактных языках, изучение языка программирования может быть ужасающим.

Однако абстрактные языки программирования очень похожи на математический язык, который студенты изучают на уроках математики. От простых равенств до сложных математических представлений, изучение математики учит учащихся искусству чтения, понимания, формулирования мыслей и общения на абстрактном языке.

Конечно, математический язык и языки компьютерного программирования — это не совсем одно и то же. Но опыт использования любого абстрактного языка дает начинающим программистам преимущество.

2. Математика учит работать с алгоритмами.

Алгоритм — один из самых обсуждаемых терминов на технологической сцене. Короче говоря, алгоритм — это абстракция некоторого процесса в форме, в которой процесс можно повторять, реализовывать по-разному и применять к новым задачам.

Слово может использоваться чаще в информатике, но большинство учеников сначала используют алгоритмы в математике. Например, рассмотрим уравнение типа $5 + x = 7$. Учащиеся учатся находить неизвестное слагаемое, вычитая известное слагаемое из суммы. Это алгоритм, который учащиеся быстро учатся применять к новым задачам и реализовывать по-разному.

3. Математика учит учеников анализировать свою работу.

За целый день программирования любой ученый-компьютерщик обязательно совершит ошибку. Таким образом, программисты должны

знать, как оценивать проблему, анализировать свою работу и исправлять ошибки.

Математика — один из немногих предметов, где ученики анализируют свою работу таким образом. Учащийся может ответить на математический вопрос (Сколько весят вместе щенок и котенок?), понять, что его ответ неразумен (231 фунт), и проанализировать свой собственный процесс, чтобы понять свою ошибку и как ее исправить (возможно, он забыл перевести унции в фунты). Короче говоря, математика готовит учащихся к исправлению ошибок.

4. Помимо общих навыков, программирование по-прежнему включает в себя много математики.

В дополнение к общим навыкам, важным для программирования, важны математические факты и цифры. По мере того, как компьютерное программирование все больше взаимодействует с нашим миром, возрастает важность точного моделирования этого мира с помощью математики.

Например, чтобы построить самоуправляемый автомобиль, уравнения, используемые для программирования его поворотов, ускорения и приемлемого расстояния до других автомобилей, должны быть точными.

Чтобы стать специалистом по информатике, требуется достаточное количество математических знаний и навыков. Что еще более важно, успех в информатике требует способности мыслить математически.

1.5. МОДЕЛЬ ФОРМИРОВАНИЯ УЧЕБНЫХ КОМПЕТЕНЦИЙ ОБУЧАЮЩИХСЯ НА ОСНОВЕ ИНТЕГРАЦИИ МАТЕМАТИКИ И ПРОГРАММИРОВАНИЯ

Модель – это абстрактная проекция сути исследования, которая состоит из несколько связанных компонентов: цель, содержание, методы, средства и формы организации обучения. В рамках данного исследования, я

считаю, что лучше рассматривать методы, средства и формы организации обучения как единый блок, отмечая при этом более тесную взаимосвязь между ними. В процессе обучения учитель, опираясь на исследованные педагогические технологии УК, полностью организует учебный процесс от целеполагания до оценки и анализа полученных результатов.

Спроектированная структурная модель развития УК на основе интеграции математики и информатики включает в себя следующие блоки:

1. Целевой. Цель является системообразующим компонентом, которая определяет содержание обучения, выбор методов, средств и форм организации обучения, способствующих ее достижению.

2. Содержательный. Под содержанием образования в настоящее время понимают «социально и личностно детерминированное, фиксированное в педагогической науке представление о социальном опыте, подлежащее усвоению подрастающим поколением», то есть это педагогическая модель социального опыта. Содержание обучения предмету (математике в том числе) представляет собой некую информацию и комплекс заданий и упражнений, совокупность которых в идеале должна обеспечить изучение системы знаний, усвоение умениями и навыками, развитие социально и профессионально значимых личностных качеств обучающегося .

3. Организационно-процессуальный. Для достижения основной цели и решения поставленных задач в качестве методов выявлены приемы проведения интегрированных уроков, использование ИКТ, проведение самостоятельных работ; в качестве форм организации выявлены работа в парах и группах, практические занятия, индивидуальные консультации.

4. Контрольно-оценочный. Выявление уровня развития УК обучающихся в результате апробации спроектированной модели путем использования выделенных критериев (параграф 1.2.) посредством методов статистического анализа.

Схематично модель представлена в виде таблицы 4.

Модель развития учебно-познавательных компетенций на основе интеграции математики и программирования в средней школе

| Целевой блок | Содержательный блок | Организационно-процессуальный блок | Контрольно-оценочный блок |
|--|---|--|--|
| <p>Цель: развитие учебно-познавательных компетенций обучающихся на основе интеграции математики и информатики</p> | <p>Методологические подходы: компетентностный, интегративный</p> | <p>Средство обучения: кружковая программа «Программирование в математике»</p> | <p>Самоконтроль, контроль</p> |
| | | <p>Формы обучения: работа в парах и группах, практические занятия, консультации.</p> | <p>Результат: низкий, средний, высокий уровни</p> |
| | | <p>Методы обучения: интегрированные занятия, использование программирования, самостоятельная работа</p> | <p>Применение математических методов обработки результатов опытно-экспериментальной работы</p> |

Таким образом, мы полагаем, что совокупность целевого, содержательного, организационно-процессуального и контрольно-оценочного блоков структурной модели способствует реализации развития УК обучающихся на основе интеграции математики и программирования посредством разработки программы кружка «Программирование в математике».

ВЫВОДЫ ПО ГЛАВЕ 1

Анализ психолого-педагогического и научно-методического материала по проблеме исследования определил выводы:

1. Уточнены определения компетенции как знания, умения, навыки и способы их применения в определенных областях, а компетентным называют человека, который в определенной сфере деятельности обладает необходимыми знаниями, умениями и навыками, которые представляют ему возможность с полной уверенностью говорить об этой деятельности и эффективно работать в ней. УК подразумевают умение учиться.

2. Выявлены основные компоненты УК: умение ставить задачи для достижения цели, умение составлять план работы, умение вести наблюдение, установление причинно-следственных связей, формулирование выводов, а также предложены критерии оценки их развития: низкий, средний, высокий.

3. Проанализированы современные педагогические технологии и приемы для формирования учебных компетенций.

4. Изучена проблема интеграции математики и программирования о том, что на интегрированных уроках учащиеся получают навыки в применении математических знаний при решении различных задач и оценивают полученные результаты, что способствует их комплексному развитию. Программирование в таком случае выступает в роли прикладной науки.

5. Спроектирована модель формирования учебных компетенций обучающихся в процессе обучения математике и программирования на основе их интеграции.

ГЛАВА 2. РЕАЛИЗАЦИЯ МОДЕЛИ ПРИМЕНЕНИЯ ИНТЕГРАЦИИ МАТЕМАТИКИ И ПРОГРАММИРОВАНИЯ НА PYTHON ДЛЯ ПРОЕКТИРОВАНИЯ ЭЛЕКТИВНОГО КУРСА ПО ФОРМИРОВАНИЮ УЧЕБНЫХ КОМПЕТЕНЦИЙ УЧАЩИХСЯ

Пояснительная записка

Элективный курс предназначен для учащихся 7-9 классов, рассчитан на 17 часов, 8 часов теоретических занятий, 8 часов практических занятий и 1 час контрольная работа. Содержание элективного курса направлено на то, чтобы учащиеся научились применять знания не только в области математики, а также создавать простейшее программное обеспечение, которое сможет решать некоторые математические задачи при разных входных данных. Язык программирования предлагается использовать python.

Курс включает в себя освоение алгоритмического языка в виде блок-схем, языка программирования python, развитие логического мышления, знакомство с разработкой программного обеспечения и его тестирования.

Курс поможет понять некоторые темы курса математики, так как написание алгоритма решения задач в виде блок-схем дает возможность посмотреть на математические задачи под другим углом. В школах учащиеся часто забывают, как решать те или иные математические задачи так как им дали немного теории, и они решили пару примеров. Такие знания забываются в скором времени если их не использовать. Наглядный пример в виде блок-схемы запоминается куда лучше, чем в виде текста.

Также курс направлен на развитие логического мышления, он позволит учащимся в будущем лучше запоминать тот или иной материал, даже не относящийся к математике. Это связано с тем, что развитое логическое мышление позволит хорошо структурировать информацию в голове и лучше запомнить ее.

Еще одна из важных особенностей курса, это получения актуальных знаний и навыков. Информационные технологии развиваются большими шагами, проанализировав сервис для поиска работы HeadHunter можно сделать вывод, что специалистов в данной области не хватает. Также изучив данный сервис можно сделать вывод, что python входит в тройку самых востребованных языков программирования.

Данный элективный курс позволяет получить или дополнить знания языка программирования python.

В период удаленной работы элективный курс можно запустить и проводить удаленно. Для введения онлайн дневника учащихся можно использовать сервис “Google Classroom”. Для удаленных встреч подойдет такое решение как “Zoom” – массовые онлайн звонки. Также существуют множество сервисов, которые обеспечивают удобное написание кода на языке программирования python, а также позволяют смотреть индивидуальные задания в режиме реального времени. Один из таких сервисов “Python Onlinegdb”. Этот сервис позволяет учащимся писать код напрямую в браузере, а когда потребуется помощь или проверка педагога, то учащиеся могут скопировать ссылку на их код и отправить педагогу. По этой ссылке можно всегда получить актуальный код программного обеспечения.

Цели курса:

- научить проецировать алгоритмы решений некоторых математических задач в алгоритмический язык – блок-схемы;
- научить использовать алгоритмический язык в написании программного обеспечения на основе языка программирования python;
- развить логическое мышление;
- сформировать понимание о способах решения математических задач с помощью информационных технологиях.

Задачи курса:*Обучающие:*

- Познакомить учащихся с алгоритмическим языком и его представлением в виде блок-схем;
- познакомить с различными алгоритмическими конструкциями языков программирования на примере языка python;
- научить использовать ту или иную алгоритмическую конструкцию языка программирования при решении задач;
- научить правильному использованию типов данных для эффективной работы программного обеспечения;
- научить тестировать и отлаживать программное обеспечение.

Развивающие:

- Сформировать или развить операционный тип мышления, который направлен на нахождение оптимального пути решения задачи;
- предоставить новые знания в области создания программного обеспечения и языка программирования python.

Воспитательные:

- Привитие навыков использования информационных технологий, в том числе компьютером, в учебной и профессиональной деятельности учащихся;
- привитие стремления к овладению информационными технологиями у учащихся.

Формы занятий

Формы занятий в данном курсе – теоретические и практические. На теоретических занятиях в основном повторение основных алгоритмов решения математических задач, решение пары задач у доски для усвоения алгоритмов и проецирование этих алгоритмов на блок-схемы. Практические же занятия больше рассчитаны на самостоятельную работу учащихся с возможностью получить помощь от педагога.

При проведении занятий используются четыре формы работы:

Интерактивный – обучающиеся работают вместе с педагогом. Такая форма в основном используется на теоретических занятиях, где педагог только задает наводящие вопросы и объясняет материал из курса математики, а учащиеся строят блок-схемы и решают задачи у доски.

Самостоятельная – учащиеся самостоятельно выполняют индивидуальные задания, с возможностью получить подсказку у преподавателя.

Контрольная – учащиеся выполняют индивидуальные задания без возможности посмотреть в записи с теоретических и практических уроков и без возможности получить подсказку от учителя или других обучающихся.

Групповая- учащиеся разбиваются на группы по 2-3 человека, обсуждают работу, выдвигают способы решения, слушают друг друга и выбирают верный ответ.

Технологии и формы обучения

- Теоретические занятия;
- практические занятия;
- информационно-коммуникационные технологии;
- технологии критического мышления;
- технологии проблемного обучения.

Режим занятий:

Занятия проводятся: 2 раз в неделю по 2 часа (итого 4 часа в неделю, 17 часов в год).

2.1.МЕТОДИЧЕСКАЯ ЧАСТЬ

Индивидуальные задания на практических уроках выполняются с помощью персонального компьютера и необходимых программных средств.

Каждый теоретический урок строится следующим образом: повторение некоторой части курса математики, выполнение пары математических задач, обсуждения общего алгоритма выполнения таких

задач, построение блок-схемы этого алгоритма. Кроме повторения теории курса математики, урок идет в интерактивном формате, учащиеся пытаются выполнить задания, выявить алгоритм и нарисовать блок-схему с помощью подсказок и наводящих вопросов от педагога.

Практические уроки же имеют немного другую структуру. В начале урока учащиеся вспоминают блок-схемы, составленные на теоретическом уроке, и выполняют индивидуальное задание по написанию программного обеспечения с помощью данных блок-схем. На практических занятиях ученики могут получить подсказки и помощь в выполнении работы от педагога

Теоретическая и практическая части курса проходят последовательно, чтобы сразу же закреплять знания на практике.

На последнем занятии ученики выполняют контрольную работу, на которой они не могут воспользоваться записями из других уроков или помощью педагога.

Планируемые результаты элективного курса:

В рамках данного курса учащиеся овладевают следующими знаниями и умениями:

- Знают, что такое алгоритмический язык, умеют пользоваться им и представлять в виде блок-схем;
- знают различные алгоритмические конструкции языков программирования;
- умеют использовать ту или иную алгоритмическую конструкцию языка программирования при решении задач;
- умеют правильно использовать типы данных для эффективной работы программного обеспечения;
- умеют тестировать и отлаживать программное обеспечение .

Форма контроля за уровнем достижения учащихся:

Уровень достижения учащихся контролируется с помощью анализа составленных алгоритмов и программных обеспечений, реализованных на языке программирования python.

Качество знаний и умений ученика оценивается следующими характеристиками:

- Знание различных алгоритмических конструкций языков программирования;
- умение правильно составить алгоритм и нарисовать к нему блок-схему;
- умение найти более эффективное решение задачи;
- умение тестировать и отлаживать программное обеспечение.

В течение всего курса выставляются оценки за индивидуальные задания, а также за работу на теоретических занятиях тем, кто участвовал в обсуждении и находился у доски. По окончании курса каждый ученик получает оценку, основанную на средней оценке за курс. Контрольная работа имеет больший вес в подсчете средней оценки.

2.2.СОДЕРЖАНИЕ ЭЛЕКТИВНОГО КУРСА

2.2.1. УЧЕБНО-ТЕМАТИЧЕСКОЕ ПЛАНИРОВАНИЕ

ЭЛЕКТИВНОГО КУРСА

Элективный курс предназначен для учащихся 9–11 классов, рассчитан на 17 часов, 8 часов теоретических занятий, 8 часов практических занятий и 1 час контрольная работа. Содержание элективного курса направлено на то, чтобы учащиеся научились применять знания не только в области математики, а также создавать простейшее программное обеспечение, которое сможет решать некоторые математические задачи при разных входных данных. Язык программирования предлагается использовать python.

Курс включает в себя освоение алгоритмического языка в виде блок-схем, языка программирования python, развитие логического мышления, знакомство с разработкой программного обеспечения и его тестирования.

Схематично учебно-тематическое планирование представлено в виде таблицы 5.

Таблица 5.

| № п/п | Тема | Количество часов |
|----------|--|------------------|
| 1 | Теоретическое занятие по теме «Задачи на движение» | 2 |
| | Теория из курса математики по задачам на движение | |
| | Решение задач на движение | |
| | Построение блок-схем для задач на движение | |
| 2 | Практическое занятие по теме «Задачи на движение» | 2 |
| | Повторение блок-схемы из теоретического занятия | |
| | <i>Индивидуальное задание 1:</i> создание программного обеспечения для решения задач на движение одного объекта на языке программирования python | |
| | <i>Индивидуальное задание 2:</i> создание программного обеспечения для решения задач на движение двух объектов на языке программирования python | |
| 3 | Теоретическое занятие по теме «Числа Фибоначчи» | 2 |
| | Теория из курса математики по числам Фибоначчи | |
| | Решение задач на поиск n-ого элемента ряда | |
| | Построение блок-схемы по определению n-ого элемента ряда Фибоначчи | |
| 4 | Практическое занятие по теме «Числа Фибоначчи» | 2 |
| | Повторение блок-схемы из теоретического занятия | |
| | <i>Индивидуальное задание 3:</i> создание программного обеспечения для решения задач на n-ого члена ряда Фибоначчи на языке программирования python | |
| | <i>Индивидуальное задание 4:</i> написание блок-схемы для нахождения минимального элемента ряда Фибоначчи, таким, чтобы количество цифр числа было равно введённому с клавиатуры | |
| 5 | Теоретическое занятие по теме «Стандартные квадратные уравнения» | 2 |
| | Теория из курса математики по квадратным уравнениям | |
| | Решение квадратных уравнений | |

| | | |
|----------|--|----------|
| | Построение блок-схемы для решения квадратных уравнений | |
| 6 | Практическое занятие по теме «Стандартные квадратные уравнения» | 2 |
| | Повторение блок-схемы из теоретического занятия | |
| | <i>Индивидуальное задание 5:</i> создание программного обеспечения для нахождения корней квадратного уравнения на языке программирования python | |
| 7 | Теоретическое занятие по теме «Решение неравенств второй степени с одной переменной» | 2 |
| | Теория из курса математики по решению неравенств второй степени с одной переменной | |
| | Решение неравенств второй степени с одной переменной | |
| | Построение блок-схемы для решения неравенств с одной переменной | |
| 8 | Практическое занятие по теме «Решение неравенств второй степени с одной переменной» | 2 |
| | Повторение блок-схемы из теоретического занятия | |
| | <i>Индивидуальное задание б:</i> создание программного обеспечения для решения неравенств второй степени с одной переменной на языке программирования python | |
| 9 | Итоговая контрольная работа | 1 |

2.2.2. СОДЕРЖАНИЕ ТЕОРЕТИЧЕСКОЙ ЧАСТИ КУРСА

1. Задачи на движение

На данном уроке вспомним несколько типов задач на движение.

Первый тип задач, это задачи, где некоторый объект движется часть пути с одной скоростью, часть пути с другой скоростью. И необходимо найти путь, пройденный объектом. Для примера рассмотрим следующую задачу: спортсмен бежал по прямой дороге со скоростью 8 км/час, потом он бежал под горку 2 часа со скоростью на 3 км/час меньше, чем по прямой дороге. Какое расстояние он пробежал за 5 часов? Пример блок-схемы изображен на рисунке 1.

Второй тип задач, задача с двумя движущимися объектами. Пример такой задачи: расстояние между двумя объектами 400 км, первый объект движется со скоростью 80 км/час, второй 100 км/час. Чему будет равно расстояние между объектами через час. Необходимо рассмотреть все 4 случая, разных направлений объектов. Пример блок-схемы изображен на рисунке 2.

Каждую задачу один из учеников решает у доски вместе с классом. После решения задачи, в формате диалога, рисуются блок-схемы решения данных типов задач.

Ключевые вопросы для подталкивания к правильной блок-схеме:

- Какой фигурой начинается и заканчивается блок-схема?
- какой фигурой изображается ввод и вывод в блок-схемах?
- как правильно отобразить условие?
- какой фигурой изображается какое-либо действие в блок-схемах?



Рис. 1. Блок-схема метода решения задач первого типа

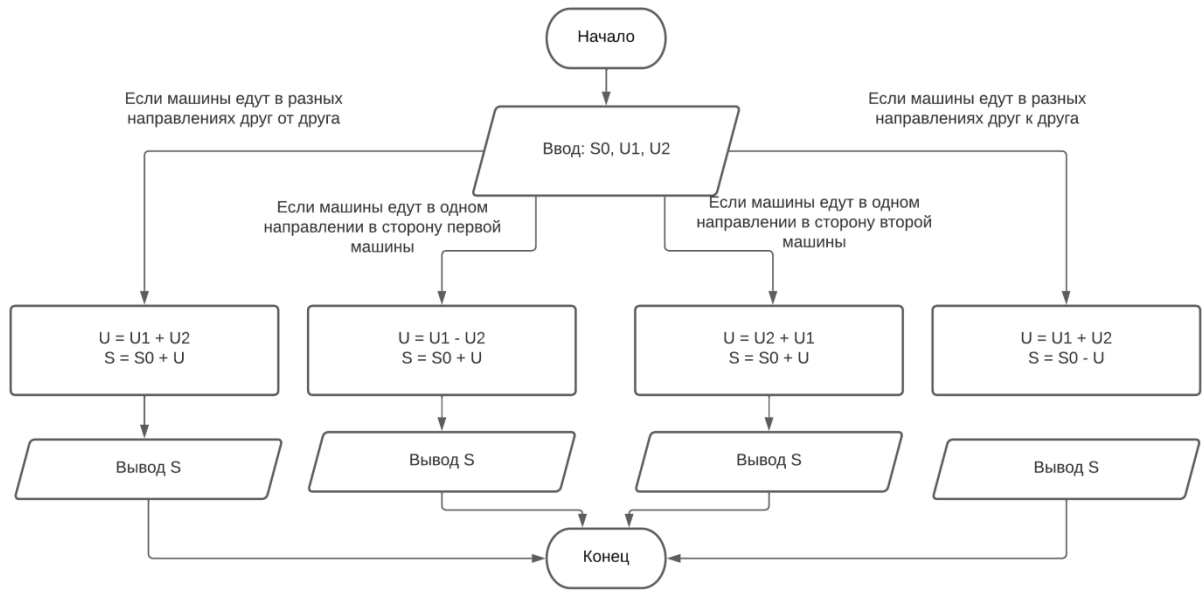


Рис. 2. Блок-схема метода решения задач второго типа

План-конспект учебного занятия

Тема учебного занятия: Задачи на движение.

Вид учебного занятия: Теоретическое.

Тип учебного занятия: С обратной связью.

Цель учебного занятия: Вспомнить как решаются задачи на движение, построить общий алгоритм решения таких задач и отобразить его в блок-схеме.

Задачи учебного занятия:

Образовательные:

- Вспомнить как решать задачи на движение из курса математики;
- научиться строить блок-схемы по задачам на движение.

Воспитательные:

- Привитие интереса к алгоритмическому языку и блок-схемам
- привитие самостоятельности в решении поставленных задач.

Развивающие:

- Развитие логического мышления;
- развитие интеллектуальных способностей учащихся.

Технические средства обучения: автоматизированное рабочее место преподавателя (ПК, интерактивная доска).

Ход занятия

1) Организационный момент

1. Приветствие и знакомство;
2. отметка присутствующих;
3. объявление темы, постановка целей занятия.

Темой сегодняшнего занятия является «Задачи на движения»

2) Изложение нового материала

Для решения задач на движения нам потребуется вспомнить из курса математики несколько формул.

Какая основная формула для задач на движение?

Формула нахождения пути по скорости и времени:

$$S = v * t$$

S – пройденный путь.

v – расстояние, которое проходит тело за единицу времени.

t – время движения тела

3) Закрепление полученных знаний

Для закрепления теоретической части решим несколько задач на движение.

Пример 1: спортсмен бежал по прямой дороге со скоростью 8 км/час, потом он бежал под горку 2 часа со скоростью на 3 км/час меньше, чем по прямой дороге. Какое расстояние он пробежал за 5 часов?

Решение: для начала определим сколько времени спортсмен бежал по прямой дороге. Для этого достаточно вычесть из общего времени время, которое он потратил на горку.

$$t_1 = t - t_2$$

$$t_1 = 5 - 2 = 3 \text{ часа}$$

Далее, определим какое расстояние пробежал спортсмен на первом и втором участке

$$S_1 = 8 * 3 = 24 \text{ км}$$

$$S_2 = 5 * 2 = 10 \text{ км}$$

Остается только сложить два расстояния и получить ответ

$$S = 24 + 10 = 34 \text{ км}$$

Построение блок схемы:

Теперь построим блок-схему для решения задач данного типа.

Для начала вспомним какой фигурой блок-схемы должны начинаться и заканчиваться. Начинается блок-схема с овала с текстом «Начало», а заканчивается овалом с текстом «Конец».

Подумаем какой блок должен использоваться следующим. Далее должен идти параллелограмм с указанием, какие необходимы входные данные. Для задач такого типа необходимы следующие данные:

- Скорость на первом участке (v_1);
- скорость на втором участке (v_2);
- время движения на втором участке (t_2);
- общее время движения (t).

Этих входных данных будет достаточно. Далее нам необходимо использовать прямоугольный блок с действиями. В этом блоке мы выполним все расчеты.

Следующий блок – блок вывода информации. Параллелограмм с выводом общего расстояния.

И в заключении блок-схемы – блок конца алгоритма. В итоге должна получиться блок-схема, изображенная на рисунке 1.

Пример 2: расстояние между двумя объектами 400 км, первый объект движется со скоростью 80 км/час, второй 100 км/час. Чему будет равно расстояние между объектами через час. Необходимо рассмотреть все 4 случая, разных направлений объектов.

Решение: для решения этой задачи нужно рассмотреть 4 случая как могут двигаться объекты.

Первый случай – объекты движутся в разные стороны друг от друга:

Скорость отдаления объектов будет равна:

$$v = v_1 + v_2$$

$$v = 180 \text{ км/час}$$

Далее необходимо найти расстояние между объектами через час.

$$S = S_0 + v * t$$

$$S = 400 + 180 * 1 = 580 \text{ км}$$

Второй случай – объекты движутся в разные стороны друг к друга:

Скорость сближения объектов будет равна:

$$v = v_1 + v_2$$

$$v = 180 \text{ км/час}$$

Далее необходимо найти расстояние между объектами через час.

$$S = S_0 - v * t$$

$$S = 400 - 180 * 1 = 240 \text{ км}$$

Третий случай – объекты движутся в одну сторону в направлении первого объекта:

Скорость второго объекта относительно первого будет равна:

$$v = v_2 - v_1$$

$$v = 20 \text{ км/час}$$

Далее необходимо найти расстояние между объектами через час.

$$S = S_0 - v * t$$

$$S = 400 - 20 * 1 = 360 \text{ км}$$

Четвертый случай – объекты движутся в одну сторону в направлении второго объекта:

Скорость второго объекта относительно первого будет равна:

$$v = v_1 - v_2$$

$$v = -20 \text{ км/час}$$

Далее необходимо найти расстояние между объектами через час.

$$S = S_0 - v * t$$

$$S = 400 + 20 * 1 = 420 \text{ км}$$

Построение блок-схемы:

Теперь построим блок-схему для решения задач данного типа.

Блок-схема строится по принципу из первого примера. Результат построения отображен на рисунке 2.

4) Подведение итогов

Подводятся итоги занятия, выставляются баллы.

2. Числа Фибоначчи

В начале урока вспомним что такое числа Фибоначчи.

Числа Фибоначчи – это последовательность чисел, где каждое новое число равно сумме предыдущих двух чисел в последовательности. В ряде Фибоначчи первое и второе число равны 0 и 1 соответственно. Числа Фибоначчи можно представить следующей формулой

$$F_n = F_{n-2} + F_{n-1}; \text{ где } F_0 = 0, F_1 = 1, n > 2$$

Напишем способ нахождения n-ого числа в данной последовательности:

1. Возьмем число $x = 2$
2. Найдем и запишем F_x
3. Если $x < n$, то прибавим к x единицу и вернемся к пункту 2

4. Когда x будет равно n , значение F_x будет искомым значением

Найдем пару значений ряда Фибоначчи для лучшего восприятия теории.

Найти F_{10}

Найти F_{15}

На каком n числа в ряду станут 4 – х значные? 5 – ти значные

Далее, в формате интерактивного общения, нарисуем на доске блок-схему нахождения n -ого числа из ряда Фибоначчи, которая изображена на рисунке 3.

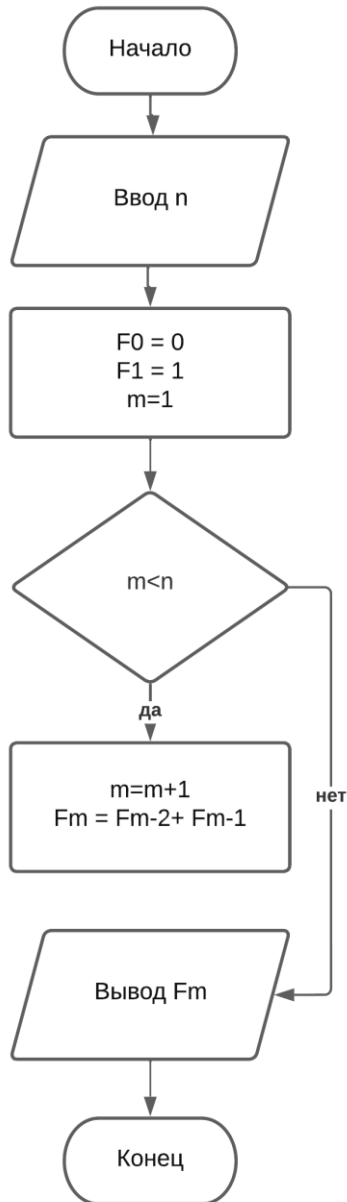


Рис. 3. Блок-схема метода решения квадратного уравнения

3. Стандартные квадратные уравнения

В начале урока вспомним что такое квадратные уравнения в стандартном виде и как их решать.

Квадратные уравнения – это уравнения типа:

$$ax^2 + bx + c = 0.$$

Вспомним метод решения таких уравнений:

1. Вычислить дискриминант. Формула для вычисления:

$$D = b^2 + 4ac$$

2. Определить по дискриминанту количество корней и формулу для дальнейшего решения. Возможны три варианта:

2.1. если $D < 0$, то уравнение не имеет корней;

2.2. если $D = 0$, то уравнение имеет один корень.

Формула для его нахождения:

$$x = \frac{-b}{2a}$$

2.3. если $D > 0$, то уравнение имеет 2 корня. Формула для нахождения корней в таком случае:

$$x = \frac{-b \pm \sqrt{D}}{2a}$$

3. Найти корни по формуле, или сделать вывод, что уравнение не имеет корней.

Далее решим пару уравнений у доски, чтобы вспомнить методы решения не в теории, а на практике.

$$x^2 - 8x + 12 = 0$$

$$5x^2 + 3x + 7 = 0$$

$$x^2 - 6x + 9 = 0$$

Следующим шагом, в формате интерактивного общения нарисуем на доске блок-схему метода решения таких уравнений, которая изображена на рисунке 4.

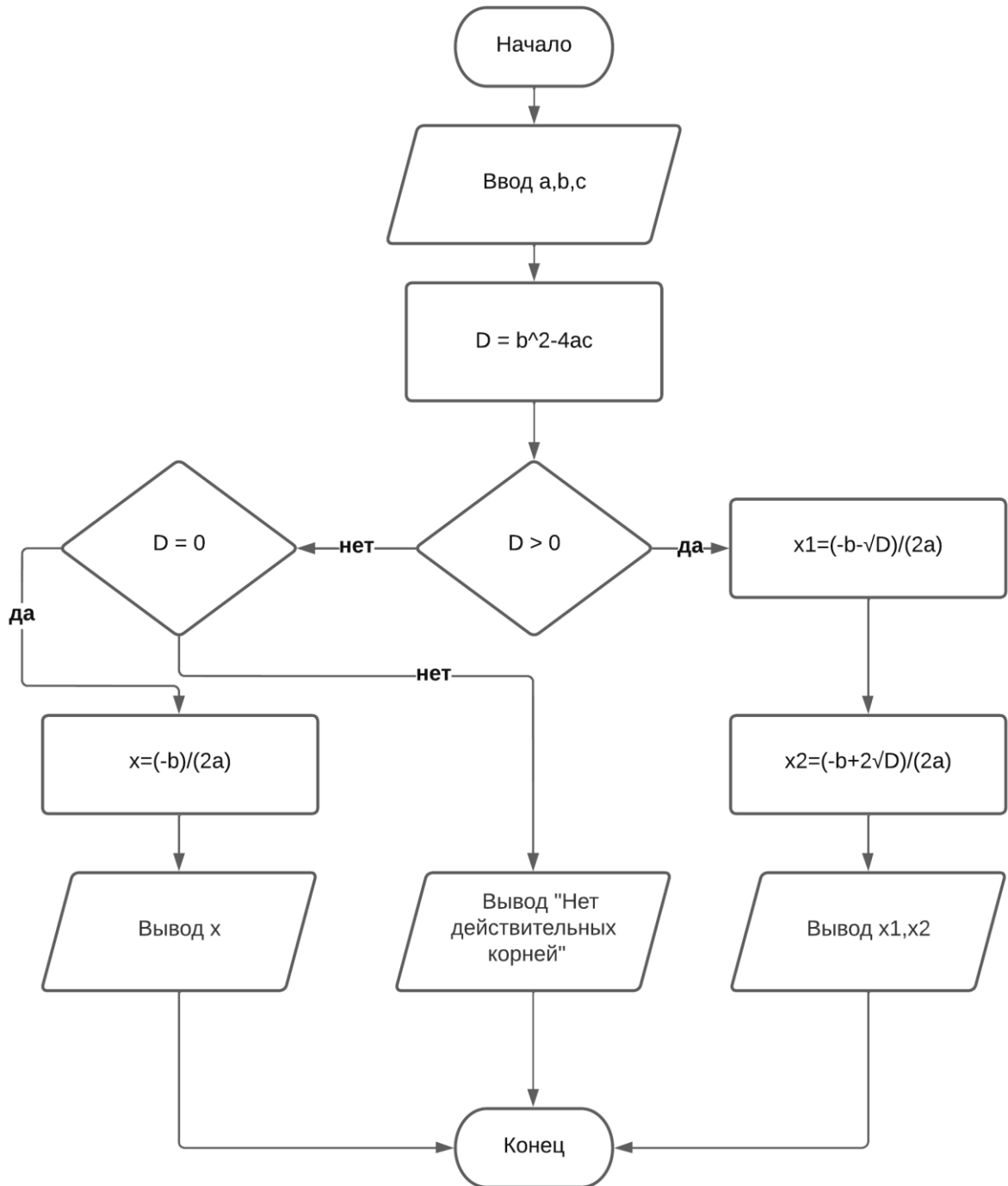


Рис. 4. Блок-схема метода решения квадратного уравнения

4. Решение неравенств второй степени с одной переменной

В начале урока разберемся что такое неравенства второй степени с одной переменной. Неравенства следующего типа:

$$ax^2 + bx + c < 0$$

$$ax^2 + bx + c > 0$$

где x – переменная; a , b , c – коэффициенты; $a \neq 0$ – называются неравенства второй степени с одной переменной.

Решением таких неравенств служат промежутки, на которых функция принимает положительные или отрицательные значения. Запишем в виде последовательности действий необходимых для нахождения этих промежутков:

1. Переходим от неравенства к уравнению

$$ax^2 + bx + c = 0$$

2. Находим корни уравнения.

3. Определяем направление веток параболы. Если $a > 0$, то ветки параболы направлены вверх. Если $a < 0$, то ветки направлены вниз.

4. Определить промежутки, удовлетворяющие условию.

- 4.1. Если у уравнения нет корней, то парабола не пересекает ось u :

- 4.1.1. Если ветви направлены вверх, то при любых x неравенство будет больше нуля.

- 4.1.1.1. Если в условии задачи было необходимо найти значения, где неравенство больше 0, то ответом будет любое x (промежуток от минус бесконечности до плюс бесконечности).

- 4.1.1.2. Если в условии задачи было необходимо найти значения, где неравенство меньше 0, то промежутков, при которых будет выполняться условие, не существует.

- 4.1.2. Если ветви направлены вниз, то при любых x неравенство будет меньше нуля.

- 4.1.2.1. Если в условии задачи было необходимо найти значения, где неравенство больше 0, то промежутков, при которых будет выполняться условие, не существует.
- 4.1.2.2. Если в условии задачи было необходимо найти значения, где неравенство меньше 0, то ответом будет любое x .
- 4.2. Если у уравнения был 1 корень, то парабола в этой точке касается ось y , но не пересекает ее. А значит в значении корня функция будет иметь значение 0.
 - 4.2.1. Если ветви направлены вверх, то при любых x кроме корня неравенство будет больше нуля.
 - 4.2.1.1. Если в условии задачи было необходимо найти значения, где неравенство больше 0, то ответом будут два промежутка: от минус бесконечности до корня; от корня до плюс бесконечности.
 - 4.2.1.2. Если в условии задачи было необходимо найти значения, где неравенство меньше 0, то промежутков, при которых будет выполняться условие, не существует.
 - 4.2.2. Если ветви направлены вниз, то при любых x кроме корня неравенство будет меньше нуля.
 - 4.2.2.1. Если в условии задачи было необходимо найти значения, где неравенство больше 0, то промежутков, при которых будет выполняться условие, не существует.
 - 4.2.2.2. Если в условии задачи было необходимо найти значения, где неравенство меньше 0, то ответом будут два промежутка: от минус бесконечности до корня; от корня до плюс бесконечности.
- 4.3. Если у уравнения было 2 корня, в этих точках ветви пересекают ось y :
 - 4.3.1. Если ветви направлены вверх.

4.3.1.1. Если в условии задачи было необходимо найти значения, где неравенство больше 0, то промежутка будет два: от минус бесконечности, до меньшего корня уравнения; от большего корня уравнения до плюс бесконечности.

4.3.1.2. Если в условии задачи было необходимо найти значения, где неравенство меньше 0, то ответом будет один промежуток: от меньшего корня уравнения до большего корня уравнения

4.3.2. Если ветви направлены вниз.

4.3.2.1. Если в условии задачи было необходимо найти значения, где неравенство больше 0, то ответом будет один промежуток: от меньшего корня уравнения до большего корня уравнения.

4.3.2.2. Если в условии задачи было необходимо найти значения, где неравенство меньше 0, то промежутка будет два: от минус бесконечности, до меньшего корня уравнения; от большего корня уравнения до плюс бесконечности.

После, решим пару примеров у доски чтобы попрактиковаться в решении данных неравенств

$$5x^2 + 9x - 2 < 0$$

$$3x^2 - 11x - 4 > 0$$

$$-5x + 11x - 6 > 0$$

$$x^2 - 3x + 4 > 0$$

На рисунке 5 изображена блок-схема для решения таких неравенств в формате интерактивного диалога.

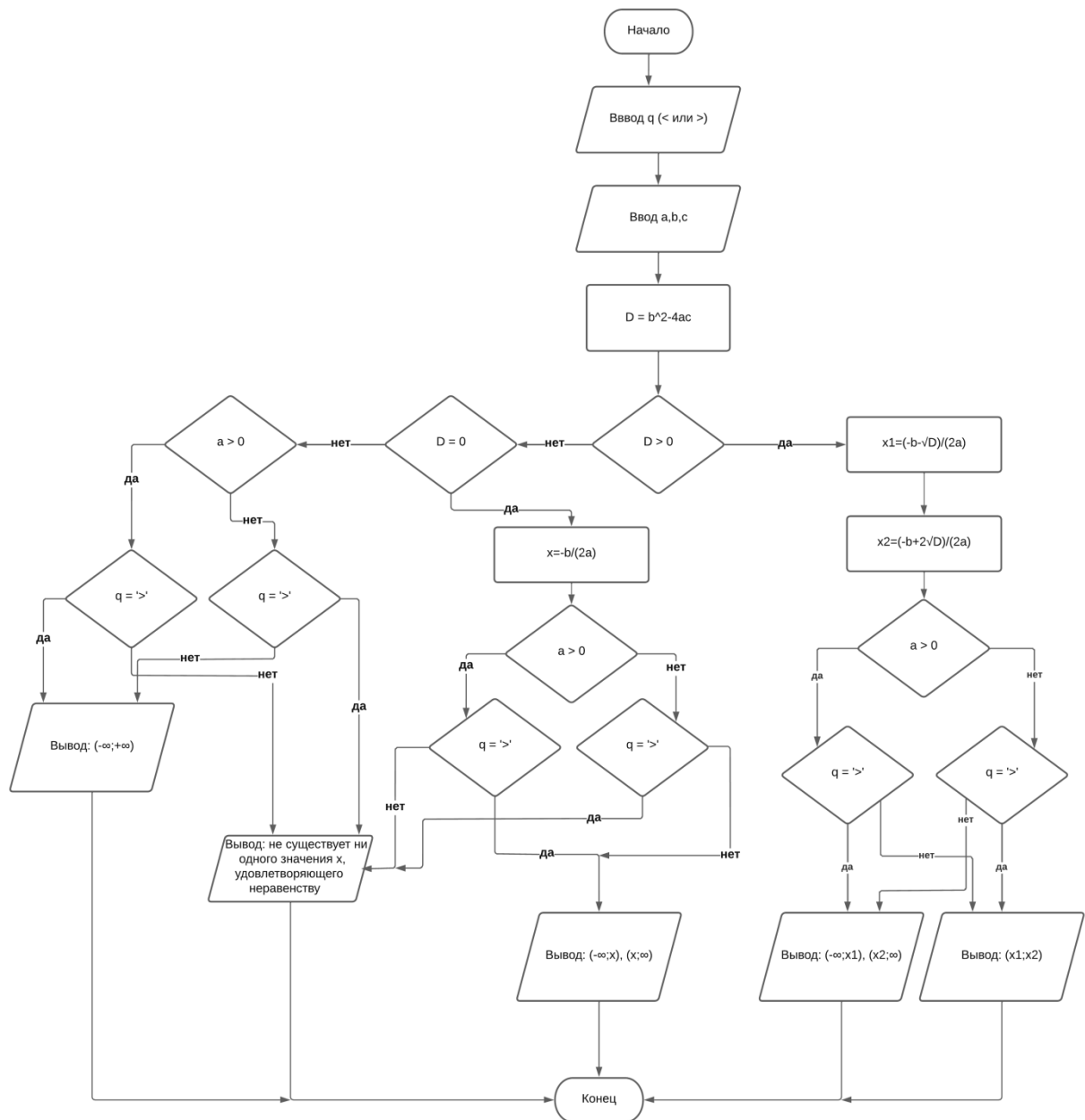


Рис. 5. Блок-схема метода решения неравенств второй степени с одной переменной

2.2.3. СОДЕРЖАНИЕ ПРАКТИЧЕСКОЙ ЧАСТИ КУРСА

1. Задачи на движение

В начале урока вспомним блок-схему(рисунок 1) из первого теоретического урока

Далее, в течение урока, ученикам ставятся задачи разработать два программных обеспечения для решения двух типов задач

На рисунках 6,7 представлен листинг кода, пример одного из возможных решений и вывод результата работы программного обеспечения

```
print("Нахождение пройденного расстояния на двух участках дороги")
u1 = float(input("Скорость на первом участке(км/час): "))
u2 = float(input("Скорость на втором участке(км/час): "))
t = float(input("Общее время движения(час): "))
t2 = float(input("Время движения на втором участке(час): "))

t1 = t - t2
s1 = t1 * u1
s2 = t2 * u2
s = s1 + s2

print("Общее преодоленное расстояние: %.2fкм" % s)
```

Рис. 6. Листинг кода решения задачи первого типа

```
Нахождение пройденного расстояния на двух участках дороги
Скорость на первом участке(км/час): 23
Скорость на втором участке(км/час): 26
Общее время движения(час): 7
Время движения на втором участке(час): 4
Общее преодоленное расстояние: 173.00км
```

Рис.7. Результат работы программного обеспечения

На рисунке 8 представлен листинг кода и результат для задачи второго типа.

```
print("Нахождение расстояния между двумя движущимися объектами через час")
s0 = float(input("Расстояние между объектами(км): "))
u1 = float(input("Скорость первого объекта(км/час): "))
u2 = float(input("Скорость второго объекта(км/час): "))

u = u1+u2
s=s0+u
print("Если объекты едут в разные стороны друг от друга, то расстояние между ними: %.2fкм" % s)

u = u1+u2
s=s0-u
print("Если объекты едут в разные стороны друг к друга, то расстояние между ними: %.2fкм" % s)

u = u1-u2
s=s0+u
print("Если объекты едут в сторону первого объекта, то расстояние между ними: %.2fкм" % s)

u = u2-u1
s=s0+u
print("Если объекты едут в сторону второго объекта, то расстояние между ними: %.2fкм" % s)
```

Рис. 8. Листинг кода решения задачи второго типа

На рисунке 9 представлен результат выполненной работы.

Нахождение расстояния между двумя движущимися объектами через час
 Расстояние между объектами(км): 400
 Скорость первого объекта(км/час): 70
 Скорость второго объекта(км/час): 90
 Если объекты едут в разные стороны друг от друга, то расстояние между ними: 560.00км
 Если объекты едут в разные стороны друг к друга, то расстояние между ними: 240.00км
 Если объекты едут в сторону первого объекта, то расстояние между ними: 380.00км
 Если объекты едут в сторону второго объекта, то расстояние между ними: 420.00км

Рис.9. Результат работы программного обеспечения

План-конспект учебного занятия

Тема учебного занятия: Задачи на движение.

Вид учебного занятия: Практическое.

Тип учебного занятия: Самостоятельная.

Цель учебного занятия: Написать программное обеспечение для решения задач на движение двух типов.

Задачи учебного занятия:

Образовательные:

- Научится переносить блок-схемы на язык программирования python;
- научиться основным конструкциям языков программирования на основе языка python.

Воспитательные:

- Привитие интереса к разработке программного обеспечения;
- привитие интереса языку программирования python;
- привитие самостоятельности в решении поставленных задач.

Развивающие:

- Развитие логического мышления;
- развитие интеллектуальных способностей учащихся.

Технические средства обучения: автоматизированное рабочее место преподавателя (ПК, интерактивная доска), автоматизированные рабочие места для учащихся (ПК).

Ход занятия

1) Организационный момент

1. Приветствие.
2. Отметка присутствующих.
3. Постановка целей занятия.

Темой сегодняшнего занятия является «Задачи на движения»

2) Постановка индивидуального задания учащимся

На данном уроке необходимо разработать два программных обеспечения на языке программирования python, которые будут решать задания аналогичные тем, что были на теоретическом уроке. Программное обеспечение должно быть реализовано согласно блок-схемам составленных на теоретическом уроке. Так же необходимо реализовать ввод различных входных данных и работу с ними.

3) Проверка индивидуальных заданий

За 10 минут до конца урока начинается проверка выполненного задания. По результатам проверки будет ставиться оценка за урок. Учащиеся могут заранее попросить проверить их работу если справились с ней заблаговременно.

4) Подведение итогов

Подводятся итоги занятия, выставляются баллы.

2. Числа Фибоначчи

В начале урока вспомним блок-схему из первого теоретического урока

Затем в течение урока ученикам ставятся 2 задачи:

1. Написать программное обеспечение для нахождения n -ого числа последовательности
2. Нарисовать блок схему для следующей задачи. С клавиатуры вводится число m , где $m > 0$. Определить при каком минимальном n количество цифр в числе станут равны m .

На рисунке 10 предоставлен листинг кода, пример одного из вариантов правильного решения первой задачи.

```
import math

print("Нахождение n-ого числа последовательности Фибоначчи")
n = int(input("n = ")) - 1
if n < 0:
    print("Введен некорректный n")
    exit()
m = 1
F = [0,1]
while m < n:
    m = m + 1
    F[m:m] = [F[m-2] + F[m-1]]

print("Искомое значение: " +str( F[n]))
```

Рис. 10. Листинг кода решения задачи

На рисунке 11 представлен скриншот результата работы такого программного обеспечения. Число n вводится с клавиатуры.

```
Нахождение n-ого числа последовательности Фибоначчи
n = 60
Искомое значение: 956722026041
```

Рис. 11. Результат работы программного обеспечения

Блок-схема для задания 2 будет выглядеть так, как изображена на рисунке 12.

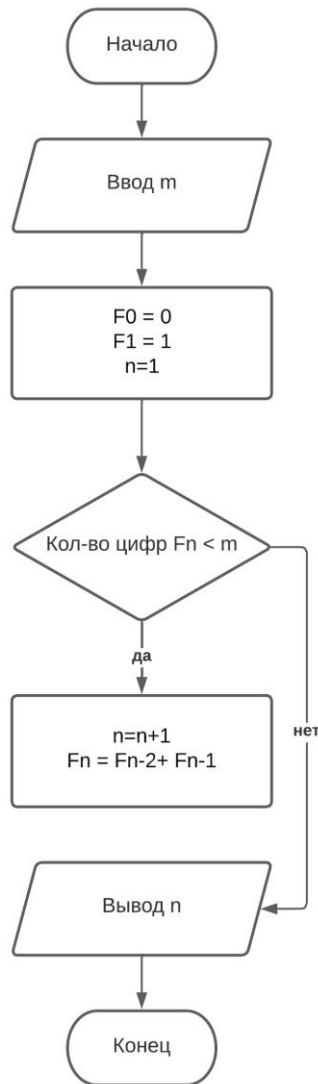


Рис. 12. Ожидаемый результат второго задания

3. Стандартные квадратные уравнения

В начале урока вспомним блок-схему из первого теоретического урока (рисунок 1)

Затем в рамках урока ученикам предоставляется время для написания программного обеспечения для решения квадратный уравнений. На рисунке 13 предоставлен листинг кода, пример одного из вариантов правильного решения данной задачи.

```

import math

print("Решение уравнения типа: ax^2 + bx + c = 0:")
print("Введите коэффициенты для уравнения")
a = float(input("a = "))
b = float(input("b = "))
c = float(input("c = "))

D = b ** 2 - 4 * a * c
print("Дискриминант D = %.2f" %D)

if D > 0:
    x1 = (-b + math.sqrt(D)) / (2 * a)
    x2 = (-b - math.sqrt(D)) / (2 * a)
    print("x1 = %.2f \nx2 = %.2f" % (x1, x2))
elif D == 0:
    x = -b / (2 * a)
    print("x = %.2f" % x)
else:
    print("Корней нет")

```

Рис. 13. Листинг кода решения задачи

На рисунке 14 представлен скриншот результата работы такого программного обеспечения. Коэффициенты a , b , c вводятся с клавиатуры.

```

Решение уравнения типа: ax^2 + bx + c = 0:
Введите коэффициенты для уравнения
a = 1
b = -8
c = 12
Дискриминант D = 16.00
x1 = 6.00
x2 = 2.00

```

Рис.14. Результат выполнения программного обеспечения

4. Решение неравенств второй степени с одной переменной

В начале урока вспомним блок-схему из первого теоретического урока (рисунок 1)

Затем в рамках урока ученикам предоставляется время для написания программного обеспечения для решения квадратный уравнений. На рисунке 15 предоставлен листинг кода, пример одного из вариантов правильного решения данной задачи.

```

import math
def getAnswer(firstString, secondString):
    if a > 0:
        if q == '>':
            return firstString
        else:
            return secondString
    else:
        if q == '>':
            return secondString
        else:
            return firstString

print("Решение неравенств типа  $ax^2 + bx + c > 0$  или  $ax^2 + bx + c < 0$ :")
print("Введите знак неравенства (только < или >)")
q = input("q = ")
if q not in ['<', '>']:
    print('Знак неравенства введен некорректно')
    exit()
print("Введите коэффициенты для уравнения")
a = float(input("a = "))
b = float(input("b = "))
c = float(input("c = "))

D = b ** 2 - 4 * a * c
print("Дискриминант D = %.2f" % D)

if D > 0:
    x1 = (-b - math.sqrt(D)) / (2 * a)
    x2 = (-b + math.sqrt(D)) / (2 * a)
    print(getAnswer("Ответ:  $(-\infty; %.2f), (%.2f; +\infty)$ " % (x1, x2), "Ответ:  $(%.2f; %.2f)$ " % (x1, x2)))
elif D == 0:
    x = -b / (2 * a)
    print(getAnswer("Ответ:  $(-\infty; %.2f), (%.2f; +\infty)$ " % (x, x), "Не существует ни одного значения x, удовлетворяющего неравенству"))
else:
    print(getAnswer("Ответ:  $(-\infty; +\infty)$ ", "Не существует ни одного значения x, удовлетворяющего неравенству"))

```

Рис.15. Листинг кода решения задачи

На рисунке 16 представлен скриншот результата работы такого программного обеспечения. Знак неравенства q и коэффициенты a , b , c вводятся с клавиатуры.

```

Решение неравенств типа  $ax^2 + bx + c > 0$  или  $ax^2 + bx + c < 0$ :
Введите знак неравенства (только < или >)
q = <
Введите коэффициенты для уравнения
a = 5
b = 9
c = -2
Дискриминант D = 121.00
5.0
Ответ: (-2.00;0.20)

```

Рис.16. Результат выполнения программного обеспечения

2.2.4. СОДЕРЖАНИЕ КОНТРОЛЬНОЙ ЧАСТИ КУРСА

В начале урока ученики делятся на группы по 3-4 человека, каждой группе выдаются задания. Затем ученики обсуждают задания, выдвигают способы решения, слушают друг друга и выбирают верный ответ. Ученикам выдается ватман, где они рисуют блок-схемы и даётся время на написание программы. Затем один из группы защищает работу перед другими учениками.

1. Нарисовать блок-схему подсчета n -ого числа в последовательности Фибоначчи
2. Нарисовать блок-схему решения квадратного уравнения
3. Написать программное обеспечение для решения задачи на движение с двумя объектами. Для проверки можно использовать следующую задачу: расстояние между двумя объектами 400 км, первый объект движется со скоростью 80 км/час, второй 100 км/час. Чему будет равно расстояние между объектами через час. Необходимо рассмотреть все 4 случая, разных направлений объектов.

Результаты на этих данных должны быть следующие:

Если объекты едут в разные стороны друг от друга, то расстояние между ними: 560.00км

Если объекты едут в разные стороны друг к друга, то расстояние между ними: 240.00км

Если объекты едут в сторону первого объекта, то расстояние между ними: 380.00км

Если объекты едут в сторону второго объекта, то расстояние между ними: 420.00

Данная контрольная работа поможет развить коммуникативные способности, не списывать готовые ответы с интернета, чтоб просто получить хорошую оценку, а поучаствовать с другими учениками в дискуссиях.

ВЫВОДЫ ПО ГЛАВЕ 2

Результатом служит готовый элективный курс для 7-9 классов на 17 часов, 8 часов из которых занимают теоретические уроки, 8 часов – практические уроки и 1 час – контрольная работа.

Курс поможет понять некоторые темы курса математики, так как написание алгоритма решения задач в виде блок-схем дает возможность посмотреть на математические задачи под другим углом. В школах учащиеся часто забывают, как решать те или иные математические задачи так как им дали немного теории, и они решили пару примеров. Такие знания забываются в скором времени если их не использовать. Наглядный пример в виде блок-схемы запоминается куда лучше, чем в виде текста.

Также курс направлен на развитие логического мышления, он позволит учащимся в будущем лучше запоминать тот или иной материал, даже не относящийся к математике. Это связано с тем, что развитое логическое мышление позволит хорошо структурировать информацию в голове и лучше запомнить ее.

Еще одна из важных особенностей курса, это получения актуальных знаний и навыков. Информационные технологии развиваются большими шагами, проанализировав сервис для поиска работы HeadHunter можно сделать вывод, что специалистов в данной области не хватает. Также изучив

данный сервис можно сделать вывод, что python входит в тройку самых востребованных языков программирования.

ЗАКЛЮЧЕНИЕ

В ходе исследования цель была достигнута, поставленные задачи решены, получены следующие результаты и выводы:

- уточнены определения «компетенция», «компетентность», «учебная компетенция»;
- выделены структурные компоненты УК;
- рассмотрены современные педагогические технологии на уроках математики;
- проанализированы проблемы интеграции математики и программирования в средней школе;
- спроектирована модель формирования учебных компетенций обучающихся в процессе обучения математике и программирования на основе их интеграции;
- реализована модель применения интеграции математики и программирования на python для проектирования элективного курса по формированию учебных компетенций учащихся;
- спроектирован элективный курс по формированию учебных компетенций
- проанализированы, обработаны и подведены итоги проведения эксперимента.

Дальнейшее исследование может быть связано с созданием методики развития УК обучающихся и ее реализации в старшей школе. Также в рамках дальнейшего развития темы исследования можно учесть недостаточность литературы по программированию на python для средней школы, а также отсутствие видеоуроков, обучающих материалов и в будущем обратить на это внимание.

БИБЛИОГРАФИЧЕСКИЙ СПИСОК

1. Edunews. Что такое элективный курс и зачем он нужен. URL: <https://edunews.ru/school/info/elektivnyj-kurs.html>(дата обращения: 17.11.2021).
2. Python: преимущества и недостатки// советы экспертов.URL: <https://www.goldenpages.ua/expert/?p=10627> (дата обращения: 10.12.2021)
3. Web-creator. Серверные языки программирования. URL: https://web-creator.ru/articles/server_side_languages (дата обращения: 17.11.2021).
4. А.М. Матюшкин, А.А. Матюшкина, И.А. Зимняя. Проблемное обучение: прошлое, настоящее, будущее: коллективная монография в 3 кн. Кн. 2: Лингво-педагогические модели проблемного обучения, 2019. 310 с. URL: <https://www.elibrary.ru/item.asp?id=37189084> (дата обращения: 23.01.2022).
5. Абрамовских В.В. Язык программирования Python в качестве дополнительного языка программирования в школе. URL: <https://konference.nvsu.ru> (дата обращения: 10.12.2021).
6. Автор 24. Системно-деятельностный подход в условиях реализации ФГОС.URL: https://spravochnick.ru/pedagogika/sistemno-deyatelnostnyu_podhod_v_usloviyah_realizacii_fgos/(дата обращения 21.11.2021)
7. Брич О. В. Характеристика учебно-познавательной компетенции как компонента иноязычной коммуникативной компетенции. М.: Вестн. МГЛУ. Сер.1. Педагогика. Методика. Психология, 2013. № 1. с. 65-76.
8. Воровщиков С.Г. Учебно-познавательная компетентность старшеклассников: состав и структура. М.: Наука и школа, 2007. №1. с. 36-38. URL: <https://cyberleninka.ru/article/n/uchebno-poznavatel'naya-kompetentnost-starsheklassnikov-sostav-i-struktura> (дата обращения: 21.10.2018).

9. Гурьянова М.В., Аброскин А.С. Ведущие языки программирования. URL: <https://www.elibrary.ru/item.asp?id=42560809> (дата обращения: 18.11.2021).

10. Дмитриенко М.В. Элективные курсы: формирование профессиональных интересов и намерений старшеклассников. URL: <https://cyberleninka.ru/article/n/elektivnye-kursy-formirovanie-professionalnyh-interesov-i-namereniy-starsheklassnikov>(дата обращения: 17.11.2021).

11. Донецкий Республиканский институт. Чем элективный курс отличается от факультативного? URL: http://donippo.blogspot.com/2015/09/blog-post_8.html(дата обращения: 17.11.2021).

12. Дьячук П.П. Системообразующие факторы интеграции курсов алгебры и информатики в средней школе // URL: <https://cyberleninka.ru/article/n/sistemoobrazuyuschie-factory-integratsii-kurov-algebry-i-informatiki-v-sredney-shkole/viewer> (дата обращения: 28.03.2022).

13. Зак А.З. Развитие и диагностика мышления подростков и старшеклассников. М.; Обнинск: ИГ-СОЦИН, 2010. 350 с. (дата обращения: 12.04.2022).

14. Захарова Т. Б., Нижников А.П., Маняхина В. Г., Смирнова И. М., Мирзоев М. С. Актуальные проблемы обучения математике и информатике в школе и педагогическом вузе. М.: Прометей, 2017. (дата обращения: 10.04.2022).

15. Зверев И.Д., Максимова В.Н. Межпредметные связи в современной школе. М.: Педагогика, 1981. 159 с. (дата обращения: 20.04.2022).

16. Зимняя И.А. Компетенция и компетентность в контексте компетентностного подхода в образовании. М.:НП «Национальное общество прикладной лингвистики». URL: <https://www.elibrary.ru/item.asp?id=21611467> (дата обращения: 27.02.2022).

17. И.Б.Адушева, Н.Б. Смирнова. К проблеме реализации системно-деятельностного подхода общеобразовательной школе. URL: <https://cyberleninka.ru/article/n/k-probleme-realizatsii-sistemno-deyatelnostnogo-podhoda-v-obscheobrazovatelnoy-shkole> (дата обращения:23.02.2022)

18. Капин А.В. Математика как наука о количественных отношениях и пространственных формах действительного мира. URL: <https://urokimatematiki.ru/matematika-kak-nauka-o-kolichestvennih-otnosheniyah-i-prostranstvennih-formah-deystvitelnogo-mira-4187.html>(дата обращения: 18.103.2022).

19. Козлов С. В. Особенности обучения школьников информатике в профильной школе // Научно-методический электронный журнал «Концепт». 2014. Раздел 1. С. 31-35. URL: <http://e-koncept.ru/2014/14006.htm> (дата обращения: 12.01.2022).

20. Компьютерный портал. Язык python. Как изучать машинное обучение? Как запустить написанную программу. URL: <https://crabo.ru/hardware-solutions/yazyk-python-kak-izuchat-mashinnoe-obuchenie-kak-zapustit.html>(дата обращения: 18.02.2022).

21. Лебедев О.Е. Компетентностный подход в образовании. М.: Школьные технологии, 2004. Раздел № 5. с. 3-12. URL: http://pedlib.ru/Books/3/0389/3_0389-1.shtml (дата обращения: 02.03.2022).

22. Мартынюк А. А. Интегрированный подход при обучении программированию в средней школе // Интерактивная наука. 2018. Раздел №2. URL: <https://cyberleninka.ru/article/n/integrirrovannyy-podhod-pri-obuchении-programmirovaniyu-v-sredney-shkole> (дата обращения: 06.04.2022).

23. Н. В. Макарова, Ю. Ф. Титова. Системно-деятельностный подход при обучении информатики в средней школе. URL: <https://cyberleninka.ru/article/n/sistemno-deyatelnostnyy-podhod-pri-obuchении-informatike-v-sredney-shkole>(дата обращения:22.03.2022)

24. О.В.Попова, Е.В. Рекк. Интеграция информатики и математики. URL: <https://cyberleninka.ru/article/n/integratsiya-informatiki-i-matematiki> (дата обращения: 21.11.2021)

25. Общие требования к выпускным квалификационным работам бакалавра, специалиста, магистра в Тюменском государственном университете (приказ № 12-1 от 20.01.2020, действует с 01.04.2020) URL: https://www.utmn.ru/upload/medialibrary/01f/12_1-_1_.PDF (дата обращения: 12.10.2021).

26. Перминов В.Я. Философия и основания математики. М., «Прогресс -Традиция», 2002. 320 с. (дата обращения: 18.11.2021).

27. Положение о проверке на объем заимствования и размещении в электронной библиотеке выпускных квалификационных работ и научных докладов об основных результатах подготовленных научно-квалификационных работ (диссертаций) в ФГАОУ ВО «Тюменский государственный университет» (приказ № 97-1 от 26.02.2018) URL: https://www.utmn.ru/upload/medialibrary/891/97_1.PDF (дата обращения: 12.10.2021).

28. Попов С.А Использование языка программирования Python для обучения школьников программированию в базовом курсе информатики и ИКТ. URL: http://www2.bigpi.biysk.ru/vkr2018/file/fii_21_06_2019_08_54_25.pdf (дата обращения: 10.12.2021).

29. Савкина Л.В. К вопросу об обучении программированию в основной и средней школе // Современные информационные технологии и ИТ-образование. 2011. №7. URL: <https://cyberleninka.ru/article/n/k-voprosu-ob-obuchenii-programmirovaniyu-v-osnovnoy-i-sredney-shkole> (дата обращения: 15.12.2021).

30. Сафонов В.И. Пути использования компьютерных программных средств при изучении математики в средней школе. URL: <https://cyberleninka.ru/article/n/puti-ispolzovaniya-kompyuternyh->

programmnyh-sredstv-pri-izuchenii-matematiki-v-sredney-shkole (дата обращения: 28.12.2021).

31. Системно-деятельностный подход в педагогическом образовании: опыт реализации и перспективы развития Материалы Всероссийской научнопрактической конференции. Белгород: Тип. «Принт-Мастер», 2018. 468с. (дата обращения: 20. 11.2021).

32. Сорокина Н.А. Python как основной язык программирования в средней школе / Н. А. Сорокина. Текст: непосредственный // Молодой ученый. 2019. № 5 (243). С. 15-16. URL: <https://moluch.ru/archive/243/56193/> (дата обращения: 25.02.2022).

33. Студопедия. Современное состояние математики как науки. Математика в системе наук URL: https://studopedia.ru/19_258066_sovremennoe-sostoyanie-matematiki-kak-nauki-matematika-v-sisteme-nauk.html/ (дата обращения: 17.11.2021).

34. Сухова Л.Н. Актуальные проблемы преподавания математики в средней школе. URL: <https://multiurok.ru/blog/aktual-nyie-probliemy-priepodavaniia-matiematiki-v-sriedniei-shkolie.html> (дата обращения: 17.11.2021).

35. Учебное пособие М.Н.Волкова. Деятельностный подход и категория деятельности в психологии. URL: https://msun.ru/upload/folders/edu_lit/kaf/phsihology/002.pdf (дата обращения: 19.11.2021).

36. Фардиева Р.Р. Актуальные проблемы преподавания математики. URL: <https://core.ac.uk/download/pdf/197422603.pdf> (дата обращения: 10.102.2022).

37. Федеральный государственный образовательный стандарт основного общего образования [Текст] / Министерство образования и науки Российской Федерации. — М.: Просвещение, 2014. 48 с. URL:

<https://mosmetod.ru/metodicheskoe-prostranstvo/srednyaya-i-starshaya-shkola/russkij-yazyk/fgos/fgos-ooo.html>(дата обращения: 18. 01.2022).

38. Фирмаль Л. Математика задачи с решением. URL: <https://9219603113.com/reshenie-zadach-po-matematike> (дата обращения: 22.01.2022).

39. Формы титульных листов выпускных квалификационных работ бакалавра, специалиста, магистра, специалиста среднего звена в Тюменском государственном университете на период сложной санитарно-эпидемиологической обстановки (приказ № 264-1 от 08.05.2020) URL: https://www.utmn.ru/upload/medialibrary/7b8/264_1.pdf (дата обращения: 12. 02.2022).

40. Хуторской А.В. Ключевые компетенции: технология конструирования. М.: Народное образование, 2003. №5(1328). с. 55-61. URL: <https://www.elibrary.ru/item.asp?id=21696549> (дата обращения: 16.10.2021).

41. Хуторской А.В. Определение общепредметного содержания и ключевых компетенций как характеристика нового подхода к конструированию образовательных стандартов. М.: Вестник Института образования человека, 2011. №1. 3 с. URL: <https://www.elibrary.ru/item.asp?id=24069404> (дата обращения: 25.09.2021).

42. Хуторской А.В. Образовательные компетенции и методология дидактики. К 90-летию со дня рождения В.В. Краевского // А.В.Хуторской. Персональный сайт – Хроника бытия: [сайт]. 2016. URL: <http://khutorskoj.ru/be/2016/0803/> (дата обращения: 28.01.2022).

43. Школа программной инженерии. Software Engineering school - «Основы программирования на языке Python». URL: <https://seschool.ru/python> (дата обращения: 18.11.2020).

44. Шоу Б. Единственный путь, ведущий к знанию – это деятельность. URL: <http://kosimowa-e.narod.ru/sisemdejatpodhod.pdf> (дата обращения: 18.11.2021).

45. Шумейко О.Н. Реализация системно-деятельностного подхода
URL:<https://moluch.ru/conf/ped/archive/188/9804/> (дата обращения: 16.11.2021).