

МИНИСТЕРСТВО ПРОСВЕЩЕНИЯ РОССИЙСКОЙ ФЕДЕРАЦИИ

Министерство образования, науки и молодежной политики Республики Коми

Управление образования МО ГО "Сыктывкар"

МОУ "ООШ № 34" г.Сыктывкара

РАССМОТРЕНО

Педагогическим советом

Протокол № 13

от «30» августа 2023 г.

УТВЕРЖДЕНО

Приказом № 360

от «30» августа 2023 г.

РАБОЧАЯ ПРОГРАММА

учебного предмета

«Информатика»

для 7-9 классов основного общего образования

Составитель: Покровкова Людмила Александровна

учитель информатики

Содержание

1. Пояснительная записка.....	3
2. Планируемые результаты изучения учебного предмета.....	10
3. Содержание учебного предмета.....	17
4. Тематическое планирование.....	25
5. Приложения.....	57

1. Пояснительная записка

Рабочая программа по предмету «**Информатика**» разработана для обучения учащихся 7-9 классов МОУ «Основная общеобразовательная школа № 34» г. Сыктывкара» (далее - МОУ «ООШ № 34» г. Сыктывкара) **в соответствии с:**

- Приказом Минпросвещения России от 18.05.2023 N 370 «Об утверждении федеральной образовательной программы основного общего образования» (Зарегистрировано в Минюсте России 12.07.2023 N 74223)
- Федеральным государственным образовательным стандартом основного общего образования, утверждённым приказом Министерства образования и науки Российской Федерации от 17 декабря 2010 г № 1897 в действующей редакции;
- Приказом Минпросвещения России от 11.12.2020 г. №712 «О внесении изменений в некоторые федеральные государственные образовательные стандарты общего образования по вопросам воспитания обучающихся».
- Приказом Минпросвещения России от 18.05.2023 № 370 «Об утверждении федеральной образовательной программы основного общего образования»

На основе:

- Требований к результатам освоения основной образовательной программы основного общего образования МОУ «ООШ № 34» г. Сыктывкара;
- Методических рекомендаций МУ ДПО «ЦРО» по доработке рабочих программ учебных предметов в связи с рабочей программой воспитания.
- Положения о рабочей программе учебного предмета, утвержденного приказом МОУ «ООШ №34» г. Сыктывкара
- УМК Босовой Л.Л., Босовой А.Ю. «Информатика».

С учетом:

- Примерной основной образовательной программы основного общего образования (протокол № 1/20 от 04.02.2020);
- Авторской программы: Информатика. Программа для основной школы: 5-6 классы. 7-9 классы / Л, Л. Босова, А. Ю. Босова. - М.: БИНОМ. Лаборатория знаний, 2014.

При реализации РПУП побуждение обучающихся соблюдать на уроке общепринятые нормы поведения, правила общения осуществляется посредством следования правилам, вытекающих из ценностей школы, выработка и принятие которых описаны в РПВ (модуль «Школьный урок»).

Программа по информатике для основной школы составлена в соответствии с: требованиями Федерального государственного образовательного стандарта основного общего образования (ФГОС ООО); требованиями к результатам освоения основной образовательной программы (личностным, метапредметным, предметным); основными подходами к развитию и формированию универсальных учебных действий (УУД) для основного общего образования. В ней соблюдается преемственность с федеральным государственным образовательным стандартом начального общего образования; учитываются возрастные и психологические особенности школьников, обучающихся на ступени основного общего образования, учитываются межпредметные связи.

Методологической основой федеральных государственных образовательных стандартов является системно-деятельностный подход, в рамках которого реализуются современные стратегии обучения, предполагающие использование информационных и коммуникационных технологий (ИКТ) в процессе изучения всех предметов, во внеурочной и внешкольной деятельности на протяжении всего периода обучения в школе. Организация учебно-воспитательного процесса в современной информационно-образовательной среде является необходимым условием формирования информационной культуры современного школьника, достижения им ряда образовательных результатов, прямо связанных с необходимостью использования информационных и коммуникационных технологий.

Средства ИКТ не только обеспечивают образование с использованием той же технологии, которую учащиеся применяют для связи и развлечений вне школы (что важно само по себе с точки зрения социализации учащихся в современном информационном обществе), но и создают условия для индивидуализации учебного процесса, повышения его эффективности и результативности. На протяжении всего периода существования школьного курса информатики преподавание этого предмета было тесно связано с информатизацией школьного образования: именно в рамках курса информатики школьники знакомились с теоретическими основами информационных технологий, овладевали практическими навыками использования средств ИКТ, которые потенциально могли применять при изучении других школьных предметов и в повседневной жизни.

Изучение информатики в 7–9 классах вносит значительный вклад в достижение **главных целей** основного общего образования, способствуя:

- **формированию целостного мировоззрения**, соответствующего современному уровню развития науки и общественной практики за счет развития представлений об информации как важнейшем стратегическом ресурсе развития личности, государства, общества; понимания роли информационных процессов в современном мире;

• **совершенствованию общеучебных и общекультурных навыков работы с информацией** в процессе систематизации и обобщения имеющихся и получения новых знаний, умений и способов деятельности в области информатики и ИКТ; развитию навыков самостоятельной учебной деятельности школьников (учебного проектирования, моделирования, исследовательской деятельности и т.д.);

• **воспитанию ответственного и избирательного отношения к информации** с учетом правовых и этических аспектов ее распространения, воспитанию стремления к продолжению образования и созидательной деятельности с применением средств ИКТ.

При изучении учебного предмета **«Информатика»** используются следующие **формы текущего контроля успеваемости**: устный (ответы на вопросы) и письменный (самостоятельная работа, проверочная работа, контрольная работа, тест, диктант, практическая работа).

В конце учебного года проводится **промежуточная аттестация** в форме контрольной работы.

Система оценки достижения планируемых результатов

Система оценки включает процедуры внутренней и внешней оценки.

Внутренняя оценка включает:

- текущую и тематическую оценку,
- внутришкольный мониторинг образовательных достижений,
- промежуточную и итоговую аттестацию обучающихся.

внешняя оценка:

- государственная итоговая аттестация,
- независимая оценка качества образования и
- мониторинговые исследования муниципального, регионального и федерального уровней.

В соответствии с ФГОС ООО система оценки образовательной организации реализует **системно-деятельностный, уровневый и комплексный подходы** к оценке образовательных достижений.

Системно-деятельностный подход к оценке образовательных достижений проявляется в оценке способности учащихся к решению учебно-познавательных и учебно-практических задач. Он обеспечивается содержанием и критериями оценки, в качестве которых выступают планируемые результаты обучения, выраженные в деятельностной форме.

Уровневый подход служит важнейшей основой для организации индивидуальной работы с учащимися. Он реализуется как по отношению к содержанию оценки, так и к представлению и интерпретации результатов измерений.

Уровневый подход к содержанию оценки обеспечивается структурой планируемых результатов, в которых выделены три блока: общецелевой, «Выпускник научится» и «Выпускник получит возможность научиться». Достижение планируемых результатов, отнесенных к блоку «Выпускник научится», выносится на итоговую оценку, которая может осуществляться как в ходе обучения, так и в конце обучения, в том числе – в форме государственной итоговой аттестации.

Уровневый подход к представлению и интерпретации результатов реализуется за счет фиксации различных уровней достижения обучающимися планируемых результатов: *базового уровня и уровней выше и ниже базового.*

Овладение базовым уровнем является достаточным для продолжения обучения и усвоения последующего материала.

Комплексный подход к оценке образовательных достижений реализуется путём

- оценки трёх групп результатов: предметных, личностных, метапредметных (регулятивных, коммуникативных и познавательных универсальных учебных действий);
- использования комплекса оценочных процедур (стартовой, текущей, тематической, промежуточной) как основы для оценки динамики индивидуальных образовательных достижений (индивидуального прогресса) и для итоговой оценки;
- использования контекстной информации (об особенностях обучающихся, условиях и процессе обучения и др.) для интерпретации полученных результатов в целях управления качеством образования;
- использования разнообразных методов и форм оценки, взаимно дополняющих друг друга (стандартизированных устных и письменных работ, проектов, практических работ, самооценки, наблюдения и др.).

Критерии оценивания различных форм работы обучающихся на уроке.

Тематический контроль осуществляется по завершении крупного блока (темы). Он позволяет оценить знания и умения учащихся, полученные в ходе достаточно продолжительного периода работы.

Итоговый контроль осуществляется по завершении каждого года обучения.

Основная форма контроля – тестирование.

Правила при оценивании:

- за каждый правильный ответ начисляется 1 балл;
- за каждый ошибочный ответ начисляется штраф в 1 балл;

- за вопрос, оставленный без ответа (пропущенный вопрос), ничего не начисляется.

Такой подход позволяет добиться вдумчивого отношения к тестированию, позволяет сформировать у школьников навыки самооценки и ответственного отношения к собственному выбору. Тем не менее, учитель может отказаться от начисления штрафных баллов, особенно на начальном этапе тестирования.

Критерии и нормы оценки знаний, умений и навыков учащихся

Оценка устных ответов учащихся

Ответ оценивается отметкой «5» если ученик:

- полно раскрыл содержание материала в объеме, предусмотренном программой и учебником;
- изложил материал грамотным языком в определенной логической последовательности, точно используя терминологию и символику;
- правильно выполнил рисунки, графики, сопутствующие ответу;
- показал умение иллюстрировать теоретические положения конкретными примерами, применять их в новой ситуации при выполнении практического задания;
- продемонстрировал усвоение ранее изученных сопутствующих вопросов, сформированность и устойчивость используемых при ответе умений и навыков;
- отвечал самостоятельно без наводящих вопросов учителя.

Возможны одна-две неточности при освещении второстепенных вопросов или выкладках, которые ученик легко исправил по замечанию учителя;

Ответ оценивается отметкой «4» если он удовлетворяет в основном требованиям на отметку «5», но при этом имеет один из недостатков:

- в изложении допущены незначительные пробелы, не искажившие содержание ответа;
- допущены один-два недочета при освещении основного содержания ответа, легко исправленные по замечанию учителя;
- допущены ошибка или более двух недочетов при освещении второстепенных вопросов или выкладках, легко исправленные по замечанию учителя.

Отметка «3» ставится в следующих случаях:

- неполно или непоследовательно раскрыто содержание материала, но показано общее понимание вопроса и продемонстрированы умения, достаточные для дальнейшего усвоения программного материала (определенные требованиями к подготовке учащихся).

Отметка «2» ставится в следующих случаях:

- не раскрыто основное содержание учебного материала;

- обнаружено незнание или неполное понимание учеником большей или наиболее важной части учебного материала;
- допущены ошибки в определении понятий, при использовании математической терминологии, в рисунках, чертежах или графиках, в выкладках, которые не исправлены после нескольких

Критерии оценок при выполнении практических заданий:

Оценка «5» - работа выполнена в заданное время, самостоятельно, с соблюдением технологической последовательности, качественно и творчески;

Оценка «4» - работа выполнена в заданное время, самостоятельно, с соблюдением технологической последовательности, при выполнении отдельных операций допущены небольшие отклонения; общий вид аккуратный;

Оценка «3» - работа выполнена в заданное время, самостоятельно, с нарушением технологической последовательности, отдельные операции выполнены с отклонением от образца (если не было на то установки); оформлено небрежно или не закончено в срок;

Оценка «2» - ученик самостоятельно не справился с работой, технологическая последовательность нарушена, при выполнении операций допущены большие отклонения, оформлено небрежно и имеет незавершенный вид.

Критерии оценок для теста:

Оценка «5» - 86% и выше

Оценка «4» - 71% - 85%

Оценка «3» - 50% - 70%

Оценка «2» - 49% и ниже

Критерии оценок для творческого проекта:

- эстетичность оформления,
- содержание, соответствующее теме работы,
- полная и достоверная информация по теме,
- отражение всех знаний и умений учащихся в данной программе,
- актуальность выбранной темы в учебно-воспитательном процессе.

В настоящей программе учтено, что сегодня, в соответствии с Федеральным государственным стандартом начального образования, учащиеся к концу начальной школы должны обладать ИКТ-компетентностью, достаточной для дальнейшего обучения. Курс информатики в 7-9 классах, опирается на опыт постоянного применения ИКТ, уже имеющийся у учащихся, дает теоретическое осмысление, интерпретацию и обобщение этого опыта.

При организации занятий школьников по информатике и информационным технологиям необходимо использовать различные методы и средства обучения с тем, чтобы с одной стороны, свести работу за ПК к регламентированной норме; с другой стороны, достичь наибольшего педагогического эффекта.

На уроках параллельно применяются общие и специфические **методы**, связанные с применением средств ИКТ:

- словесные методы обучения (рассказ, объяснение, беседа, работа с учебником, рабочей тетрадью);
- наглядные методы (наблюдение, иллюстрация, демонстрация наглядных пособий, презентаций);
- практические методы (устные и письменные упражнения, практические работы за ПК);
- проблемное обучение;
- метод проектов;
- ролевой метод.

Форма организации образовательного процесса: классно-урочная система, дистанционная.

Применение на уроках интерактивных форм работы обучающимися является ведущей формой организации учебной деятельности учащихся. На уроках в соответствии с Программой формирования/развития УУД и РПВ используются следующие формы совместной деятельности учащихся: интеллектуальных игр, стимулирующих познавательную мотивацию учащихся; дидактического театра, где полученные на уроке знания обыгрываются в театральных постановках; дискуссий, которые дают учащимся возможность приобрести опыт ведения конструктивного диалога; групповой работы или работы в парах, которые учат учащихся командной работе и взаимодействию с другими учащимися.

Включение в урок игровых процедур, которые помогают поддержать мотивацию детей к получению знаний, налаживанию позитивных межличностных отношений в классе, помогают установлению доброжелательной атмосферы во время урока: «Зигзаг», «Учебное задание-проект», «Занимательная информатика», «Ключевые термины» и др.

Применение на уроках организации приемов шевства – заданий на помощь и взаимовыручку.

Формы уроков:

- уроки «открытия» нового знания;
- уроки рефлексии;

- уроки общеметодологической направленности;
- уроки развивающего контроля.

Технологии, используемые в обучении:

- развивающего обучения,
- обучения в сотрудничестве,
- проблемного обучения,
- развития исследовательских навыков,
- информационно-коммуникационные,
- здоровьесбережения и т. д.

На изучение предмета Информатика на уровне основного общего образования отводится 1 учебный час в неделю в 7-9 классах, всего 105 часов.

В соответствии с требованиями Федерального государственного образовательного стандарта основного общего образования предмет «Информатика» изучается с 7 класса по 9 класс и входит в предметную область «Естественно - научные предметы».

2. Планируемые результаты освоения учебного предмета

Изучение информатики на уровне основного общего образования направлено на достижение обучающимися личностных, метапредметных и предметных результатов освоения содержания учебного предмета.

Личностные результаты имеют направленность на решение задач воспитания, развития и социализации обучающихся средствами учебного предмета.

В результате изучения информатики на уровне основного общего образования у обучающегося будут сформированы следующие личностные результаты в части:

1) патриотического воспитания: ценностное отношение к отечественному культурному, историческому и научному наследию, понимание значения информатики как науки в жизни современного общества, владение достоверной информацией о передовых мировых и отечественных достижениях в области информатики и информационных технологий, заинтересованность в научных знаниях о цифровой трансформации современного общества;

2) духовно-нравственного воспитания: ориентация на моральные ценности и нормы в ситуациях нравственного выбора, готовность оценивать свое поведение и поступки, а также поведение и поступки других людей с позиции нравственных и правовых норм с учетом осознания последствий поступков, активное неприятие асоциальных поступков, в том числе в Интернете;

3) гражданского воспитания: представление о социальных нормах и правилах межличностных отношений в коллективе, в том числе в социальных сообществах, соблюдение правил безопасности, в том числе навыков безопасного поведения в Интернет-среде, готовность к разнообразной совместной деятельности при выполнении учебных, познавательных задач, создании учебных проектов, стремление к взаимопониманию и взаимопомощи в процессе этой учебной деятельности, готовность оценивать свое поведение и поступки своих товарищей с позиции нравственных и правовых норм с учетом осознания последствий поступков;

4) ценностей научного познания: сформированность мировоззренческих представлений об информации, информационных процессах и информационных технологиях, соответствующих современному уровню развития науки и общественной практики и составляющих базовую основу для понимания сущности научной картины мира; интерес к обучению и познанию, любознательность, готовность и способность к самообразованию, осознанному выбору направленности и уровня обучения в дальнейшем; овладение основными навыками исследовательской деятельности, установка на осмысление опыта, наблюдений, поступков и стремление совершенствовать пути достижения индивидуального и коллективного благополучия; сформированность информационной культуры, в том числе навыков самостоятельной работы с учебными текстами, справочной литературой, разнообразными средствами информационных технологий, а также умения самостоятельно определять цели своего обучения, ставить и формулировать для себя новые задачи в учебе и познавательной деятельности, развивать мотивы и интересы своей познавательной деятельности;

5) формирования культуры здоровья: осознание ценности жизни, ответственное отношение к своему здоровью, установка на здоровый образ жизни, в том числе и за счет освоения и соблюдения требований безопасной эксплуатации средств информационных и коммуникационных технологий;

6) трудового воспитания: интерес к практическому изучению профессий и труда в сферах профессиональной деятельности, связанных с информатикой, программированием и информационными технологиями, основанными на достижениях науки информатики и научно-технического прогресса; осознанный выбор и построение индивидуальной траектории образования и жизненных планов с учетом личных и общественных интересов и потребностей;

7) экологического воспитания: осознание глобального характера экологических проблем и путей их решения, в том числе с учетом возможностей информационных и коммуникационных технологий;

8) адаптации обучающегося к изменяющимся условиям социальной и природной среды: освоение обучающимися социального опыта, основных социальных ролей, соответствующих ведущей деятельности возраста, норм и правил общественного поведения, форм социальной жизни в группах и сообществах, в том числе существующих в виртуальном пространстве.

Метапредметные результаты освоения программы по информатике отражают овладение универсальными учебными действиями - познавательными, коммуникативными, регулятивными.

Овладение универсальными учебными познавательными действиями:

1) базовые логические действия: умение определять понятия, создавать обобщения, устанавливать аналогии, классифицировать, самостоятельно выбирать основания и критерии для классификации, устанавливать причинно-следственные связи, строить логические рассуждения, проводить умозаключения (индуктивные, дедуктивные и по аналогии) и выводы; умение создавать, применять и преобразовывать знаки и символы, модели и схемы для решения учебных и познавательных задач; самостоятельно выбирать способ решения учебной задачи (сравнивать несколько вариантов решения, выбирать наиболее подходящий с учетом самостоятельно выделенных критериев).

2) базовые исследовательские действия: формулировать вопросы, фиксирующие разрыв между реальным и желательным состоянием ситуации, объекта, и самостоятельно устанавливать искомое и данное; оценивать на применимость и достоверность информацию, полученную в ходе исследования; прогнозировать возможное дальнейшее развитие процессов, событий и их последствия в аналогичных или сходных ситуациях, а также выдвигать предположения об их развитии в новых условиях и контекстах.

3) работа с информацией: выявлять дефицит информации, данных, необходимых для решения поставленной задачи; применять различные методы, инструменты и запросы при поиске и отборе информации или данных из источников с учетом предложенной учебной задачи и заданных критериев; выбирать, анализировать, систематизировать и интерпретировать информацию различных видов и форм представления; самостоятельно выбирать оптимальную форму представления информации и иллюстрировать решаемые задачи несложными схемами, диаграммами, иной графикой и их комбинациями; оценивать надежность информации по критериям, предложенным учителем или сформулированным самостоятельно; эффективно запоминать и систематизировать информацию.

Овладение универсальными учебными коммуникативными действиями:

1) общение: сопоставлять свои суждения с суждениями других участников диалога, обнаруживать различие и сходство позиций; публично представлять результаты выполнен-

ного опыта (эксперимента, исследования, проекта);самостоятельно выбирать формат выступления с учетом задач презентации и особенностей аудитории и в соответствии с ним составлять устные и письменные тексты с использованием иллюстративных материалов.

2) совместная деятельность (сотрудничество):понимать и использовать преимущества командной и индивидуальной работы при решении конкретной проблемы, в том числе при создании информационного продукта;принимать цель совместной информационной деятельности по сбору, обработке, передаче, формализации информации, коллективно строить действия по ее достижению: распределять роли, договариваться, обсуждать процесс и результат совместной работы;выполнять свою часть работы с информацией или информационным продуктом, достигая качественного результата по своему направлению и координируя свои действия с другими членами команды;оценивать качество своего вклада в общий информационный продукт по критериям, самостоятельно сформулированным участниками взаимодействия;сравнивать результаты с исходной задачей и вклад каждого члена команды в достижение результатов, разделять сферу ответственности и проявлять готовность к предоставлению отчета перед группой.

Овладение универсальными учебными регулятивными действиями:

1) самоорганизация:выявлять в жизненных и учебных ситуациях проблемы, требующие решения; ориентироваться в различных подходах к принятию решений (индивидуальное принятие решений, принятие решений в группе);самостоятельно составлять алгоритм решения задачи (или его часть), выбирать способ решения учебной задачи с учетом имеющихся ресурсов и собственных возможностей, аргументировать предлагаемые варианты решений;составлять план действий (план реализации намеченного алгоритма решения), корректировать предложенный алгоритм с учетом получения новых знаний об изучаемом объекте;проводить выбор в условиях противоречивой информации и брать ответственность за решение.

2) самоконтроль (рефлексия):владеть способами самоконтроля, самомотивации и рефлексии;давать оценку ситуации и предлагать план ее изменения;учитывать контекст и предвидеть трудности, которые могут возникнуть при решении учебной задачи, адаптировать решение к меняющимся обстоятельствам;объяснять причины достижения (недостижения) результатов информационной деятельности, давать оценку приобретенному опыту, уметь находить позитивное в произошедшей ситуации;вносить коррективы в деятельность на основе новых обстоятельств, изменившихся ситуаций, установленных ошибок, возникших трудностей;оценивать соответствие результата цели и условиям.

3) эмоциональный интеллект:ставить себя на место другого человека, понимать мотивы и намерения другого.

4) принятие себя и других: осознавать невозможность контролировать все вокруг даже в условиях открытого доступа к любым объемам информации.

Предметные результаты освоения программы по информатике на уровне основного общего образования.

К концу обучения в 7 классе у обучающегося будут сформированы умения:

- ✓ пояснять на примерах смысл понятий "информация", "информационный процесс", "обработка информации", "хранение информации", "передача информации";
- ✓ кодировать и декодировать сообщения по заданным правилам, демонстрировать понимание основных принципов кодирования информации различной природы (текстовой, графической, аудио);
- ✓ сравнивать длины сообщений, записанных в различных алфавитах, оперировать единицами измерения информационного объема и скорости передачи данных;
- ✓ оценивать и сравнивать размеры текстовых, графических, звуковых файлов и видеофайлов;
- ✓ приводить примеры современных устройств хранения и передачи информации, сравнивать их количественные характеристики;
- ✓ выделять основные этапы в истории и понимать тенденции развития компьютеров и программного обеспечения;
- ✓ получать и использовать информацию о характеристиках персонального компьютера и его основных элементах (процессор, оперативная память, долговременная память, устройства ввода-вывода);
- ✓ соотносить характеристики компьютера с задачами, решаемыми с его помощью;
- ✓ ориентироваться в иерархической структуре файловой системы (записывать полное имя файла (каталога), путь к файлу (каталогу) по имеющемуся описанию файловой структуры некоторого информационного носителя);
- ✓ работать с файловой системой персонального компьютера с использованием графического интерфейса, а именно: создавать, копировать, перемещать, переименовывать, удалять и архивировать файлы и каталоги, использовать антивирусную программу;
- ✓ представлять результаты своей деятельности в виде структурированных иллюстрированных документов, мультимедийных презентаций;
- ✓ искать информацию в Интернете (в том числе по ключевым словам, по изображению), критически относиться к найденной информации, осознавая опасность для личности и общества распространения вредоносной информации, в том числе экстремистского и террористического характера; понимать структуру адресов веб-ресурсов; использовать современные сервисы интернет-коммуникаций; соблюдать требования безопас-

ной эксплуатации технических средств информационных и коммуникационных технологий, соблюдать сетевой этикет, базовые нормы информационной этики и права при работе с приложениями на любых устройствах и в Интернете, выбирать безопасные стратегии поведения в сети;

- ✓ применять методы профилактики негативного влияния средств информационных и коммуникационных технологий на здоровье пользователя.

К концу обучения **в 8 классе** у обучающегося будут сформированы умения:

- ✓ пояснять на примерах различия между позиционными и непозиционными системами счисления;
- ✓ записывать и сравнивать целые числа от 0 до 1024 в различных позиционных системах счисления (с основаниями 2, 8, 16), выполнять арифметические операции над ними;
- ✓ раскрывать смысл понятий "высказывание", "логическая операция", "логическое выражение";
- ✓ записывать логические выражения с использованием дизъюнкции, конъюнкции и отрицания, определять истинность логических выражений, если известны значения истинности входящих в него переменных, строить таблицы истинности для логических выражений;
- ✓ раскрывать смысл понятий "исполнитель", "алгоритм", "программа", понимая разницу между употреблением этих терминов в обыденной речи и в информатике;
- ✓ описывать алгоритм решения задачи различными способами, в том числе в виде блок-схемы;
- ✓ составлять, выполнять вручную и на компьютере несложные алгоритмы с использованием ветвлений и циклов для управления исполнителями, такими, как "Робот", "Черепашка", "Чертежник";
- ✓ использовать константы и переменные различных типов (числовых, логических, символьных), а также содержащие их выражения, использовать оператор присваивания;
- ✓ использовать при разработке программ логические значения, операции и выражения с ними;
- ✓ анализировать предложенные алгоритмы, в том числе определять, какие результаты возможны при заданном множестве исходных значений;
- ✓ создавать и отлаживать программы на одном из языков программирования (Python, C++, Паскаль, Java, C#, Школьный Алгоритмический Язык), реализующие несложные алгоритмы обработки числовых данных с использованием циклов и ветвлений, в том

числе реализующие проверку делимости одного целого числа на другое, проверку натурального числа на простоту, выделения цифр из натурального числа.

К концу обучения в **9 классе** у обучающегося будут сформированы умения:

- ✓ разбивать задачи на подзадачи, составлять, выполнять вручную и на компьютере несложные алгоритмы с использованием ветвлений, циклов и вспомогательных алгоритмов для управления исполнителями, такими как Робот, Черепашка, Чертежник;
- ✓ составлять и отлаживать программы, реализующие типовые алгоритмы обработки числовых последовательностей или одномерных числовых массивов (поиск максимумов, минимумов, суммы или количества элементов с заданными свойствами) на одном из языков программирования (Python, C++, Паскаль, Java, C#, Школьный Алгоритмический Язык);
- ✓ раскрывать смысл понятий "модель", "моделирование", определять виды моделей, оценивать соответствие модели моделируемому объекту и целям моделирования;
- ✓ использовать графы и деревья для моделирования систем сетевой и иерархической структуры, находить кратчайший путь в графе;
- ✓ выбирать способ представления данных в соответствии с поставленной задачей (таблицы, схемы, графики, диаграммы) с использованием соответствующих программных средств обработки данных;
- ✓ использовать электронные таблицы для обработки, анализа и визуализации числовых данных, в том числе с выделением диапазона таблицы и упорядочиванием (сортировкой) его элементов;
- ✓ создавать и применять в электронных таблицах формулы для расчетов с использованием встроенных арифметических функций (суммирование и подсчет значений, отвечающих заданному условию, среднее арифметическое, поиск максимального и минимального значения), абсолютной, относительной, смешанной адресации;
- ✓ использовать электронные таблицы для численного моделирования в простых задачах из разных предметных областей;
- ✓ использовать современные интернет-сервисы (в том числе коммуникационные сервисы, облачные хранилища данных, онлайн-программы (текстовые и графические редакторы, среды разработки)) в учебной и повседневной деятельности;
- ✓ приводить примеры использования геоинформационных сервисов, сервисов государственных услуг, образовательных сервисов Интернета в учебной и повседневной деятельности;
- ✓ использовать различные средства защиты от вредоносного программного обеспечения, защищать персональную информацию от несанкционированного доступа и его

последствий (разглашения, подмены, утраты данных) с учетом основных технологических и социально-психологических аспектов использования сети Интернет (сетевая анонимность, цифровой след, аутентичность субъектов и ресурсов, опасность вредоносного кода);

- ✓ распознавать попытки и предупреждать вовлечение себя и окружающих в деструктивные и криминальные формы сетевой активности (в том числе кибербуллинг, фишинг).

3. Содержание учебного предмета

При реализации программы учебного предмета «Информатика» у учащихся формируется информационная алгоритмическая культура; умения формализации и структурирования информации, способ представления данных в соответствии с поставленной задачей – таблицы, схемы, графики, диаграммы, с использованием соответствующих программных средств обработки данных; представления о компьютере как универсальном устройстве обработки информации; представления об основных изучаемых понятиях: информация, алгоритм, модель и их свойствах; развивается алгоритмическое мышление, необходимое для профессиональной деятельности в современном обществе; формируются представления о том, как понятия и конструкции информатики применяются в реальном мире, о роли информационных технологий и роботизированных устройств в жизни людей, промышленности и научных исследованиях; навыков и умений безопасного и целесообразного поведения при работе с компьютерными программами и в сети Интернет, умения соблюдать нормы информационной этики и права.

7 класс:

Теоретические основы информатики.

Информация и информационные процессы.

Информация - одно из основных понятий современной науки.

Информация как сведения, предназначенные для восприятия человеком, и информация как данные, которые могут быть обработаны автоматизированной системой.

Дискретность данных. Возможность описания непрерывных объектов и процессов с помощью дискретных данных.

Информационные процессы - процессы, связанные с хранением, преобразованием и передачей данных.

Представление информации

Символ. Алфавит. Мощность алфавита. Разнообразие языков и алфавитов. Естественные и формальные языки. Алфавит текстов на русском языке. Двоичный алфавит. Количество всевозможных слов (кодových комбинаций) фиксированной длины в двоичном алфавите. Преобразование любого алфавита к двоичному. Количество различных слов фиксированной длины в алфавите определенной мощности.

Кодирование символов одного алфавита с помощью кодовых слов в другом алфавите, кодовая таблица, декодирование.

Двоичный код. Представление данных в компьютере как текстов в двоичном алфавите.

Информационный объем данных. Бит - минимальная единица количества информации - двоичный разряд. Единицы измерения информационного объема данных. Бит, байт, килобайт, мегабайт, гигабайт.

Скорость передачи данных. Единицы скорости передачи данных.

Кодирование текстов. Равномерный код. Неравномерный код. Кодировка ASCII. Восьмибитные кодировки. Понятие о кодировках UNICODE. Декодирование сообщений с использованием равномерного и неравномерного кода. Информационный объем текста.

Искажение информации при передаче.

Общее представление о цифровом представлении аудиовизуальных и других непрерывных данных.

Кодирование цвета. Цветовые модели. Модель RGB. Глубина кодирования. Палитра.

Растровое и векторное представление изображений. Пиксель. Оценка информационного объема графических данных для растрового изображения.

Кодирование звука. Разрядность и частота записи. Количество каналов записи.

Оценка количественных параметров, связанных с представлением и хранением звуковых файлов.

Цифровая грамотность.

Компьютер - универсальное устройство обработки данных.

Компьютер - универсальное вычислительное устройство, работающее по программе. Типы компьютеров: персональные компьютеры, встроенные компьютеры, суперкомпьютеры. Мобильные устройства.

Основные компоненты компьютера и их назначение. Процессор. Оперативная и долговременная память. Устройства ввода и вывода. Сенсорный ввод, датчики мобильных устройств, средства биометрической аутентификации.

История развития компьютеров и программного обеспечения. Поколения компьютеров. Современные тенденции развития компьютеров. Суперкомпьютеры.

Параллельные вычисления.

Персональный компьютер. Процессор и его характеристики (такты частота, разрядность). Оперативная память. Долговременная память. Устройства ввода и вывода. Объем хранимых данных (оперативная память компьютера, жесткий и твердотельный диск, постоянная память смартфона) и скорость доступа для различных видов носителей.

Техника безопасности и правила работы на компьютере.

Программы и данные.

Программное обеспечение компьютера. Прикладное программное обеспечение. Системное программное обеспечение. Системы программирования. Правовая охрана программ и данных. Бесплатные и условно-бесплатные программы. Свободное программное обеспечение.

Файлы и папки (каталоги). Принципы построения файловых систем. Полное имя файла (папки). Путь к файлу (папке). Работа с файлами и каталогами средствами операционной системы: создание, копирование, перемещение, переименование и удаление файлов и папок (каталогов). Типы файлов. Свойства файлов. Характерные размеры файлов различных типов (страница текста, электронная книга, фотография, запись песни, видеоклип, полнометражный фильм). Архивация данных. Использование программ-архиваторов. Файловый менеджер. Поиск файлов средствами операционной системы.

Компьютерные вирусы и другие вредоносные программы. Программы для защиты от вирусов.

Компьютерные сети.

Объединение компьютеров в сеть. Сеть Интернет. Веб-страница, веб-сайт. Структура адресов веб-ресурсов. Браузер. Поисковые системы. Поиск информации по ключевым словам и по изображению. Достоверность информации, полученной из Интернета.

Современные сервисы интернет-коммуникаций.

Сетевой этикет, базовые нормы информационной этики и права при работе в Интернете. Стратегии безопасного поведения в Интернете.

Информационные технологии.

Текстовые документы.

Текстовые документы и их структурные элементы (страница, абзац, строка, слово, символ).

Единицы измерения длины двоичных текстов: бит, байт, Килобайт и т. д. Количество информации, содержащееся в сообщении.

Текстовый процессор - инструмент создания, редактирования и форматирования текстов. Правила набора текста. Редактирование текста. Свойства символов. Шрифт. Типы шрифтов (рубленые, с засечками, моноширинные). Полужирное и курсивное начертание. Свойства абзацев: границы, абзацный отступ, интервал, выравнивание. Параметры страницы. Стилизовое форматирование.

Структурирование информации с помощью списков и таблиц. Многоуровневые списки. Добавление таблиц в текстовые документы.

Вставка изображений в текстовые документы. Обтекание изображений текстом. Включение в текстовый документ диаграмм, формул, нумерации страниц, колонтитулов, ссылок и других элементов.

Проверка правописания. Расстановка переносов. Голосовой ввод текста. Оптическое распознавание текста. Компьютерный перевод. Использование сервисов Интернета для обработки текста.

Компьютерная графика.

Знакомство с графическими редакторами. Растровые рисунки. Использование графических примитивов.

Операции редактирования графических объектов, в том числе цифровых фотографий: изменение размера, обрезка, поворот, отражение, работа с областями (выделение, копирование, заливка цветом), коррекция цвета, яркости и контрастности.

Векторная графика. Создание векторных рисунков встроенными средствами текстового процессора или других программ (приложений). Добавление векторных рисунков в документы.

Ввод изображений с использованием различных цифровых устройств (цифровых фотоаппаратов, микроскопов, видеокамер, сканеров и т. д.).

Мультимедийные презентации.

Подготовка мультимедийных презентаций. Слайд. Добавление на слайд текста и изображений. Работа с несколькими слайдами.

Добавление на слайд аудиовизуальных данных. Анимация. Гиперссылки.

8 класс:

Теоретические основы информатики.

Системы счисления.

Непозиционные и позиционные системы счисления. Алфавит. Основание. Развернутая форма записи числа. Перевод в десятичную систему чисел, записанных в других системах счисления.

Римская система счисления.

Двоичная система счисления. Перевод целых чисел в пределах от 0 до 1024 в двоичную систему счисления. Восьмеричная система счисления. Перевод чисел из восьмеричной системы в двоичную и десятичную системы и обратно. Шестнадцатеричная система счисления. Перевод чисел из шестнадцатеричной системы в двоичную, восьмеричную и десятичную системы и обратно.

Арифметические операции в двоичной системе счисления.

Элементы математической логики.

Логические высказывания. Логические значения высказываний. Элементарные и составные высказывания. Логические операции: "и" (конъюнкция, логическое умножение), "или" (дизъюнкция, логическое сложение), "не" (логическое отрицание). Приоритет логических операций. Определение истинности составного высказывания, если известны значения истинности входящих в него элементарных высказываний. Логические выражения. Правила записи логических выражений. Построение таблиц истинности логических выражений.

Логические элементы. Знакомство с логическими основами компьютера.

Алгоритмы и программирование.

Исполнители и алгоритмы. Алгоритмические конструкции.

Понятие алгоритма. Исполнители алгоритмов. Алгоритм как план управления исполнителем.

Свойства алгоритма. Способы записи алгоритма (словесный, в виде блок-схемы, программа).

Алгоритмические конструкции. Конструкция "следование". Линейный алгоритм. Ограниченность линейных алгоритмов: невозможность предусмотреть зависимость последовательности выполняемых действий от исходных данных.

Конструкция "ветвление": полная и неполная формы. Выполнение и невыполнение условия (истинность и ложность высказывания). Простые и составные условия.

Конструкция "повторения": циклы с заданным числом повторений, с условием выполнения, с переменной цикла.

Разработка для формального исполнителя алгоритма, приводящего к требуемому результату при конкретных исходных данных. Разработка несложных алгоритмов с использованием циклов и ветвлений для управления формальными исполнителями, такими

как Робот, Черепашка, Чертежник. Выполнение алгоритмов вручную и на компьютере. Синтаксические и логические ошибки. Отказы.

Язык программирования.

Язык программирования (Паскаль, Школьный Алгоритмический Язык).

Система программирования: редактор текста программ, транслятор, отладчик.

Переменная: тип, имя, значение. Целые, вещественные и символьные переменные.

Оператор присваивания. Арифметические выражения и порядок их вычисления.

Операции с целыми числами: целочисленное деление, остаток от деления.

Ветвления. Составные условия (запись логических выражений на изучаемом языке программирования). Нахождение минимума и максимума из двух, трех и четырех чисел.

Решение квадратного уравнения, имеющего вещественные корни.

Диалоговая отладка программ: пошаговое выполнение, просмотр значений величин, отладочный вывод, выбор точки останова.

Цикл с условием. Алгоритм Евклида для нахождения наибольшего общего делителя двух натуральных чисел. Разбиение записи натурального числа в позиционной системе с основанием, меньшим или равным 10, на отдельные цифры.

Цикл с переменной. Алгоритмы проверки делимости одного целого числа на другое, проверки натурального числа на простоту.

Обработка символьных данных. Символьные (строковые) переменные.

Посимвольная обработка строк. Подсчет частоты появления символа в строке.

Встроенные функции для обработки строк.

Анализ алгоритмов.

Определение возможных результатов работы алгоритма при данном множестве входных данных, определение возможных входных данных, приводящих к данному результату.

9 класс:

Теоретические основы информатики.

Моделирование как метод познания.

Модель. Задачи, решаемые с помощью моделирования. Классификации моделей. Материальные (натурные) и информационные модели. Непрерывные и дискретные модели. Имитационные модели. Игровые модели. Оценка соответствия модели моделируемому объекту и целям моделирования.

Табличные модели. Таблица как представление отношения.

Базы данных. Отбор в таблице строк, удовлетворяющих заданному условию.

Граф. Вершина, ребро, путь. Ориентированные и неориентированные графы. Длина (вес) ребра. Весовая матрица графа. Длина пути между вершинами графа. Поиск оптимального пути в графе. Начальная вершина (источник) и конечная вершина (сток) в ориентированном графе. Вычисление количества путей в направленном ациклическом графе.

Дерево. Корень, вершина (узел), лист, ребро (дуга) дерева. Высота дерева. Поддерево. Примеры использования деревьев. Перебор вариантов с помощью дерева.

Понятие математической модели. Задачи, решаемые с помощью математического (компьютерного) моделирования. Отличие математической модели от натурной модели и от словесного (литературного) описания объекта.

Этапы компьютерного моделирования: постановка задачи, построение математической модели, программная реализация, тестирование, проведение компьютерного эксперимента, анализ его результатов, уточнение модели.

Алгоритмы и программирование.

Разработка алгоритмов и программ.

Разбиение задачи на подзадачи. Составление алгоритмов и программ с использованием ветвлений, циклов и вспомогательных алгоритмов для управления исполнителем Робот или другими исполнителями, такими как Черепашка, Чертежник и другими.

Табличные величины (массивы). Одномерные массивы. Составление и отладка программ, реализующих типовые алгоритмы обработки одномерных числовых массивов, на одном из языков программирования (Python, C++, Паскаль, Java, C#, Школьный Алгоритмический Язык): заполнение числового массива случайными числами, в соответствии с формулой или путем ввода чисел, нахождение суммы элементов массива, линейный поиск заданного значения в массиве, подсчет элементов массива, удовлетворяющих заданному условию, нахождение минимального (максимального) элемента массива. Сортировка массива.

Обработка потока данных: вычисление количества, суммы, среднего арифметического, минимального и максимального значения элементов последовательности, удовлетворяющих заданному условию.

Управление.

Управление. Сигнал. Обратная связь. Получение сигналов от цифровых датчиков (касания, расстояния, света, звука и другого). Примеры использования принципа обратной связи в системах управления техническими устройствами с помощью датчиков, в том числе в робототехнике.

Примеры роботизированных систем (система управления движением в транспортной системе, сварочная линия автозавода, автоматизированное управление отопления дома, автономная система управления транспортным средством и другие системы).

Информационные технологии.

Электронные таблицы.

Понятие об электронных таблицах. Типы данных в ячейках электронной таблицы. Редактирование и форматирование таблиц. Встроенные функции для поиска максимума, минимума, суммы и среднего арифметического. Сортировка данных в выделенном диапазоне. Построение диаграмм (гистограмма, круговая диаграмма, точечная диаграмма). Выбор типа диаграммы.

Преобразование формул при копировании. Относительная, абсолютная и смешанная адресация.

Условные вычисления в электронных таблицах. Суммирование и подсчет значений, отвечающих заданному условию. Обработка больших наборов данных. Численное моделирование в электронных таблицах.

Информационные технологии в современном обществе.

Роль информационных технологий в развитии экономики мира, страны, региона. Открытые образовательные ресурсы.

Профессии, связанные с информатикой и информационными технологиями: веб-дизайнер, программист, разработчик мобильных приложений, тестировщик, архитектор программного обеспечения, специалист по анализу данных, системный администратор.

Цифровая грамотность.

Глобальная сеть Интернет и стратегии безопасного поведения в ней.

Глобальная сеть Интернет. IP-адреса узлов. Сетевое хранение данных. Методы индивидуального и коллективного размещения новой информации в Интернете. Большие данные (интернет-данные, в частности данные социальных сетей).

Понятие об информационной безопасности. Угрозы информационной безопасности при работе в глобальной сети и методы противодействия им. Правила безопасной аутентификации. Защита личной информации в Интернете. Безопасные стратегии поведения в Интернете. Предупреждение вовлечения в деструктивные и криминальные формы сетевой активности (кибербуллинг, фишинг и другие формы).

Работа в информационном пространстве.

Виды деятельности в Интернете, интернет-сервисы: коммуникационные сервисы (почтовая служба, видео-конференц-связь и другие), справочные службы (карты, распи-

сания и другие), поисковые службы, службы обновления программного обеспечения и другие службы. Сервисы государственных услуг. Облачные хранилища данных. Средства совместной разработки документов (онлайн-офисы). Программное обеспечение как веб-сервис: онлайн-овые текстовые и графические редакторы, среды разработки программ.

4. Тематическое планирование учебного предмета Информатика

Предметное содержание темы уроков	Кол – во часов на изучение каждой темы	Этнокультурная составляющая (количество часов с указанием темы ЭКС)	Практическая часть	Основные виды учебной деятельности	Организация обсуждения учащимися ценностных аспектов изучаемых явлений, организация работы с социально-значимой информацией	Тексты для чтения	Кейсы для организации проектной и исследовательской деятельности
7 класс (34 часа – 1 час в неделю).							
1 раздел «Теоретические основы информатики» - 9 часов							
Тема №1 «Цели изучения курса информатики и ИКТ. Техника безопасности и организация рабочего места»	1 час			Анализирует компьютер, с точки зрения, устройства, обрабатываемого информацию			
Тема №2 «Информация и ее свойства»	1 час			Аналитическая деятельность: • оценивает информацию с позиции ее свойств (актуальность, достоверность, полнота и пр.); • приводит примеры кодирования с использованием различных алфавитов, встречаю-	Что для Вас информация?		Кейс № 1 «Источники информации».
Тема №3 «Информационные процессы. Обработка информации»	1 час		Решение задач по теме «Информационные процессы. Обработка информации»				
Тема №4 «Информационные	1 час		Решение задач по те-		Что интересней: найти информацию		

процессы. Хранение и передача информации»			ме «Информационные процессы. Хранение и передача информации»	щихся в жизни; • классифицирует информационные процессы по принятому основанию; • выделяет информационную составляющую процессов в биологических, технических и социальных системах;	или поделиться ею с другими?		
Тема №5 «Всемирная паутина»	1 час		Решение задач по теме «Всемирная паутина»	• анализирует отношения в живой природе, технических и социальных (школа, семья и пр.) системах с позиций управления. Практическая деятельность:			
Тема №6 «Дискретная форма представления информации»	1 час		Решение задач по теме «Дискретная форма представления информации»	• кодирует и декодирует сообщения по известным правилам кодирования; • определяет количество различных символов, которые могут быть закодированы с помощью двоичного		Представление информации	
Тема №7 «Кодирование цвета. Кодирование звука»	1 час		Практическая работа № 1 по теме «Оценка информационного объёма графических данных для растрового изображения»				

			ния»				
Тема №8 «Измерение информации»	1 час		Решение задач по теме «Измерение информации»	кода фиксированной длины (разрядности); • определяет разрядность двоичного кода, необходимого для кодирования всех символов алфавита заданной мощности; • оперирует с единицами измерения количества информации (бит, байт, килобайт, мегабайт, гигабайт); оценивает числовые параметры информационных процессов (объем памяти, необходимой для хранения информации; скорость передачи информации, пропускную способность выбранного канала и пр.)	Как вы думаете, зачем нам нужно знать, как измеряется информация?		
Тема №9 Контрольная работа № 1 по теме «Теоретические основы информатики».	1 час		Контрольная работа № 1 по теме «Теоретические основы информатики»..				
2 раздел «Цифровая грамотность» -7 часов							
Тема №10 «Компьютер –	1 час			<i>Аналитическая деятельность:</i>		Первый расчетны е устройства	

универсальное вычислительное устройство, работающее по программе. Техника безопасности и правила работы на компьютере»				<ul style="list-style-type: none"> • анализирует компьютер с точки зрения единства программных и аппаратных средств; • анализирует устройства компьютера с точки зрения 			
Тема №11 «История и современные тенденции развития компьютеров»	1 час			организации процедур ввода, хранения, обработки, вывода и передачи информации;			
Тема №12 «Программное обеспечение компьютера. Правовая охрана программ и данных»	1 час			<ul style="list-style-type: none"> • определяет программные и аппаратные средства, необходимые для осуществления информационных процессов при решении задач; 	Какую роль играет программное обеспечение при работе компьютера?	Оборудование и программное обеспечение	Кейс № 2 «Как отличить лицензионный диск от пиратского»
Тема №13 «Системы программирования и прикладное программное обеспечение»	1 час			<ul style="list-style-type: none"> • анализирует информацию (сигналы о готовности и неполадке) при включении компьютера; 			
Тема №15 «Архивация данных. Использование программ-архиваторов Компьютерные	1 час			<ul style="list-style-type: none"> • определяет основные характеристики операционной системы; 			

вирусы и анти-вирусные программы»				<ul style="list-style-type: none"> • планирует собственное информационное пространство. 			
Тема №15 «Файлы и папки. Основные операции с файлами и папками»	1 час		Решение задач по теме «Файлы и файловые структуры»	<ul style="list-style-type: none"> • получает информацию о характеристиках компьютера; <p>Практическая деятельность:</p> <ul style="list-style-type: none"> • оценивает числовые параметры информационных процессов (объем памяти, необходимой для хранения информации; скорость передачи информации, пропускную способность выбранного канала и пр.); • выполняет основные операции с файлами и папками; • оперирует компьютерными информационными объектами в наглядно-графической форме; 	Какие действия можно выполнять над папками, но нельзя над файлами (или наоборот)?		
Тема № 16 «Контрольная работа № 2 по теме «Компьютер как универсальное устройство для работы с информацией»».	1 час		Контрольная работа № 2 по теме «Компьютер - универсальное устройство обработки данных».				

				<ul style="list-style-type: none"> • оценивает размеры файлов, подготовленных с использованием различных устройств ввода информации в заданный интервал времени (клавиатура, сканер, микрофон, фотокамера, видеокамера); • использует программы-архиваторы; • осуществляет защиту информации от компьютерных вирусов с помощью антивирусных программ 			
3 раздел «Информационные технологии» - 18 часов							
Тема № 17 «Формирование изображения на экране монитора»	1 час		Решение задач по теме «Формирование изображения на экране монитора»	Аналитическая деятельность: <ul style="list-style-type: none"> • анализирует пользовательский интерфейс используемого программного средства; 			
Тема № 18 «Графический редактор. Раст-	1 час		Практическая работа № 1 по те-	<ul style="list-style-type: none"> • определяет условия и возможности применения 	Почему файлы одного формата имеют разный инфор-	Фрактальная графика	

ровые рисунки»			ме«Операции и редактирования графических объектов»	программного средства для решения типовых задач; • выявляет общее и отличия в разных программных продуктах, предназначенных для решения одного класса задач.	мационный объем?		
Тема № 19 «Векторная графика»	1 час		Практическая работа № 2 «Создание векторных графических изображений»	Практическая деятельность: • определяет код цвета в палитре RGB в графическом редакторе;			Кейс № 3 «Графический редактор. Создание рисунков в графическом редакторе»
Тема № 20 «Самостоятельная работа № 1 по теме «Компьютерная графика»».	1 час		Самостоятельная работа № 1 по теме «Компьютерная графика».	• создает и редактирует изображения с помощью инструментов растрового графического редактора; создает и редактирует изображения с помощью инструментов векторного графического редактора			

Тема № 21 «Текстовые документы и технологии их создания»	1 час			<i>Аналитическая деятельность:</i> • анализирует пользовательский интерфейс используемого программного средства; • определяет условия и возможности применения программного средства для решения типовых задач;	Нужно ли современному человеку владеть технологией создания текстовых документов и почему (современная компетенция)?		
Тема № 22 «Создание текстовых документов на компьютере»	1 час		Практическая работа № 3 «Ввод и редактирование текстовых документов в текстовом процессоре»	• выявляет общее и отличия в разных программных продуктах, предназначенных для решения одного класса задач.			Кейс № 4 «Текстовый редактор. Вставка символов, которых нет на клавиатуре»
Тема № 23 «Форматирование текста»	1 час			<i>Практическая деятельность:</i> • создает небольшие текстовые документы посредством квалифицированного клавиатурного письма с использованием базовых средств текстовых			
Тема № 24 «Стилевое форматирование»	1 час						
Тема № 25 «Визуализация информации в текстовых документах»	1 час		Практическая работа № 4 «Вставка нетекстовых объектов в				

			текстовые документы»	редакторов; • форматирует текстовые документы (установка параметров страницы документа; форматирование символов и абзацев; вставка колонтитулов и номеров страниц);			
Тема № 26 «Интеллектуальные возможности современных систем обработки текстов»	1 час			• форматирует текстовые документы (установка параметров страницы документа; форматирование символов и абзацев; вставка колонтитулов и номеров страниц);	Почему программы-переводчики успешно переводят деловые документы, но не годятся для перевода текстов художественных произведений?		
Тема № 27 «Оценка количественных параметров текстовых документов»	1 час		Решение задач по теме «Оценка количественных параметров текстовых документов»	• вставляет в документ формулы, таблицы, списки, изображения; • выполняет коллективное создание текстового документа;	Какие параметры необходимо знать для расчета информационного объема текста?		
Тема № 28 «Оформление реферата «История вычислительной техники»»	1 час		Практическая работа № 5 «Оформление реферата «История вычислительной техники»»	• создает гипертекстовые документы; • выполняет кодирование декодирование текстовой информации, используя кодовые таблицы			
Тема № 29 «Самостоятельная работа № 2 по теме «Текстовые доку-»	1 час		Самостоятельная работа № 2 по теме «Текстовые доку-»	(Юникод, КОИ-8Р, Windows 1251); использует ссылки и цитирование источников			

менты»»».			кументы».	при создании на их основе собственных информационных объектов.			
Тема № 30 «Технология мультимедиа»	1 час			Аналитическая деятельность: • анализирует пользовательский интерфейс используемого программного средства;	Что относится к стандартным средствам мультимедиа?		
Тема № 31 «Компьютерные презентации»	1 час			пользовательский интерфейс используемого программного средства;	В каких отраслях применяется программа создания презентаций?		
Тема № 32 «Добавление на слайд аудиовизуальных данных, анимации и гиперссылок»	1 час		Практическая работа № 6 «Создание мультимедийной презентации»	• определяет условия и возможности применения программного средства для решения типовых задач;			
Тема № 33 «Контрольная работа № 3 по теме «Информационные технологии»».	1 час		Контрольная работа № 3 по теме «Информационные технологии».	• выявляет общее и отличия в разных программных продуктах, предназначенных для решения одного класса задач. Практическая деятельность: • создает презентации с использованием готовых шаблонов;			

				• записывает звуковые файлы с различным качеством звучания (глубиной кодирования и частотой дискретизации).			
Тема № 34 «Промежуточная аттестация»	1 час		Контрольная работа № 4 за курс 7 класса				
8 класс (34 часа – 1 час в неделю).							
1 раздел «Теоретические основы информатики» - 13 часов							
Тема № 1 «Техника безопасности и организация рабочего места. Непозиционные и позиционные системы счисления»	1 час		Компьютерный тест по теме «Техника безопасности и организация рабочего места» в кабинете информатики»	<i>Аналитическая деятельность:</i> - выявляет различие в унарных, позиционных и непозиционных системах счисления; - выявляет общее и отличия в разных позиционных системах счисления;			
Тема № 2 «Развернутая форма записи числа»	1 час			- анализирует логическую структуру высказываний.	Зачем люди кодируют информацию?		
Тема № 3 «Двоичная система счисления. Арифметические операции в двоичной системе счисления»	1 час		Решение задач по теме «Двоичная система счисления»	<i>Практическая деятельность:</i> - переводит небольшие (от 0 до 1024) целые числа из десятичной системы			Кейс № 1 «Системы счисления»

ния»				<p>системы счисления в двоичную (восьмеричную, шестнадцатеричную) и обратно;</p> <p>- выполняет операции сложения и умножения над небольшими двоичными числами;</p> <p>- записывает вещественные числа в естественной и нормальной форме;</p> <p>- строит таблицы истинности для логических выражений;</p> <p>- вычисляет истинностное значение логического выражения.</p>			
Тема № 4 «Восьмеричная система счисления»	1 час		Решение задач по теме «Восьмеричная система счисления»				
Тема № 5 «Шестнадцатеричная система счисления»	1 час		Решение задач по теме «Шестнадцатеричная система счисления»				
Тема № 6 «Представление чисел в памяти компьютера»	1 час		Практическая работа №2 по теме «Представление целых и вещественных чисел»		Как числа, записанные в различных системах счисления, хранятся и обрабатываются в памяти компьютера?		
Тема № 7 «Обобщение и систематизация знаний по теме «Системы счисления»»	1 час		Компьютерный тест по теме «Системы счисления»				
Тема № 8 «Логические высказывания. Логические операции «и», «или»,	1 час		Практическая работа №3 по теме «Построение таблиц		С какой целью можно использовать логические операции применительно к различным видам		

«не». Определен ие истинности составного вы- сказывания»			истинности для логиче- ских выра- жений»		информации?		
Тема № 9 «Таб- лицы истинно- сти»	1 час		Решение задач по те- ме «По- строение таблиц ис- тинности для логиче- ских выра- жений»				
Тема № 10 «Свойствало- гических операц ий»	1 час		Компью- терный тест по теме «Свойства логических операций»				
Тема № 11 «Решение логиче ских задач»	1 час		Решение ло- гических задач				
Тема № 12 «Логические эле менты»	1 час		Компью- терный тест по теме «Логиче- ские эле- менты»		Почему необходимо уметь строить логи- ческие схемы?		
Тема № 13 «Контрольная работа № 1 по теме «Теорети- ческие основы	1 час		«Контроль- ная работа № 1 по теме «Теоретиче-				

информатики»»			ские основы информатики»»				
2 раздел «Алгоритмы и программирование» - 21 час							
Тема № 14 «Понятие алгоритма. Исполнители алгоритмов»	1 час		Компьютерный тест по теме «Алгоритмы и исполнители»	<i>Аналитическая деятельность:</i> - определяет по блок-схеме, для решения какой задачи предназначен		История развития понятия алгоритма	Кейс № 2 «Алгоритмы»
Тема № 15 «Свойства алгоритма. Способы записи алгоритма»	1 час		Теоретический диктант по теме «Способы записи алгоритмов»	данный алгоритм; - анализирует изменение значений величин при пошаговом выполнении алгоритма;	Где вы встречали алгоритмы, записанные табличным способом?		
Тема № 16 «Объекты алгоритмов»	1 час		Проверочная работа по теме «Объекты алгоритмов»	- определяет по выбранному методу решения задачи, какие алгоритмические конструкции могут			
Тема № 17 «Алгоритмическая конструкция «следование». Линейный алгоритм»	1 час		Практическая работа № 4 по теме «Алгоритмическая конструкция «следование»»	войти в алгоритм; - сравнивает различные алгоритмы решения одной задачи. <i>Практическая деятельность:</i> - исполняет готовые алгоритмы для			
Тема № 18 «Алгоритмическая	1 час		Практическая работа	конкретных исходных данных;			

конструкция «ветвление». Полная форма ветвления.			№ 5 по теме «Алгоритмическая конструкция «ветвление»»	- преобразовывает запись алгоритма с одной формы в другую; - строит цепочки команд, дающих нужный результат			
Тема № 19 «Алгоритмическая конструкция «ветвление». Сокращенная форма ветвления»			Практическая работа № 6 по теме «Сокращенная форма ветвления»	при конкретных исходных данных для исполнителя арифметических действий; - строит цепочки команд, дающих			
Тема № 20 «Алгоритмическая конструкция «повторение». Цикл с заданным условием продолжения работы»	1 час		Практическая работа № 7 по теме «Цикл с заданным условием продолжения работы»	нужный результат при конкретных исходных данных для исполнителя, преобразующего строки символов; - строит арифметические, строковые, логические			
Тема № 21 «Алгоритмическая конструкция «повторение». Цикл с заданным условием окончания работы»	1 час		Практическая работа № 8 по теме «Цикл с заданным условием окончания работы»	выражения и вычислять их значения.			
Тема № 22 «Алгоритмическая	1 час		Практическая работа				

конструкция «повторение». Цикл с заданным числом повторений»			№ 9 по теме «Цикл созданным с заданным числом повторений»				
Тема № 23 «Язык программирования. Система программирования»	1 час		Решение примеров по теме «Общие сведения о языке программирования Паскаль»	Аналитическая деятельность: - анализирует готовые программы; - определяет по программе, для решения какой задачи она предназначена;	В чем сходство и отличия структуры алгоритма со структурой программы на языке Паскаль?		Кейс № 3 «Основные понятия языка Паскаль»
Тема № 24 «Переменные. Оператор присваивания»	1 час		Практическая работа № 10 по теме «Организация ввода и вывода данных»	- выделяет этапы решения задачи на компьютере. Практическая деятельность: - программирует линейные алгоритмы, предполагающие вычисление арифметических, строковых и логических выражений;		Компьютерное программирование	
Тема № 25 «Программирование линейных алгоритмов»	1 час		Практическая работа № 11 по теме «Программирование линейных алгоритмов»	- разрабатывает программы, содержащие оператор/операторы ветвления (реше-			
Тема № 26 «Разработка программ, со-	1 час		Решение задач по теме ««Про-				

держущих оператор ветвления»			граммирование разветвляющихся алгоритмов»	ние линейного неравенства, решение квадратного уравнения и пр.), в том числе с использованием логических операций; - разрабатывает программы, содержащие оператор (операторы) цикла			
Тема № 27 «Диалоговая отладка программ»	1 час		Практическая работа № 12 по теме «Программирование разветвляющихся алгоритмов»				
Тема № 28 «Программирование циклов с заданным условием продолжения работы»	1 час		Практическая работа № 13 по теме «Программирование циклов с заданным условием продолжения работы»				
Тема № 29 «Программирование циклов с заданным условием окончания работы»	1 час		Практическая работа № 14 по теме «Программирование циклов с заданным				

			условием окончания работы»				
Тема № 30 «Программирование циклов с заданным числом повторений»	1 час		Практическая работа № 15 по теме «Программирование циклов с заданным числом повторений»				
Тема № 31 «Обработка символьных данных»	1 час		«Решение задач на обработку символьных данных»				
Тема № 32 «Анализ алгоритмов. Определение возможных результатов работы алгоритма при заданном множестве входных данных. Определение возможных входных данных, приводящих к данному результату».	1 час		Практическая работа № 16 «Анализ алгоритмов»				
Тема № 33 «Контрольная»	1 час		Контроль-				

работа №2 по теме «Алгоритмы и программирование».			ная работа №2 по теме «Алгоритмы и программирование»				
Тема № 34 «Промежуточная аттестация»	1 час		Контрольная работа № 3 за курс 8 класса				
9 класс (34 часа – 1 час в неделю).							
1 раздел «Теоретические основы информатики» - 9 часов							
Тема № 1 «Цели изучения курса информатики и ИКТ. Техника безопасности и организация рабочего места»	1 час			<p><i>Аналитическая деятельность:</i></p> <ul style="list-style-type: none"> • различает натурные и информационные модели, изучаемые в школе, встречающиеся в жизни; • осуществляет системный анализ объекта, выделять среди его свойств существенные свойства с точки зрения целей моделирования; 			
Тема № 2 «Модели и моделирование. Классификация моделей»	1 час				Могут ли разные объекты описываться одной и той же моделью (если да, приведите примеры)?		Кейс №1 «Виды моделей»
Тема № 3 «Математическое моделирование. Этапы компьютерного мо-	1 час						

делирования »				<ul style="list-style-type: none"> оценивает адекватность модели моделируемому объекту и целям моделирования; определяет вид информационной модели в зависимости от стоящей задачи; приводит примеры использования таблиц, диаграмм, схем, графов и т.д. при описании объектов окружающего мира. <p>Практическая деятельность:</p> <ul style="list-style-type: none"> строит и интерпретирует различные информационные модели (таблицы, диаграммы, графы, схемы, 			
Тема № 4 «Графические модели».	1 час		Практическая работа № 1 «Вычисление количества путей в направленном ациклическом графе»				
Тема № 5 «Табличные информационные модели».	1 час		Практическая работа № 2 «Табличные информационные модели»				
Тема № 6 «База данных как модель предметной области. Реляционные базы данных».	1 час		Практическая работа № 3 «Реляционные базы данных»				
Тема № 7 «Система управления базами данных».	1 час						
Тема № 8 «Разработка однотабличной базы данных. Составление запросов	1 час		Практическая работа № 4 «Создание базы данных.				

к базе данных».			Запросы на выборку данных»	<p>блок-схемы алгоритмов);</p> <ul style="list-style-type: none"> • преобразовывает объект из одной формы представления информации в другую с минимальными потерями в полноте информации; • исследует с помощью информационных моделей объекты в соответствии с поставленной задачей; • работает с готовыми компьютерным и моделями из различных предметных областей; • создает однотабличные базы данных; • осуществляет поиск записей в готовой базе 			
Тема № 9 «Контрольная работа № 1 по теме «Теоретические основы информатики»».	1 час		Контрольная работа № 1 по теме «Теоретические основы информатики»»				

				данных; <ul style="list-style-type: none"> • осуществляет сортировку записей в готовой базе данных. 			
2 раздел «Алгоритмы и программирование» - 8 часов							
Тема № 10 «Разбиение задачи на подзадачи. Составление алгоритмов и программ с использованием ветвлений, циклов и вспомогательных алгоритмов».	1 час			<i>Аналитическая деятельность:</i> <ul style="list-style-type: none"> • приводит примеры формальных и неформальных исполнителей; • придумывает задачи по управлению учебными исполнителями; • выделяет примеры ситуаций, которые могут быть описаны с помощью линейных алгоритмов, алгоритмов с ветвлениями и циклами; • определяет по блок-схеме, для 			
Тема № 11 «Одномерные массивы. Типовые алгоритмы обработки массивов».	1 час						
Тема № 12 «Вычисление суммы элементов массива».	1 час		Практическая работа № 5 «Вычисление суммы элементов массива»				

Тема № 13 «Последовательный поиск в массиве».	1 час		Практическая работа № 6 «Последовательный поиск в массиве»	решения какой задачи предназначен данный алгоритм; • анализирует изменение значений величин при пошаговом выполнении алгоритма;			Кейс № 2 «Последовательный поиск в массиве»
Тема № 14 «Сортировка массива. Обработка потока данных».	1 час		Практическая работа № 7 «Сортировка массива»	• определяет по выбранному методу решения задачи, какие алгоритмические конструкции могут войти в алгоритм;			
Тема № 15 «Управление. Сигнал. Обратная связь».	1 час			• осуществляет разбиение исходной задачи на подзадачи;			
Тема № 16 «Роботизированные системы».	1 час		Практическая работа № 8 «Запись вспомогательных алгоритмов на языке Паскаль»	• сравнивает различные алгоритмы решения одной задачи.			
Тема № 17 «Контрольная работа № 2 по теме «Алгоритмы и программирование»».	1 час		Контрольная работа № 2 по теме «Алгоритмы и программирование»	• анализирует готовые программы			

				<p>ы;</p> <ul style="list-style-type: none"> • определяет по программе, для решения какой задачи она предназначена; • выделяет этапы решения задачи на компьютере. <p>Практическая деятельность:</p> <ul style="list-style-type: none"> • исполняет готовые алгоритмы для конкретных исходных данных; • преобразовывает запись алгоритма с одной формы в другую; • строит цепочки команд, дающих нужный результат при конкретных исходных данных для исполнителя арифметических действий; 			
--	--	--	--	---	--	--	--

				<ul style="list-style-type: none">• строит цепочки команд, дающих нужный результат при конкретных исходных данных для исполнителя, преобразующего строки символов;• составляет линейные алгоритмы по управлению учебным исполнителем;• составляет алгоритмы с ветвлениями по управлению учебным исполнителем;• составляет циклические алгоритмы по управлению учебным исполнителем;• строит арифметические, строковые,			
--	--	--	--	--	--	--	--

				<p>логические выражения и вычисляет их значения;</p> <ul style="list-style-type: none">• строит алгоритм (различные алгоритмы) решения задачи с использованием основных алгоритмических конструкций и подпрограмм.• программирует линейные алгоритмы, предполагающие вычисление арифметических, строковых и логических выражений;• разрабатывает программы, содержащие оператор/операторы ветвления (решение линейного неравен-			
--	--	--	--	---	--	--	--

				<p>ства, решение квадратного уравнения и пр.), в том числе с использованием логических операций;</p> <ul style="list-style-type: none">• разрабатывает программы, содержащие оператор (операторы) цикла;• разрабатывает программы, содержащие подпрограмму;• разрабатывает программы для обработки одномерного массива: нахождение минимального (максимального) значения в данном массиве; подсчёт количества элементов массива, удовлетворяющих некоторому условию; нахождение суммы всех			
--	--	--	--	--	--	--	--

				элементов массива; нахождение количества и суммы всех четных элементов в массиве; сортировка элементов массива и пр.			
3 раздел «Информационные технологии» - 6 часов							
Тема № 18 «Электронные таблицы. Типы данных в ячейках электронной таблицы».	1 час		Практическая работа № 9 «Редактирование и форматирование таблиц»	Аналитическая деятельность: <ul style="list-style-type: none"> анализирует пользовательский интерфейс используемого программного средства; определяет условия и возможности применения программного средства для решения типовых задач; выявляет общее и отличия в разных программных продуктах, предназначенн 			
Тема № 19 «Организация вычислений. Относительные, абсолютные и смешанные ссылки».	1 час		Практическая работа № 10 «Относительные, абсолютные и смешанные ссылки»				
Тема № 20 «Встроенные функции. Логические функции»	1 час		Практическая работа № 11 «Встроенные функции»				
Тема № 21 «Сортировка и фильтрация»	1 час		Практическая работа № 12 «Сор-				

данных в выделенном диапазоне».			тировка и поиск данных»	ых для решения одного класса задач. Практическая деятельность:			
Тема № 22 «Построение диаграмм и графиков».	1 час		Практическая работа № 13 «Построение диаграмм и графиков»	<ul style="list-style-type: none"> создает электронные таблицы, выполняет в них расчёты по встроенным и вводимым пользователем формулам; строит диаграммы и графики в электронных таблицах. 	Как вы думаете, в каких областях деятельности применяются диаграммы?		Кейс № 3 «Создание диаграмм в табличном процессе MS Excel»
Тема № 23 «Контрольная работа № 3 «Информационные технологии».	1 час		Контрольная работа № 3 «Информационные технологии».				
4 раздел «Цифровая грамотность» - 11 часов							
Тема № 24 «Локальные и глобальные компьютерные сети».	1 час			Аналитическая деятельность: <ul style="list-style-type: none"> выявляет общие черты и отличия способов взаимодействия на основе компьютерных сетей; 	Каждый ли компьютер, на ваш взгляд, должен быть подключен к Интернету?		Кейс № 4 «Поиск информации в глобальной сети Интернет»
Тема № 25 «Глобальная сеть Интернет. IP-адреса узлов. Большие данные».	1 час						

Тема № 26 «Доменная система имен. Протоколы передачи данных».	1 час			<ul style="list-style-type: none"> • анализирует доменные имена компьютеров и адреса документов в Интернете; • приводит примеры ситуаций, в которых требуется поиск информации; • анализирует и сопоставляет различные источники информации, оценивает достоверность найденной информации. 			
Тема № 27 «Всемирная паутина. Файловые архивы».	1 час						
Тема № 28 «Виды деятельности в сети Интернет».	1 час				В чем причины агрессии и хамства в сети?		Кейс № 5 «Электронная почта»
Тема № 29 «Технологии создания сайта Облачные технологии. Использование онлайн-офиса для разработки документов».	1 час					Первый миресайт	в
Тема № 30 «Содержание и структура сайта».	1 час		Практическая работа № 14 «Создание комплексных информационных объектов в виде веб-страниц учетом по-		Практическая деятельность: <ul style="list-style-type: none"> • осуществляет взаимодействие посредством электронной почты, чата, форума; • определяет 		

			нения об информационной безопасности»	минимальное время, необходимое для передачи известного объема данных по каналу связи с известными характеристиками;			
Тема № 31 «Оформление сайта».	1 час		Практическая работа № 15 «Оформление сайта»	• проводит поиск информации в сети Интернет по запросам с использованием логических операций;			
Тема № 32 «Размещение сайта в Интернете».	1 час		Практическая работа № 16 «Размещение сайта в Интернете»	• создает с использованием конструкторов (шаблонов) комплексные информационные объекты в виде веб-странички, включающей графические объекты;			
Тема № 33 «Контрольная работа №4 по теме «Цифровая грамотность»».	1 час		Контрольная работа №4 по теме «Цифровая грамотность»»	• проявляет избирательность в работе с			

				информацией, исходя из морально-этических соображений, позитивных социальных установок и интересов индивидуально го развития.			
Тема № 34 «Промежуточная аттестация»	1 час		Контрольная работа № 5 за курс 9 класса				

Тексты для чтения

Представление информации

Чтобы сохранить полученную информацию, необходимо представить её на носителе, то есть сохранить её в виде знаков (букв или цифр), рисунка, схемы, фотографии.

Существуют разные способы представления информации. Рисунок, схема, фотография — примеры графического способа представления информации. Текст — это текстовое представление информации. Числа (цифры) — это числовое (цифровое) представление информации.


Название способа представления информации	Примеры
Графическое представление информации — фотография или рисунок	
Знаковое (текстовое) представление информации — знаки (буквы, числа и другие)	Приятного аппетита!
Числовое (цифровое) представление информации — знаки (цифры)	1234

Рисунок — это наглядный способ представления информации. С помощью рисунка удобно выражать образы, внешние признаки предметов и явлений, движение, настроение. Так, на рисунке изображены два человека. Они общаются. Но главный смысл рисунка в том, что они не понимают друг друга и сердятся, доказывая каждый своё.



Текст обладает меньшей наглядностью, чем рисунок, но текстом можно представить такую информацию, которую нельзя представить рисунком. Например, рисунком нельзя представить имена и фамилии людей, содержание разговора, причину взаимного непонимания и многое другое.

Настроение человека можно представить и текстом, и графически — в виде схемы.

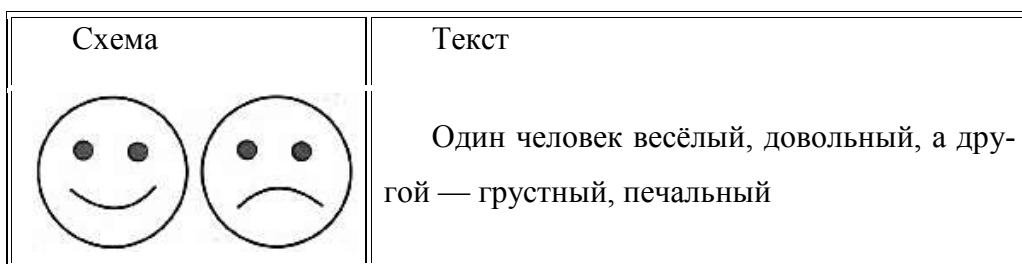


Схема по сравнению с рисунком — более упрощённое представление информации, полученной, например, в результате наблюдения. Упрощённое представление — значит, не содержащее подробностей, не имеющее портретного, внешнего сходства с изображаемым объектом.

Есть ещё и другие способы представления информации. Например, математическое выражение.

Вопросы и задания:

1. Зачем люди представляют информацию?
2. Рассмотрите рисунок. Можно ли представить эту информацию текстом? Если да, то сделайте это в своей рабочей тетради. Представьте эту же информацию в виде текста в текстовом редакторе. Сохраните файл под именем «Представление информации» в папке «Моё портфолио».



Первые расчетные устройства

Давайте посмотрим на историю компьютеров, которые мы знаем сегодня. Самым первым используемым вычислительным устройством были десять пальцев рук человека. Собственно, поэтому сегодня мы все еще считаем десятками и кратными десяткам.

Затем были изобретены счеты. Люди продолжали использовать ту или иную форму счеты еще в 16 веке, и они все еще используются в некоторых частях мира, потому что их можно понять, не зная, как читать. В XVII и XVIII веках многие люди пытались найти простые способы вычислений. Дж. Нэпир, шотландец, изобрел механический способ умножения и деления, который теперь стал современной пластинкой. Генри Бриггс использовал идеи Напьера для создания таблиц логарифмов, которые сегодня используют все математики. Исчисление, еще один раздел математики, независимо изобрели англичанин сэр Исаак Ньютон и немецкий математик Лейбниц. Первая настоящая вычислительная машина появилась в 1820 году в результате экспериментов нескольких людей.

Чарльз Бэббидж, одаренный английский математик, предложил построить универсальную машину для решения задач, которую он назвал «аналитической машиной». Эта машина, которую Бэббидж показал на Парижской выставке в 1855 году, была попыткой полностью избавиться от человека, за исключением предоставления машине необходимых фактов о проблеме, которую нужно было решить. Он так и не закончил эту работу, но многие из его идей легли в основу создания современных компьютеров.

К началу двадцатого века электромеханические машины были разработаны и использовались для обработки бизнес-данных. Доктор Герман Холлерит, молодой статистик из Бюро переписи населения США, успешно составил таблицу переписи 1890 года. Холлерит изобрел способ кодирования данных, пробивая отверстия в карточках. Он построил одну машину, чтобы пробивать дыры, а другие - для табулирования собранных данных. Позже Холлерит покинул Бюро переписи и основал свою собственную компанию по производству счетных машин. В результате серии слияний компания в конечном итоге превратилась в IBM Corporation.

До середины двадцатого века машины, предназначенные для обработки данных перфокарт, широко использовались для обработки бизнес-данных. Эти первые электромеханические процессоры данных назывались машинами для записи единиц, потому что каждая перфокарта содержала единицу данных.

В середине 1940-х годов были разработаны электронные компьютеры для выполнения расчетов в военных и научных целях. К концу 1960-х годов коммерческие модели этих компьютеров широко использовались как для научных вычислений, так и для обработки бизнес-данных. Первоначально эти компьютеры принимали входные данные с перфокарт. К концу 1970-х перфокарты почти повсеместно были заменены терминалами с клавиатурой. С тех пор достижения науки привели к распространению компьютеров в нашем обществе, и прошлое - всего лишь пролог, который дает нам представление о природе.

Вопросы:

1. Что было самым первым счетным устройством?
2. Что такое современная логарифмическая линейка?
3. Как Ньютон и Лейбниц внесли свой вклад в проблему вычислений?
4. В чем заключалась основная идея гл. Машина Бэббиджа?
5. Какие средства кодирования данных изобрел Холлерит?
6. Что нового было в компьютерах 1970-х?

Оборудование и программное обеспечение

Единицы, которые видны на любом компьютере, являются физическими компонентами системы обработки данных или аппаратного обеспечения. Таким образом, устройства ввода, хранения, обработки и управления являются аппаратными. Не видно программного обеспечения - набора компьютерных программ, процедур и связанной документации, которые делают возможной эффективную работу компьютерной системы. Программное обеспечение бывает двух типов: системное программное обеспечение и прикладное программное обеспечение. Системное программное обеспечение - это программы, предназначенные для управления работой компьютерной системы. Они не решают конкретных задач. Они написаны, чтобы помочь людям в использовании компьютерной системы путем выполнения таких задач, как управление всеми необходимыми операциями, перемещение данных в компьютер и из него, а также все этапы выполнения прикладной программы.

Человек, который готовит системное программное обеспечение, называется системным программистом. Системные программисты - высококвалифицированные специалисты и важные члены архитектурной команды. Прикладное программное обеспечение - это программы, написанные для решения конкретных задач (приложений), таких как расчет заработной платы, управление запасами и анализ инвестиций. Слово «программа» обычно относится к прикладной программе, а слово «программист» - это обычно человек, который разрабатывает прикладное программное обеспечение. Часто программы, особенно системное программное обеспечение, хранятся в области памяти, не используемой для прикладного программного обеспечения. Эти защищенные программы хранятся в области памяти, называемой постоянным запоминающим устройством (ROM), из которой можно читать, но нельзя записывать.

Прошивка - это термин, который обычно используется для описания определенных программ, хранящихся в ПЗУ. Прошивка часто относится к последовательности инструкций (программного обеспечения), заменяющей оборудование.

Например, в случае, когда стоимость важнее производительности, архитектор компьютерной системы может решить не использовать специальные электронные схемы (оборудование) для умножения двух чисел, а вместо этого написать инструкции (программное обеспечение), чтобы машина выполняла ту же функцию. путем многократного использования схем, уже предназначенных для выполнения сложения.

Вопросы:

1. Какие задачи выполняет системное программное обеспечение?
2. Какие проблемы решает прикладное программное обеспечение?
3. Как архитектор компьютерной системы может использовать прошивку?

История развития понятия алгоритма

Понятие алгоритма является одним из основных понятий современной математики. Еще на самых ранних ступенях развития математики (Древний Египет, Вавилон, Греция) в ней стали возникать различные вычислительные процессы чисто механического характера. С их помощью искомые величины ряда задач вычислялись последовательно из исходных величин по определенным правилам и инструкциям. Со временем все такие процессы в математике получили название алгоритмов.

Термин алгоритм происходит от имени средневекового узбекского математика Аль-Хорезми, который еще в IX в. дал правила выполнения четырех арифметических действий в десятичной системе счисления.

Вплоть до 30-х годов прошлого века понятие алгоритма имело скорее методологическое, чем математическое значение. Под алгоритмом понимали конечную совокупность точно сформулированных правил, которые позволяют решать те или иные классы задач. Однако в этом определении не содержится точной характеристики того, что следует понимать под классом задач и под правилами их решения. В течение длительного времени математики довольствовались этим определением, поскольку общей теории алгоритмов фактически не существовало. Однако, практически не было серьезных случаев, когда математики разошлись бы во мнении относительно того, является ли алгоритмом тот или иной конкретно заданный процесс.

Положение существенно изменилось, когда на первый план выдвинулись такие алгоритмические проблемы, положительное решение которых было сомнительным. Действительно, одно дело доказать существование алгоритма, другое – его отсутствие. Первое можно сделать путем фактического описания процесса, решающего задачу. В этом случае достаточно и интуитивного понятия алгоритма, чтобы удостовериться в том, что описан-

ный процесс есть алгоритм. Доказать несуществование алгоритма таким путем невозможно. Для этого надо точно знать, что такое алгоритм.

В двадцатых годах прошлого века задача такого определения понятия алгоритма стала одной из центральных математических проблем. Решение ее было получено в середине тридцатых годов в работах известных математиков: Гильберта, Геделя, Черча, Клини, Поста и Тьюринга.

В 50-е годы прошлого столетия существенный вклад в развитие теории алгоритмов внесли работы Колмогорова и Маркова. Формальные модели алгоритмов Поста, Тьюринга и Черча, равно как и модели Колмогорова и Маркова, оказались эквивалентными в том смысле, что любой класс проблем, разрешимых в одной модели, разрешим и в другой.

Хотя первоначально теория алгоритмов возникла в связи с внутренними потребностями теоретической математики (математическая логика, алгебра, геометрия и т.д. остаются и сегодня одной из основных областей приложения теории алгоритмов), другая область ее применения возникла в 40-х годах в связи с созданием быстродействующих электронных вычислительных и управляющих машин. Появление ЭВМ способствовало развитию теории алгоритмов, вызвало к жизни разделы этой теории, имеющие ярко выраженную прикладную направленность. Это, прежде всего, алгоритмические системы и алгоритмические языки, являющиеся основой современной теории программирования для универсальных ЭВМ, и способы точного описания отображений, реализуемых цифровыми автоматами.

Теория алгоритмов оказалась тесно связанной и с рядом областей лингвистики, экономики, физиологии мозга и психологии, философии, естествознания. Примером одной из задач этой области может служить точное описание алгоритмов, реализуемых человеком в процессе умственной деятельности.

Вопросы:

1. Какие математические процессы получили название алгоритмов?
2. Когда было сформулировано точное определение понятия алгоритма?
3. Что повлияло на развитие алгоритмических языков?
4. Какие сферы активно используют теорию алгоритмов?

Компьютерное программирование

Программирование - это процесс подготовки набора закодированных инструкций, которые позволяют компьютеру решать определенные проблемы или выполнять определенные функции. Суть компьютерного программирования - это кодирование программы для компьютера с помощью алгоритмов. Дело в том, что любая проблема выражается ма-

тематически, содержит формулы, уравнения и расчеты ». Но компьютер не может управлять формулами, уравнениями и вычислениями. Любая проблема должна быть специально обработана, чтобы компьютер ее понял, то есть - закодирован или запрограммирован. Фаза, на которой пишутся компьютерные программы системы, называется фазой разработки.

Программы представляют собой списки инструкций, которым будет следовать блок управления центрального процессора (ЦП). Инструкции программы должны быть полными и в соответствующей последовательности, иначе будут получены неправильные ответы. Для защиты от этих логических ошибок и документирования логического подхода программы необходимо разработать логические планы.

Есть два распространенных метода планирования логики программы. Первый метод - это блок-схема. Блок-схема - это план в форме графического или графического представления, в котором используются заранее определенные символы для иллюстрации логики программы. Таким образом, это «картина» логических шагов, которые должен выполнить компьютер. Каждая из предопределенных форм символа обозначает общую операцию. Форма символа передает характер общей операции, а детали записываются внутри символа. Пластиковая или металлическая направляющая, называемая шаблоном, используется для облегчения рисования символов. Второй метод планирования логики программы называется псевдокодом. Псевдокод - это имитация реальных программных инструкций. Это позволяет создавать программную структуру без бремени правил программирования. Псевдокод занимает меньше времени для профессионального программиста, чем создание блок-схем. Он также подчеркивает нисходящий подход к структуре программы. Псевдокод имеет три основные структуры: последовательность, решение и логику цикла. С помощью этих трех структур может быть выражена любая необходимая логика.

Вопросы:

1. В чем суть программирования?
2. Что делать с проблемой перед обработкой на компьютере?
3. Каковы основные методы планирования логики программы?
4. Что такое шаблон и для чего он используется?
5. Каковы основные структуры псевдокода?

Первый в мире сайт

В мире, где существует более 1 миллиарда веб-сайтов, трудно поверить, что когда-то был только один. Так было 6 августа 1991 года, когда без лишнего шума британский ученый-компьютерщик Тим Бернерс-Ли опубликовал первый в истории веб-сайт, работая в ЦЕРНе, огромной лаборатории физики элементарных частиц в Швейцарии. В течение

десятилетия дебютировали такие сайты, как eBay, Amazon и Google, и Интернет произвел революцию в обществе, изменив способы общения, совершения покупок и получения информации.

Бернерс-Ли, сын компьютерных ученых, родился в Лондоне в 1955 году (в том же году, что и Стив Джобс и Билл Гейтс) и изучал физику в Оксфорде. Работая в ЦЕРН в 1980-х, Бернерс-Ли заметил, насколько сложно было отслеживать проекты и компьютерные системы тысяч исследователей организации, разбросанных по всему миру. Как он позже заявил: «В те дни на разных компьютерах была разная информация, но для доступа к ней приходилось входить в систему на разных компьютерах. Кроме того, иногда вам приходилось изучать разные программы на каждом компьютере».

Этот компьютер был использован в ЦЕРН британским ученым Тимом Бернерсом-Ли для создания всемирной паутины (WWW). В марте 1989 года Бернерс-Ли представил менеджерам ЦЕРН предложение о системе управления информацией, которая использовала бы гипертекст для связывания документов на разных компьютерах, подключенных к Интернету. (Гипертекст, термин, появившийся в 1963 году, позволяет человеку получить документ или фрагмент контента, щелкнув закодированное слово или фразу.) Его начальник назвал это «расплывчатым, но захватывающим» предложением сначала не было принято. Бернерс-Ли объединился с Робертом Кайо, бельгийским инженером из CERN, чтобы доработать предложение, и в 1990 году начальник англичанина дал ему время для работы над проектом. Первоначально назвав проект «Управление информацией», Бернерс-Ли опробовал такие названия, как «Шахта информации» и «Информационная сетка», прежде чем остановиться на WorldWideWeb.

К концу 1990 года Бернерс-Ли, используя компьютер NeXT, разработанный Стивом Джобсом, разработал ключевые технологии, лежащие в основе Интернета, включая язык гипертекстовой разметки (HTML), для создания веб-страниц; Протокол передачи гипертекста (HTTP), набор правил для передачи данных через Интернет; и унифицированные указатели ресурсов (URL) или веб-адреса для поиска документа или страницы. Он также разработал базовый браузер и программное обеспечение веб-сервера.

Начало Интернета как общедоступной службы в Интернете наступило 6 августа 1991 года, когда Бернерс-Ли опубликовал первый в истории веб-сайт. Соответственно, сайт был посвящен проекту WorldWideWeb, описывая Интернет и способы его использования. Этот сайт размещался в ЦЕРН на компьютере NeXT Бернерса-Ли и имел адрес <http://info.cern.ch>.

Бернерс-Ли не стал нажиться на своем изобретении и отклонил призыв ЦЕРН запатентовать его веб-технологии. Он хотел, чтобы Интернет был открытым и бесплатным,

чтобы он мог расширяться и развиваться как можно быстрее. Как он позже сказал: «Если бы технология была запатентованной и находилась под моим полным контролем, она бы, вероятно, не получила широкого распространения. Вы не можете предложить что-то универсальное пространство и в то же время держать его под контролем».

В 1993 году команда из Национального центра суперкомпьютерных приложений Университета Иллинойса выпустила Mosaic, первый веб-браузер, ставший популярным среди широкой публики. В следующие несколько лет были запущены такие веб-сайты, как Yahoo (1994), Amazon (1995), eBay (1995) и Google (1998). К тому времени, когда Facebook дебютировал в 2004 году, по данным InternetLiveStats, насчитывалось более 51 миллиона веб-сайтов.

Тем временем в 1994 году Бернерс-Ли покинул ЦЕРН и перешел в Массачусетский технологический институт, где он основал Консорциум WorldWideWeb (W3C), организацию, которая поддерживает стандарты для Интернета. Этот скромный провидец был назван одним из 100 самых важных людей XX века по версии журнала Time, а в 2004 году был посвящен в рыцари королевой Елизаветой II. В 2009 году Бернерс-Ли основал WorldWideWebFoundation, организацию, деятельность которой направлена на то, чтобы Интернет приносил пользу человечеству. Во время церемонии открытия летних Олимпийских игр 2012 года в Лондоне он был удостоен чести, изобрел Интернет и написал в Твиттере: «Это для всех».

Вопросы:

1. Когда был опубликован первый веб –сайт?
2. Кто из британских ученых хотел, чтобы Интернет был открытым и бесплатным?
3. Как назывался первый браузер?

Фрактальная графика

Математика буквально пронизана гармонией, и графика фрактальная – прямое тому подтверждение. Наука присутствует при создании каждого ее элемента, поэтому она отражает всю красоту. Создатель фрактальной геометрии, профессор Мальдерброт, писал в своих книгах, что рассматриваемая графика представляет собой не просто повторяющиеся изображения. Это – структура любого существа или объекта на планете, живого и неживого. К примеру, ДНК является основой, одной интеграцией. Но если код начинает повторяться, тогда появляется человек.

Что такое фрактальная графика? Это одна или несколько геометрических фигур, каждая из которых подобна другой. То есть, изображение составляется из одинаковых частей.

Само слово "фрактал" может употребляться, если фигура обладает одним или несколькими из этих свойств: Нетривиальная структура. Когда рассматривается небольшая деталь всего изображения, то фрагмент схож со всем рисунком. Увеличение масштаба не приводит к ухудшению. Изображение всегда остается одинаково сложным. Каждая часть рисунка является самоподобной. Имеется математическая размерность. Строится при помощи повторения. Множество объектов природного или искусственного происхождения наделяются свойствами фракталов. К ним относятся кровеносные системы человека и животного, кроны и корни деревьев и так далее. Фрактальная компьютерная графика становится популярной потому, что добиться красоты и реалистичности можно посредством простого построения при помощи соответствующего оборудования. Нужно только задать правильную математическую формулу и указать количество повторений.

Создание фрактальной графики будет различаться в зависимости от ее классификации: геометрическая, алгебраическая или стохастическая. Несмотря на разницу, итог всегда будет одинаковым. Поскольку фрактальная графика начинается с геометрии, то следует рассмотреть ее создание на соответствующем примере: Задают условие. Это фигура, на основе которой будет строиться все изображение. Задают процедуру. Она преобразует условие. Получают геометрический фрактал. Обычно нулевое условие представляется в виде треугольника.

Чтобы построить изображение, нужно применить две процедуры. Во-первых, DrawTriangle. Она строит треугольник по точкам, заданным пользователем. Во-вторых, DrawGenerator. Она указывает количество точек. Каждая процедура может повторяться несколько раз или бесконечно долго. Для определения этого показателя применяется численный аргумент n .

После того как элемент фрактальной графики был создан, с ним можно производить различные дополнительные действия: Повороты и растяжения увеличивают отдельные детали рисунка, либо они принимают нужную пользователю форму. Группирование объектов обычно применяется для того, чтобы назначить требуемый масштаб. Преобразование цветов. Изображение можно окрасить в любой оттенок, задать тон. Изменение формы всего объекта или отдельных деталей. Нужно помнить, что изображения фрактальной графики в конечном итоге предсказать невозможно. Когда треугольник слишком увеличивается, то просмотр будет нереальным, пользователь увидит только черное окно. Когда желаемая текстура обнаружена, все изменения с ней нужно проводить в минимальном порядке, постоянно сохраняя допустимый вариант.

Нет такого человека, которого бы не привлекала фрактальная графика. Программы, участвующие в ее создании, представлены в большом количестве. Поэтому надо разобратся в наиболее подходящих для новичков.

Продукт ArtDabbler представляет собой лучший вариант, если пользователь раньше не имел дело с его аналогами. Здесь можно не только освоить графику, но и научиться рисовать на компьютере. К другим преимуществам следует отнести небольшое количество занимаемой памяти и интуитивно понятный интерфейс. Другая программа – UltraFractal. Она уже ориентирована на работу профессионалов, новичкам сложно будет в ней разобратся. Интерфейс здесь достаточно сложный, но производители выполнили его на примере обычного Photoshop. Если пользователь имел дело с этой программой, то в кнопках разберется быстро. Особенность UltraFractal заключается в том, что здесь выполняется не только графика фрактальная в качестве стандартного и обычного изображения, но и анимация. Формулы для составления прилагаются, но при необходимости пользователь сможет задействовать свою.

Форматы фрактальной графики определяют форму и способ хранения файловых данных. Некоторые из них включают в себя большой объем информации. Поэтому их необходимо сжимать. Причем делать это не посредством архивирования, а непосредственно в файле. Если правильно его выбрать, то сжатие будет происходить автоматически. Есть несколько алгоритмов этой процедуры. Если перед пользователем аппликация, большая часть которой выдержана в одном цвете, то разумно использовать форматы BMP и PCX. Здесь заменяется последовательность повторяющихся величин. Диаграмму, которая очень редко, но все-таки используется во фрактальной графике, логично поместить в TIFF или GIF. Часть форматов является универсальной. То есть, их можно просмотреть в большинстве редакторов. Но если пользователю важна качественная обработка изображений, тогда нужно применять оригинальную программу. Форматы фракталы не поддерживаются браузерами. Именно поэтому осуществляется их преобразование, если есть необходимость загрузить на тот или иной сайт.

Применение фрактальной графики можно назвать фактически повсеместным. Более того, эта область постоянно расширяется. На данный момент можно отметить следующие области: Компьютерная графика. Реалистично изображаются рельефы и природные объекты. Это применяется в создании компьютерных игр. Анализ фондовых рынков. Фракталы здесь используются для того, чтобы отметить повторения, которые впоследствии сыграют трейдерам на руку. Естественные науки. В физике с помощью фрактальной графики моделируются нелинейные процессы. В биологии она описывает строение кровеносной системы. Сжатие изображений, чтобы уменьшить объем информации. Создание децентра-

лизованной сети. Посредством фракталов удастся обеспечить прямое подключение, а не через центральное регулирование. Поэтому сеть становится более устойчивой. На данный момент практикуется применение фракталов в производстве различного оборудования. Например, уже запущен конвейер по созданию антенн, отлично принимающих сигналы. Примеры фрактальной графики распространены от примитивных до очень сложных повторяющихся элементов. Уникальной особенностью данного типа является то, что рисунок можно составить исключительно из восклицательных или вопросительных знаков. Стандартными, но относительно сложными примерами компьютерной фрактальной графики являются облака, горы, морские побережья и так далее. Их зачастую используют при создании игр. Самым простым примером можно назвать кривую Коха. Во-первых, она не имеет конкретной длины, и ее называют бесконечной. Во-вторых, здесь полностью отсутствует гладкость. Поэтому невозможно построить касательную.

Свое распространение совсем недавно заполучила фрактальная графика. Достоинства и недостатки ее слишком размыты, поскольку отсутствует нормальная теоретическая база. Терминология и принципы ее использования до конца не изучены, несмотря на то, что они действенные и рабочие. Достоинства фрактальной графики заключаются в нескольких факторах: Небольшой размер при масштабном рисунке. Нет конца масштабированию, сложность картинку можно увеличивать бесконечно. Нет другого такого же инструмента, который позволит создавать сложные фигуры. Реалистичность. Простота в создании работ. Недостатки фрактальной графики тоже присутствуют. Во-первых, без компьютера здесь не обойтись. Причем, чем длиннее количество повторений, тем больше загружается процессор. Соответственно, только качественное компьютерное оборудование способно справиться с построением сложных изображений. Во-вторых, присутствуют ограничения в исходных математических фигурах. Некоторые изображения создать посредством фракталов не удастся.

Векторная и фрактальная графика очень различаются между собой: Вектор задействует контуры разных геометрических фигур, фрактал – математическую формулу, в основе которой лежит треугольник. Вектор используют везде, где нужно получить четкий контур. Фрактальная графика более специализирована, она нашла свое применение в математике и искусстве. Векторными аналогами являются слайды или функции на графиках. У фракталов это – снежинки или кристаллы. Несмотря на многообразие отличительных черт, эти два вида графики объединяет качество изображения. Оно остается неизменным, независимо от уровня масштабирования. Трехмерная, векторная, растровая, фрактальная графика схожи в одном – все они широко используются в решении различных компьютерных задач. Чтобы получить действительно качественное изображение, нужно задей-

ствовать каждую из них. Графика фрактальная не имеет аналогов. Она уникальна в своем роде. Во-первых, один ее небольшой участок может рассказать сразу обо всем рисунке или изображении. Информация обо всем фрактале доступна, т.к. он является самоподобным. В центре любого изображения, относящегося к данному типу графики, располагается равносторонний треугольник. Все остальные детали рисунка являются либо его частями, либо уменьшенными/увеличенными копиями. То есть, в составлении изображения принимает участие один конкретный элемент. Для того чтобы использовать фрактальную графику, не нужны никакие объекты, хранящиеся в памяти компьютера. Приступить к созданию можно, имея под рукой одну только математическую формулу.

Графика фрактальная очень реалистична. Происходит это потому, что ее детали и элементы постоянно встречаются в окружении человека – горы, облака, морские берега, различные природные явления. Часть из них остается постоянно в одном и том же состоянии, вроде деревьев, каменистых участков. Остальные же непрерывно меняются, как мерцающее огненное пламя или кровь,двигающаяся по сосудам. Развитие фрактальных технологий на сегодняшний день – одна из прогрессирующих областей науки. Она используется не только в компьютерной графике. Возможно, если ученым удастся докопаться до их сути, человек начнет намного лучше понимать этот мир.

Вопросы:

1. Может ли фрактальная графика быть анимационной?
2. Где находит применение фрактальная графика?
3. Какими недостатками наделена фрактальная графика?
4. Что необходимо для создания фрактала?
5. Почему фрактальная графика реалистична?

Приложение 2

Кейсы для организации проектной и исследовательской деятельности

7 класс

Кейс № 1 «Источники информации».

В редакцию детского журнала пришло письмо следующего содержания. «Здравствуй, дорогая редакция! Меня зовут Маша. Мне 7 лет. На днях мы с моим братом Сережей (ему 10 лет) помогли маме делать генеральную уборку. Когда дело дошло до книжных шкафов, Сережа сказал: «Как здорово, что через 5-10 лет никаких книг не будет вообще, их вытеснят разные гаджеты». Мы с мамой долго спорили с ним, что книги будут существовать всегда, но так и не смогли убедить его.

Вопрос кейса: Дорогая редакция! А что ты думаешь по этому поводу? Неужели книги, и в самом деле, исчезнут? А я так люблю читать!» Журналисту, ведущему рубрику «Спрашивай – отвечаем», необходимо подготовить небольшую аргументированную статью-ответ.

Кейс № 2 «Как отличить лицензионный диск от пиратского»

Испанские исследователи продемонстрировали простой и недорогой способ проверки подлинности товаров на оптических носителях. Установить происхождение компакт-дисков помогло понимание технологии их изготовления и простая лазерная указка.

Отличить пиратский CD от лицензионного по внешнему виду невозможно, ведь имитация фабричной упаковки и голографических наклеек весьма качественна. Само явление дифракции и помогло учёным признать разницу между дисками.

Вопрос кейса: Представьте себя испанскими учёными и объясните характерные отличия между этими двумя дисками.

Кейс № 3 «Графический редактор. Создание рисунков в графическом редакторе»

Сейчас вы художники. Ваша задача — изобразить на полотне, как вы себе представляете свою родину. Просмотрите видеофильм о России (на усмотрение учителя). Надеюсь, что этот видеофрагмент поможет вам с выбором сюжета будущего шедевра.

Вопрос кейса: А как вы себе представляете свою родину? Изобразите её, используя средства графического редактора.

Кейс № 4 «Текстовый редактор. Вставка символов, которых нет на клавиатуре»

Работник типографии должен набрать текст статьи, содержащей много символов, которых нет на клавиатуре. Использование таблицы, которая содержится в офисных программах и предназначена для вставки символов, отсутствующих на клавиатуре, требует определённых временных затрат на поиск нужного символа. Что нужно сделать работнику типографии, чтобы быстро вставить в текст такие символы, как тире, среднее тире, плюс, минус, параграф и знак умножения?

Вопрос кейса: Какая специальная таблица предусмотрена в офисных программах для вставки символов, отсутствующих на клавиатуре? Можно ли символы в Word вставить, используя кодировку Unicode? в каких текстовых редакторах возможно использовать кодировку Unicode?

8 класс

Кейс № 1 «Системы счисления»

Ежегодно в ГБПОУ «Сергиевский губернский техникум» проходит декада математических и общих естественнонаучных дисциплин. В рамках указанной декады преподаватели проводят олимпиаду по математическим дисциплинам. В 2013-2014 учебном году среди олимпиадных заданий по дисциплине «Математика и информатик» были следующие:

1) Восстановите цифры двоичной системы счисления, на месте которых в арифметических выражениях стоит знак "*".

$$а) **0*0*1**1_2 + 10111*10**_2 = 100*1*00010_2$$

$$б) ***0**00_2 - 11*11*11_2 = 1101*1_2$$

2) В классе 1000112 учеников. 1111002% из них учатся на хорошо и отлично.

Сколько учеников учатся на хорошо и отлично?

Студенты первокурсники плохо справились с данными заданиями, несмотря на то, что задания относились к уровню низкой сложности.

Что помешало студентам решить задания?

Вопросы и задания:

1. Составьте конспект по теме «Позиционные системы счисления и действия над числами в позиционных системах счисления, отличных от десятичной».

2. Выполните задания и изложите решение, включая основания для него (письменная форма).

Кейс № 2 «Алгоритмы»

Лекция 1. Этапы решения задач на компьютере. Алгоритм и его свойства. Способы записи алгоритмов.

На первом этапе участвует человек, хорошо представляющий предметную область задачи.

Он должен четко определить цель задачи, дать словесное описание содержания задачи и предложить общий подход к её решению.

Для задачи вычисления суммы двух целых чисел человек, знающий, как складываются числа, может описать задачу следующим образом: ввести два целых числа, сложить их и вывести сумму в качестве результата решения задачи.

Цель второго этапа – создать такую математическую модель решаемой задачи, которая может быть реализована в компьютере.

Существует целый ряд задач, где математическая постановка сводится к простому перечислению формул и логических условий.

Этот этап тесно связан с первым этапом, и его можно отдельно рассматривать, однако возможно, что для полученной модели известны несколько методов решения, и тогда предстоит выбрать лучший.

Для вышеописанной задачи данный этап сведется к следующему: введенные в компьютер числа започнем в памяти под именами А и В, затем вычислим значение суммы этих чисел по формуле $A+B$, и результат започнем в памяти под именем S.

На третьем этапе на основе математического описания необходимо разработать алгоритм решения.

Составление программы на четвертом этапе обеспечивает возможность выполнения алгоритма и соответственно поставленной задачи исполнителем – компьютером.

Во многих задачах при программировании на алгоритмическом языке часто пользуются заменой блока алгоритма на один или несколько операторов, введением новых блоков, заменой одних блоков другими. Программа и исходные данные вводятся в ЭВМ с клавиатуры с помощью редактора текстов, и для постоянного хранения осуществляется их запись на гибкий или жесткий магнитный диск.

На шестом этапе происходят выполнение алгоритма с помощью ЭВМ, поиск и исключение ошибок.

При этом программисту приходится выполнять рутинную работу по проверке работы программы, поиску и исключению ошибок, и поэтому для сложных программ этот часто требует гораздо больше времени и сил, чем написание первоначального текста программы.

Далее программист запускает программу и задает исходные данные, требуемые по условию задачи.

Алгоритм — это понятное и точное указание исполнителю совершить последовательность действий, направленных на решение поставленной задачи.

Термин имеет интересное историческое происхождение. В IX веке великий узбекский математик аль-Хорезми разработал правила арифметических действий над десятичными числами. Совокупность этих правил в Европе стали называть "алгоритм". Впоследствии слово трансформировалось до известного нам сейчас вида и, кроме того, расширило свое значение: алгоритмом стали называть любую последовательность действий (не только арифметических), которая приводит к решению той или иной задачи. Можно сказать, что понятие вышло за рамки математики и стало применяться в самых различных областях.

Для того чтобы произвольное описание последовательности действий было алгоритмом, оно должно обладать следующими свойствами.

Дискретность

Процесс решения задачи должен быть разбит на последовательность отдельных шагов, каждый из которых называется командой. Примером команд могут служить пункты инструкции, нажатие на одну из кнопок пульта управления, рисование графического примитива (линии, дуги и т.п.), оператор языка программирования. Наиболее существенным здесь является тот факт, что алгоритм есть последовательность четко выделенных пунктов, — такие "прерывные" объекты в науке принято называть дискретными.

Понятность

Каждая команда алгоритма должна быть понятна тому, кто исполняет алгоритм; в противном случае эта команда и, следовательно, весь алгоритм в целом не могут быть выполнены. Данное требование можно сформулировать более просто и конкретно. Составим полный список команд, которые умеет делать исполнитель алгоритма, и назовем его системой команд исполнителя (СКИ). Тогда понятными будут являться только те команды, которые попадают в этот список. Именно из такой формулировки становится ясно, почему компьютер такой "привередливый" при приеме введенных в него команд: даже если неверно написана всего одна буква, команда уже не может быть обнаружена в СКИ.

Команды, образующие алгоритм (или, можно сказать, входящие в СКИ), должны быть предельно четкими и однозначными. Их результат не может зависеть от какой-либо дополнительной информации извне алгоритма. Сколько бы раз вы не запускали программу, для одних и тех же исходных данных всегда будет получаться один и тот же результат.

Результативность

Результат выполнения алгоритма должен быть обязательно получен, т.е. правильный алгоритм не может обрываться безрезультатно из-за какого-либо непреодолимого препятствия в ходе выполнения. Кроме того, любой алгоритм должен завершиться за конечное число шагов.

Корректность

Любой алгоритм создан для решения той или иной задачи, поэтому нам необходима уверенность, что это решение будет правильным для любых допустимых исходных данных. Указанное свойство алгоритма принято называть его корректностью

Массовость

Алгоритм имеет смысл разрабатывать только в том случае, когда он будет применяться многократно для различных наборов исходных данных.

Способы записи алгоритмов:

1. Словесный способ записи

2. Запись на алгоритмическом языке
3. Запись на языке блок-схем
4. Запись на языке программирования

Задания:

1. Которые из документов являются алгоритмами?
 - 1) Программа телепередач
 - 2) Правило правописания приставок, оканчивающихся на з, с
 - 3) Кулинарный рецепт приготовления блюда
 - 4) Инструкция по сборке проданного в разобранном виде шкафа
 - 5) Порядок набора международного телефонного номера
 - 6) Настенный календарь на текущий год
 - 7) Каталог книг в библиотеке
 - 8) Рецепт приготовления клея
 - 9) Правила игры в футбол
 - 10) Политическая карта мира
 - 11) Телефонный справочник
 - 12) Файл Readme, содержащий информацию об установке программы
 - 13) Вычисление корней квадратного уравнения
 - 14) Решение шахматной задачи
 - 15) Инструкция по сборке видеомаягнитофона
 - 16) Морфологический разбор слова
2. Установите соответствие между свойствами алгоритма и ситуациями, в которых эти свойства были нарушены. Переставьте строки второго столбца, чтоб они соответствовали строкам первого. Запишите в ответе последовательность из пяти букв, соответствующую последовательности названий строк второго столбца.

1. Результативность	А. Компьютер посчитал результат вычислений, но не вывел его на экран.
2. Конечность	Б. Программист составил программу для одного конкретного значения исходных данных.
3. Массовость	В. В алгоритме в одной из строк программист написал «И так далее».
4. Дискретность	Г. В программе для Черепашки кто-то вместо команды НАПРАВО написал ВПРАВО.

5. Понятность	Д. В инструкции по приготовлению горячего блюда содержался пункт «Ждать, пока закипит», но оказалась ошибочно пропущена строка «Включить плиту».
---------------	--

3. Вы обучаете Светлого Робота перемещаться по клетчатому полю, в котором между соседними клетками могут располагаться стены.

Начальное положение Светлого Робота – клетка a1.

Робот умеет выполнять команды:

- (1) – передвигается на одну клетку вверх;
- (2) - передвигается на одну клетку вниз;
- (3) - передвигается на одну клетку вправо;
- (4) - передвигается на одну клетку влево.

В какую клетку попадёт Робот, если выполнит команды: 1111314?

6			█		●	
5	█					
4		█	█	█		●
3						
2						
1				●		
	a	b	c	d	e	f

Лекция 2. Виды алгоритмов. Запись алгоритмов с помощью блок-схем

Виды алгоритмов

Название	Определение	Примеры
Линейный (последовательный)	Действия выполняются однократно в заданном порядке	Приготовление одного бутерброда
Циклический	Действия повторяются указанное число раз в заданном порядке	Создание рисунков из повторяющихся элементов
Разветвляющийся (условный)	В зависимости от условия выполняется либо одна, либо другая последовательность действий	Правило правописание приставок, оканчивающихся на з, с

Для более наглядного представления алгоритма используется графический способ.

Существует несколько способов графического описания алгоритмов. Наиболее широко

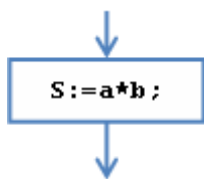
используемым на практике графическим описанием алгоритмов является использование блок-схем. Несомненное достоинство блок-схем – наглядность и простота записи алгоритма.

Каждому действию алгоритма соответствует геометрическая фигура (блочный символ). Перечень наиболее часто употребляемых символов приведен в таблице:

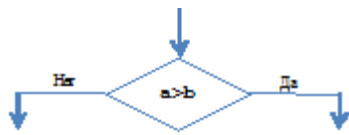
Название символа	Обозначение и пример заполнения	Пояснения
Пуск-останов		Начало, завершение алгоритма или подпрограммы
Ввод-вывод данных		Ввод исходных данных или вывод результатов
Процесс		Внутри прямоугольника записывается действие, например, расчетная формула
Решение		Проверка условия, в зависимости от которого меняется направление выполнения алгоритма
Модификация		Организация цикла
Предопределенный процесс		Использование ранее созданных подпрограмм
Комментарий		Пояснения

Пояснения:

- блок Процесс обозначает вычислительный процесс и применяется для обозначения действия или последовательности действий, изменяющих значения переменных или данных

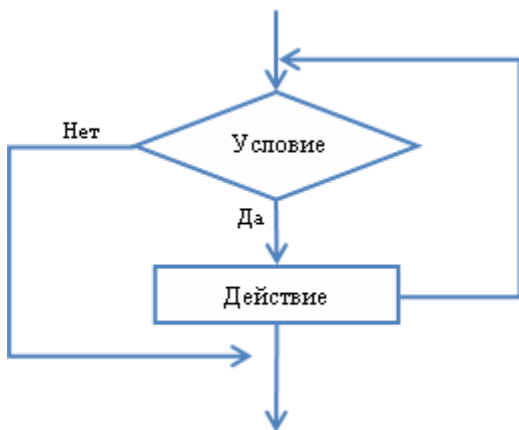


- блок Решение обозначает проверку условия

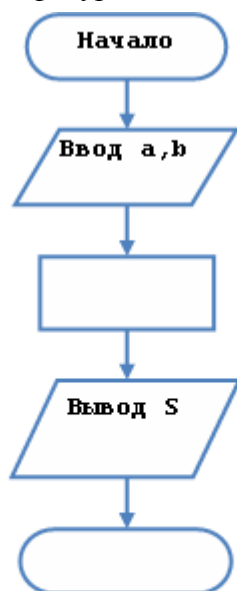


Если условие выполняется, то есть $a > b$, то следующим выполняется действие по стрелке «Да». Если условие не выполняется, то осуществляется переход по стрелке «Нет».

- блок Модификация используется для организации циклических (повторяющихся) действий.



- блок Предопределенный процесс используется для указания обращений к ранее созданным алгоритмам и программам, в том числе и библиотечным подпрограммам.
- блок Ввод-Вывод. При решении задачи на компьютере ввод исходных данных может осуществляться различными способами, например, с клавиатуры, с жесткого диска, с флэш-карты т. д. Задание численных значений исходных данных называется вводом, а отображение результатов расчета на экране монитора или с помощью принтера на бумаге – выводом. Если ввод-вывод не привязан к конкретному устройству, то обозначается параллелограммом. Если необходимо указать конкретное устройство ввода или вывода, то используются специальные геометрические фигуры.



В качестве примера графического способа описания алгоритмов с помощью блок-схем запишем алгоритм нахождения площади прямоугольника:

Внутри каждого блока записывается соответствующее действие. Последовательность выполнения задается соединительной линией со стрелочкой.

Последовательность выполнения сверху вниз и слева направо принята за основную.

Рис. 232. Алгоритм вычисления площади прямоугольника.

Если в алгоритме не нарушается основная последовательность, то стрелочки можно не указывать. В остальных случаях последовательность выполнения блоков обозначается стрелочкой обязательно. В нашем примере основная последовательность выполнения – сверху вниз.

Задания:

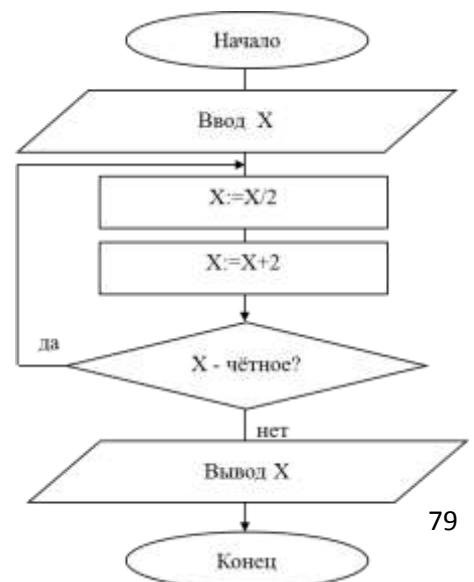
1. Чему будет равно значение переменной s, если a=5, b=7?



2. Чему будет равно значение переменной c, если a=5, b=7?



3. Сколько раз выполнится тело цикла для числа 52? В ответе запишите число повторов.



Кейс № 3 «Основные понятия языка Паскаль»

Алфавит языка - это конечный набор символов, которые используются при написании любой конструкции на языке.

Алфавит языка Паскаль можно разбить на 3 группы:

- a. буквы строчные и прописные буквы латинского и русского алфавита;
- b. цифры - арабские 0..9
- c. специальные символы - (+, -, *, /, =, ,, ::, {} ' := ()), служебные слова: and, begin, const, div, do, else, for, и т.д.

Переменная - это область памяти, названная собственным именем, которая может менять свое значение в процессе выполнения программы. Переменная характеризуется именем, типом и значением.

Константа - не изменяет своего значения в процессе выполнения программы, она может быть задана явно своим значением или обозначена именем.

Имя (идентификатор) переменной или константы задается латинскими буквами и арабскими цифрами. В качестве идентификатора нельзя использовать служебные слова. Идентификатор должен быть уникальным, т.е. в данном блоке программы один идентификатор не может быть использован для обозначения более чем одного объекта.

Например: Zap, gor, X, p1, summa, a28, rar_1, proba, x1, y1, max, min ит.д.

Типы данных

В Паскале существуют следующие стандартные типы данных:

- целочисленный (integer);
- вещественный (real);
- логический (boolean);
- символьный (char);
- строка (string).

Итак, если переменная принадлежит типу integer, то она принимает только целочисленные значения, если принадлежит логическому типу boolean, то принимает значения либо true, либо false и т.д.

В отличие от языка Паскаль, где определен один целый тип Integer и один вещественный тип Real, в TurboPascal имеется 5 стандартных целых типов и 5 стандартных вещественных типов. Их рассмотрим позже.

Из вышеперечисленных типов целые типы – логический и символьный - относятся к порядковым типам, т.е. каждый из них имеет конечное число возможных значений.

Вещественные типы тоже имеют конечное число значений, но количество возможных значений настолько велико, что сопоставить с каждым из них целое число (его номер) невозможно.

Порядковые и вещественные типы относятся к простым типам.

Выражения

Арифметические выражения строятся из чисел и переменных с помощью знаков арифметических операций:

Сложение +, вычитание -, умножение *, деление/, целочисленное деление div, вычисление остатка от целочисленного деления mod.

Div – это целочисленное деление, вычисляющее частное от деления двух натуральных чисел.

Пример: 6:4=1 (остаток 2);

6 div 4=1 (целое частное);

Mod – это вычисление остатка от целочисленного деления

Пример: 6 mod 4=2 (остаток).

В выражениях могут быть использованы круглые скобки и некоторые функции. Арифметические функции можно использовать только с величинами целого и вещественного типов (таблица 1)

таблица 1

Арифметические функции

Математическая запись	Запись на Паскале	Тип результата
$ x $	abs(x)	совпадает с типом x
x^2	sq(x)	-----//-----
\sqrt{x}	sqrt(x)	вещественный
$\sin x$	sin(x)	-----//-----
$\cos x$	cos(x)	-----//-----
arctg x	arctan(x)	-----//-----
e^x	exp(x)	-----//-----
$\ln x$	ln(x)	-----//-----
дробная часть числа	frac(x)	-----//-----
целая часть числа	int(x)	-----//-----
$\pi=3.1415926535$	pi	-----//-----
округление вещественного числа до ближайшего	round(x)	целый

целого		
получение целой части вещественного числа	trunc(x)	-----//-----

Для возведения в произвольную степень используется формула:

обобщенная показательно-степенная функция

$$x^8 = e^{8 \ln x}, \text{ на Паскале запишется так: } \exp(8 * \ln(x)).$$

Вычисление значений выражений выполняется в определенном порядке. Все действия выполняются в соответствии с их приоритетом:

1. вычисления в круглых скобках;
2. вычисления значений функций;
3. *, /, div, mod;
4. +, -.

Выражения на Паскале записываются в одну строку. Для указания правильного порядка действий используются круглые скобки.

Пример: записать на Паскале арифметические выражения:

$$1. \frac{6 \cos(x) - x^2}{\sqrt{2 \sin(x)}} \quad 2. \frac{6 y^3}{\ln(x)}$$

Итак, на Паскале запишутся они так:

1. (6*cos(x) *sqr(x))/sqrt(2*sin(x));
2. (6*exp(3*ln(y)))/ln(x);

Первая программа на языке Pascal. Линейный алгоритм

Программа на языке Pascal состоит из двух разделов – раздела описаний и раздела операторов или, как часто выражаются программисты, главной программы. В разделе описаний мы сообщаем компьютеру о том, какие данные нам понадобятся для выполнения программы и другую необходимую информацию. Данные, которые меняются в процессе выполнения программы, называются переменными. Например, мы хотим написать программу, которая вычисляет площадь и периметр квадрата. Очевидно, что для решения этой задачи нам необходимы три переменные: для стороны квадрата, для площади и для периметра. Назовем их соответственно a, s и p.

Далее: мы должны определить, какого типа это будут переменные. В языке Pascal есть несколько типов переменных, с которыми мы познакомимся в ходе обучения. В данном случае мы имеем дело с вещественными переменными, что и должны отразить в нашей программе. Связано это вот с чем. Когда мы описываем переменную, компьютер отводит под нее память. Понятно, что для, например, целой и вещественной переменной требуется

разная структура памяти. Для целого числа не нужно отводить память под десятичный разделитель (в языке Pascal это точка, а не запятая, к которой мы привыкли) и для знаков «после запятой». Вещественные переменные в языке Pascal описываются словом *real*.

Приступим теперь к созданию текста программы вычисления площади и периметра квадрата. В ранних версиях языка Pascal текст программы должен начинаться со слова *program* и имени программы. Например:

```
Programkvadrat;
```

Седьмая версия не требует обязательно начинать программу со слова *program*. В ней можно сразу начинать программу с раздела описаний.

Описание переменных начинается со слова *var* (от англ. *Variable* – переменная)

Итак, набираем в окне редактор текст:

```
Vara,s,p: real;
```

После слова *var* мы должны поместить хотя бы один список переменных. Переменные в списке перечисляются через запятую. Перед их типом ставится двоеточие. Точка с запятой является разделителем, означающим окончание списка. В тексте главной программы точка с запятой означает окончание той или иной операции

Обозначения *a,s,p* в языках программирования называются идентификаторами. Язык Pascal не различает заглавные или строчные буквы: *a* и *A* для него один и тот же идентификатор.

Есть три правила, по которым создаются идентификаторы в языке Pascal:

1. Идентификатор может содержать только буквы и цифры.
2. Идентификатор начинается только с буквы (*a2* – правильно, *2a* – неправильно)
3. Нельзя использовать в качестве идентификатора зарезервированные слова из языка Pascal (например, *begin*, *end*, *var* и т.д.).

Можно использовать в идентификаторе подчеркик. Например, можно ввести такой идентификатор: *my_program*. Нельзя использовать один и тот же идентификатор дважды для обозначения двух разных переменных. Язык Pascal не чувствителен к регистру: он не различает прописные и строчные буквы. Рекомендуем создавать осмысленные идентификаторы. Так легче в большой программе вспомнить их значение.

После этих предварительных замечаний мы можем продолжить писать вашу программу. Переходим теперь к разделу операторов.

Главная программа начинается со слова *begin* и заканчивается словом *end*. В конце главной программы ставится точка. Между этими словами расположены команды-операторы, которые будут выполняться при исполнении программы. В целом программа должна выглядеть так:

```

{раздел описаний}
Var
a,s,p:real;
{главная программа}
begin
{операторы}
end. {точка после end обязательна в конце программы}

```

Пришло время разработать алгоритм нашей программы. Ее назначение мы определили: она будет вычислять площадь и периметр квадрата. Какие же действия и в какой последовательности мы должны выполнить?

Наша программа ориентирована на некоторого пользователя, который должен ввести в нее исходные данные – сторону квадрата. Следовательно, пользователя нужно пригласить сделать это, для чего следует вывести на экран соответствующий текст. Например: «Введите длину стороны квадрата». Вывод на экран осуществляется в специальном пользовательском окне. Вы можете увидеть его, набрав комбинацию клавиш Alt+F5. Далее, необходимо ввести в компьютер длину стороны квадрата. Потом проводятся вычисления и, наконец, нужно вывести на экран значения площади и периметра квадрата.

Эта последовательность действий не зависит от значений входных данных. Такой алгоритм называется в программировании линейным.

Приведем теперь полный текст программы и поясним значение каждого оператора. Вам следует набрать этот текст на своем компьютере. Текст в фигурных скобках называется комментариями. Его видит программист, но не воспринимает компьютер. Программисту рекомендуется писать развернутые комментарии к своей программе. Это поможет при отладке и доработке ее текста. Но сейчас для краткости вы можете их не печатать, особенно если вы не вполне освоились с клавиатурой вашего РС. Вам понадобится использовать русский шрифт. На вашем компьютере должен быть установлен русификатор, с помощью которого вы переключите клавиатуру на русский шрифт.

```

Итак:
{раздел описаний}
Var
a,s,p:real;
{главная программа}
begin
writeln('Введите длину стороны квадрата'); {Печать приглашения}
readln(a); {Ввод стороны квадрата}

```

```

s:=a*a; {Вычисление площади}
p:=4*a; {Вычисление периметра}
writeln ('Площадь квадрата =',s:7:3); {Вывод значения площади}
writeln ('Периметр квадрата =',s:7:3); {Вывод значения периметра}
readln; {Остановка до нажатия}
end. {точка после обязательна в конце программы}

```

Обращаем ваше внимание на то, что операторы, расположенные между begin и end сдвинуты вправо. Это называется структурированием программы. В вашей сложной программе будет много операторных скобок begin – end. Найти эту ошибку в неструктурированной программе непросто. Поэтому советуем с самого начала привыкать писать программу структурно.

Оператор writeln ('Введите длину стороны квадрата'); выводит на экран сообщение, которое необходимо взять в апострофы. На экране появляется соответствующая надпись. Оператор readln (a) — это оператор считывания. После того, как вы напечатаете на экране значение стороны квадрата и нажмете клавишу Enter он считает это значение и поместит его в память компьютера, отведенную под переменную a.

Далее следуют вычисления. Над вещественными числами в языке Pascal определены четыре арифметических действия

```

+ - сложение;
- - вычитание;
* - умножение;
/ - деление.

```

Оператор:= называется оператором присваивания. Он вычисляет (в поле операций) значение выражения, представленного в правой части оператора, и записывает его в память, отведенную под переменную, указанную в левой части. Таким образом, переменные s и p принимают значения соответственно площади и периметра квадрата.

Далее: знакомый нам уже оператор writeln выводит на экран надпись и значение переменной s. Переменная отделяется от надписи запятой. Выражение s:7:3 обозначает, что для вывода значения s отведено 7 знаков, в том числе 3 «после точки».т.е. под дробную часть числа. Это называется «форматом числа». Приведенный формат не единственный, могут быть и другие варианты. Если этого не сделать, компьютер выдает ответ в неудобной для пользователя экспоненциальной форме (когда программа у вас заработает, попробуйте написать вместо s:7:3 просто s.Посмотрите, что у вас получится).

Вывод на экран значения периметра делается таким же образом. Далее, мы видим в программе оператор readln. Если этот оператор не имеет скобок, то он выполняет следу-

ющую функцию: останавливает программу до нажатия клавиши Enter. Если этого не сделать, то после выполнения программы пользовательское окно закроется, и мы увидим перед собой окно отладчика-редактора с текстом программы. Результаты вычислений можно будет увидеть, нажав комбинацию клавиш Alt-F5. Это, однако, неудобно. Оператор `readln` позволяет нам видеть результаты вычислений до тех пор, пока мы не нажмем Enter.

Теперь, когда текст программы, готов нам необходимо его сохранить. Для этого нажмите клавишу F2. Появится диалоговое окно SaveAs (сохранить как). Введите в него имя. Например, `kvadrat`. Расширение `pas` будет автоматически присвоено файлу (помеченной области на диске компьютера), в котором сохранится текст. В верхней части окна редактора появится имя файла: `kvadrat.pas`. Отметим, что при повторном нажатии клавиши F2 файл сохранится под тем же именем. Советуем чаще сохранять файл после внесения в него изменений. Предыдущая версия файла также будет сохранена в файле с расширением `bak`. Чтобы сохранить файл под новым именем, необходимо войти в меню File, и выбрать пункт SaveAs.

Теперь, после того, как программа сохранена, необходимо провести ее компиляцию. Нажмите для этого клавишу F9 или Alt+F9. Специальная программа – компилятор – переработает ваши команды в коды, понятные компьютеру и сохранит их в исполняемом файле с расширением `exe`. Если в тексте программы есть ошибки, компиляция не будет завершена и вам будет указано на место в программе, где находится ошибка. Курсор остановится под строкой, в которой находится ошибка, и на экране появится красная полоса, в которой будет указан характер ошибки. После исправления ошибки повторите компиляцию если в программе есть еще ошибки, компилятор найдет их. В противном случае перед вами появится диалоговое окно, в котором будет написано «Compilesuccessful. Pressanykey». Нажмите после этого любую клавишу.

Наиболее часто встречающаяся у начинающих ошибка – «Unknownidentifier» (неизвестный идентификатор). Как правило, она возникает по двум причинам: либо переменная не указана в списке переменных после слова `var`, либо просто неправильно написан какой-нибудь идентификатор. Но могут быть и другие ошибки. Если вам непонятен смысл сообщения в красной полосе, нажмите клавишу F1. Вы получите развернутую подсказку. Умение исправлять свои ошибки пригодится вам в программировании, и в жизни!

Теперь можно запускать программу и продолжать ее отладку. Компилятор находит ошибки только в программировании, ошибки в алгоритме программист должен найти сам. Например, если вы вместо знака умножения поставите знак сложения, то сообщения об ошибке не появится, но программа будет работать неправильно. Для ее проверки запускают какой-нибудь тест, результат которого вам известен заранее. В данном случае се,

конечно, очень просто. Примите сторону квадрата равной, скажем, двум. Если в результате вычислений вы получите 4 и 8, значит, ваша программа работает правильно.

Итак, запускаем программу. Для этого нажмите комбинацию клавиш Ctrl+F9. Появится приглашение. Наберите на клавиатуре значение стороны квадрата. Нажмите Enter. Посмотрите на результаты вычислений. Нажмите опять Enter и вернитесь в окно редактора к тексту программы.

Если все у вас получилось, примите мои поздравления: вы написали и отладили первую в своей жизни программу.

Теперь нужно выйти из среды TurboPascal. Для этого нужно либо выбрать в File пункт Exit, либо нажать комбинацию клавиш Alt+X. Возможно при этом появится предложение сохранить программу. Выполните соответствующие действия по сохранению.

Когда выйдете из TurboPascal, откройте папку, в которой вы сохраняете свои файлы. Там вы должны найти файлы kvadrat.pas и kvadrat.exe. щелкните мышью по exe файлу. При этом заработает ваша программа вычисления площади и периметра квадрата. В дальнейшем вы сможете использовать исполняемый файл своей игры «отдельно» от системы TurboPascal.

Итак, мы завершили создание нашей первой программы. Напишем теперь вторую: вычисление площади и периметра прямоугольника. Основная наша задача при этом, научиться использовать свои собственные наработки. Для этого нужно уметь модифицировать текст программы и сохранять его под другим именем. Общий совет: старайтесь писать, как можно меньше. Используйте переименование, копирование, перенос текста из окна в окно. Всем этим приемам мы вас постепенно научим.

Приступаем. Войдите в TurboPascal. Откройте созданный вами файл kvadrat.pas. Для этого нажмите клавишу F3 или выберите в меню File пункт Open. Появится диалоговое окно, в котором вы увидите список файлов с расширением pas. Войти в этот список можно с помощью клавиши Tab. По списку перемещаемся стрелками, выбрав нужный файл нажимаем Enter.

После открытия файла, начинаем его модифицировать. Для одной из сторон прямоугольника мы по-прежнему будем использовать переменную a. Но нам понадобится еще одна переменная для другой стороны прямоугольника. Назовем ее b и включим в список вещественных переменных. Оператор чтения переменных приобретет вид: readln (a,b). При этом при запуске программы значение переменных можно вводить двумя способами: либо набирать их по очереди, каждый раз нажимая Enter, либо набрать их через пробел, а потом нажать один раз Enter.

Изменяются также формулы для вычисления периметра и площади, и в тексте приглашения и вывода результатов вместо слова «квадрат» появится слово «прямоугольник». Когда вы проделаете все эти операции, получится следующий текст (комментарии для краткости не приводятся):

```

Var
a,b,s,p:real;
begin
writeln( 'Введите длину сторон прямоугольника' );
readln(a,b);
s:=a*b;
p:=2*(a+b);
writeln( 'Площадь прямоугольника=',s:7:3);
writeln( 'Периметр прямоугольника=',p:7:3);
readln;
end.

```

Выберите теперь в меню File пункт SaveAs. Сохраните файл под именем gram.pas. Вот и все. Теперь откомпилируйте программу, исправьте ошибки и приступайте к запуску и отладке.

Задания:

1..Вычислить

$$24/(3*4)-24/3/4+24/3*4;$$

2.Записать на Паскале арифметические выражения:

$$\frac{5y^6}{\sqrt{x+\cos(y-3x^2)}} ; \frac{|\cos(x-5y^2)|}{\sqrt{6 \ln(y)}}$$

3. Можно ли утверждать, что значение выражения $(1/6)^6-1$ равно нулю?

4. Почему параметрами процедуры ввода read могут быть только переменные, а не числа или выражения, скажем, вида X+1?

5.Если в качестве исходных данных задано пять чисел, то можно ли ввести пятое из них, не введя первые четыре?

9 класс

Кейс №1 «Виды моделей»

Начиная с древнейших времен, становление человеческой цивилизации неразрывно связано с моделированием, то есть с построением, изучением и использованием моделей различных объектов, процессов и явлений.

В своей деятельности – в практической сфере, художественной, научной - человек всегда создает некий слепок, заменитель того объекта, процесса или явления с которым ему приходится иметь дело:

- это может быть натурная копия – картина или скульптура;
- это может быть модель самолета (например для изучения его аэродинамических характеристик);
- это может быть макет какого-либо изделия, по которому в дальнейшем будет изготавливаться оригинал;
- математическая формула, описывающая некий процесс (например, закона тяготения).

Таким образом, мы с детства сталкиваемся с понятием "модель". Модель дает нам образ реального объекта или явления, то есть модель является представлением объекта в некоторой форме, отличной от формы его реального существования. Модель – это мощное орудие познания. К созданию моделей прибегают, когда исследуемый объект либо очень велик (модель солнечной системы), либо очень мал (модель атома), когда процесс пробегает очень быстро (модель двигателя внутреннего сгорания) или очень медленно (геологические модели), исследование объекта может привести к его разрушению (модель самолета) или создание модели очень дорого (архитектурный макет города) и т. д.

Каждый объект имеет большое количество различных свойств. В процессе построения модели выделяются главные, наиболее существенные, свойства, те которые интересуют исследователя. В этом главная особенность и главное назначение моделей.

Задачи:

1. Изучить предложенный в кейсе материал и материал учебника
2. Составить схему или ментальную карту – виды моделей.
3. Для каждого вида моделей привести примеры.
4. Подготовить выступление для класса по теме: **«Виды моделей»**

Кейс № 2 «Последовательный поиск в массиве»

Цена земельного участка зависит от площади, наличия газопровода и электросетей. Однако на стоимость влияет и тот факт, что на отдельных участках требуется отсыпка территории или снос старой хозяйственной постройки.

В соответствии со статьей 38 Жилищного кодекса РФ норма жилой площади равно 12 квадратным метрам в расчете на одного человека.

Площадь земельного участка оценивается в сотках. 1 сотка=100 м².

Земли населенных пунктов могут иметь виды разрешенного использования:

- индивидуальное жилищное строительство (ИЖС);
- ведение личного подсобного хозяйства (ЛПХ);
- дачное строительство (ДНП).

Земли сельскохозяйственного назначения могут иметь виды разрешенного использования:

- ведение личного подсобного хозяйства (ЛПХ);
- дачное строительство (ДНП);
- садоводство (СНТ, СНП, СОТ).

На сайте yakutia.info представлен обзор цен на дачные и земельные участки вокруг г. Якутска по данным «Правового центра недвижимости».

Адрес	Назначение	Площадь земельного участка (м ²)	Цена в рублях за 1 сотку	Газ, электричество	Расстояние от центра г. Якутска (км)
Шестаковка проезд	ИЖС	800	68750	эл-во, газ рядом	9
Покровский тракт, 9 км	СОТ	800	150000	подведены	10
Покровский тракт, 8 км	ИЖС	800	137500	рядом	9
Покровский тракт, 15 км	сельхоз	800	31250	нет	16
Сергеляхское шоссе, 10 км	ИЖС	800	125000	эл-во, г/провод у забора	9
Сергеляхское шоссе, 11 км	СОТ	800	200000	проведены	6
Сергеляхское шоссе, 8 км	СОТ	800	200000	проведены	7
Намцырский тракт	СОТ	800	87500	рядом	17

Намский тракт, 1 км	ИЖС	850	81250	эл-во рядом, стояк выведен	10
Намский тракт, 13 км	СОТ	800	68750	эл-во, газ от 30 м	11
Хатынг-Юряхское шоссе	ИЖС	800	100000	рядом	4
Окружное шоссе	СОТ	700	187500	рядом	3
Окружное шоссе	ИЖС	800	250000	проведены	3
Маганский тракт, 2 км	ИЖС	800	118750	рядом	8
Маганский тракт, 2 км	ИЖС	800	131250	газ рядом	8

Задания кейса.

Семья Ивановых из четырех человек решили купить земельный участок под индивидуальное жилищное строительство, расположенный недалеко от города Якутска, с проведенным электричеством.

Задание 1. Рассчитайте стоимость участка, который расположен в самом близком расстоянии от города Якутска.

Задание 2. Рассчитайте самый дешевый по цене участок под ИЖС.

Задание 3. Рассчитайте самый дешевый участок, с проведенным электричеством.

Кейс № 3 «Создание диаграмм в табличном процессоре MS Excel»

Вашим заданием было в течение одной недели измерять и записать температуру воздуха.

Итак, вам нужно выступить в роли метеорологов и, используя табличный процессор, создать таблицу с температурными режимами, которые вы фиксировали. На основе данных таблицы построить диаграмму, определить самый тёплый день, самый холодный день и среднюю температуру недели.

Вопрос кейса: Какие функции используют для нахождения максимального или минимального значения? Каким способом можно создать диаграмму к данным числовым значениям? Будет ли каждая диаграмма отображать температурный режим воздуха?

Кейс № 4 «Поиск информации в глобальной сети Интернет»

Петя Иванов живет в городе Чебоксары. Он решил завести аквариум, и его интересует любая информация по данной теме. Петя захотел узнать все об аквариумах, в том числе, где их можно купить в его городе и сколько они стоят. На первый взгляд, самое простое — это поиск по слову «аквариум». Такой вариант и выбрал Петя — он задал ключевое слово «аквариум» в поисковой системе «Яндекс». Результатом поиска явилось огромное количество страниц (ссылок). Причем среди них оказались сайты, упоминающие группу Бориса Гребенщикова «Аквариум», торговые центры и неформальные объединения с таким же названием, и многое другое, не имеющее отношения ни к аквариумам, ни к аквариумным рыбкам. Нетрудно догадаться, что такой поиск не может удовлетворить даже неспешного пользователя. Слишком много времени придется потратить на то, чтобы отобрать среди всех предложенных документов те, которые касаются нужного предмета, и уж тем более на то, чтобы ознакомиться с их содержанием.

Вопросы и задания кейса:

- 1) Проанализируйте описанную ситуацию.
- 2) Что влияет на поиск информации?
- 3) Что нужно сделать Пете, чтобы решить данную проблему, учитывая его пожелания?
- 4) Как задать запрос поиска, чтобы быстро найти информацию? (Ответ запишите в любой поисковой системе.)
- 5) В чем была ошибка Пети?

Кейс № 5 «Электронная почта»

Фирма «Уют», которая специализируется на изготовлении изделий, делающих жилище уютным, комфортным, открывает свой новый магазин. На открытие магазина нужно пригласить более 100 гостей, среди которых есть как частные лица, так и другие фирмы-друзья. Приглашение нужно послать за короткое время (1 рабочий день). Секретарь фирмы «Уют» смогла за рабочий день подготовить и отправить только половину приглашений.

Вопрос кейса: Почему, по вашему мнению, секретарь фирмы не смогла выполнить поставленную перед ней задачу? Какие способы решения проблемы вы можете предложить?

Приложение 3

Описание форм организации совместной деятельности учащихся на уроке

Урок-игра. Пример: посещение виртуального музея вычислительной техники, тематическое повторение или в игровой форме расширение кругозора детей по предмету «Знатоки информатики».

Учебное задание-проект. Выполнение совместных проектов в группах; подготовка текстовых файлов; создание документов с изображением, таблицами и другими графическими объектами; обсуждение правок в документах с другими соавторами на основе облачных вычислений, применяемыми в образовательном процессе, являются Google Apps Education Edition.

Пример:

Задание: Составьте сравнительную характеристику мобильных операционных систем IOS и Android.

Учащиеся самостоятельно делятся на группы по 3 человека.

Сбор информации происходит из различных источников, которые указаны в файле «Список литературы».

Учащиеся составляют сравнительную характеристику мобильных операционных систем по плану в виде таблицы.

Выполненные работы, учащиеся сохраняют в папку общего доступа для проверки и оценивания результатов деятельности участников учебного процесса.

Учащиеся самостоятельно оценивают свою работу и работу участников учебного сотрудничества. Создают файл под именем «Оценки», в который заносят оценки каждого участника.

Применяемые облачные сервисы: GoogleDocs (Документы), Google (Таблицы).

После выполнения задания, учащиеся научатся: объективно оценивать результаты собственной деятельности; объективно и аргументировано оценивать результаты деятельности участников учебного сотрудничества; адекватно относиться к оценке участников учебного сотрудничества.

«Занимательная информатика» - ребусы, кроссворды, чайнворды, загадки, шарады... – это игры, которые, несмотря на свою простоту и массовое использование, кажутся наименее интересными некоторым взрослым, но очень нравятся своей доступностью и легкостью всем детям, их не боятся ученики любого уровня и темпа работы и с удовольствием выполняют задания.

Найди ошибку. Детям выдаются карточки со словами, изученными по какой-либо теме, в них допущены ошибки. Нужно их найти, исправить и объяснить.

Прав ли Знайка. Доказать, что высказывание верно. Учащиеся обобщают полученные знания.

Игра «Продолжи». Основана на выполнении заданий разного рода группой «по цепочке». Учащиеся обобщают и систематизируют изученный материал.

Охота за сокровищами. Учитель составляет вопросы. Вопросы могут требовать как знаний фактов, так и осмысления или понимания. Учащийся или группа должны ответить на вопросы, используя ресурсы интернета, дополнительную литературу, учебник.

Снежный ком. Работа в группе, которая начинается с решения индивидуального задания. Все учащиеся получают аналогичные задания и самостоятельно выполняют их. После этого следует работа в парах. В парах учащиеся предлагают свои способы решения данного задания, из которых выбирается лучшее. Далее две пары объединяются, и работа продолжается в группе из четырех человек, где снова происходит обсуждение решений и выбирается лучшее из них. В конце работы все учащиеся попадают в одну группу. На этом последнем этапе уже не происходит обсуждения решений, группы делают доклады о своей работе.

Пазлы. Учитель делит тему на несколько частей так, чтобы каждая группа получила бы свою часть темы. Также все группы получают список необходимых источников или сами учебные материалы, с помощью которых они изучают основы предложенной части темы. После изучения материала или выполнения задания группы реформируются так, чтобы в каждую новую группу попали по 1 человеку от каждой прежней группы.

Каждый член новой группы объясняет своим новым коллегам свою часть темы, основы которой он изучил в составе предыдущей группы и отвечает на заданные вопросы.

В заключение работы делают выводы.

Взаимоопрос. Один из способов работы в парах. Используется на стадии «осмысления». Технология применения: Два ученика читают текст, останавливаясь после каждого абзаца, и задают друг другу вопросы разного уровня по содержанию прочитанного. Данная форма способствует развитию коммуникативных навыков.

Ключевые термины. Учитель выбирает из текста 4-5 ключевых слов и выписывает их на доску.

Вариант «а». Парам отводится 5 минут на то, чтобы методом мозговой атаки дать общую трактовку этих терминов и предположить, как они будут фигурировать в последующем тексте.

Вариант «б». Учащимся предлагается в группе или индивидуально составить и записать свою версию рассказа, употребив все предложенные ключевые термины.

При знакомстве с исходным содержанием, учащиеся сопоставляют «свою» версию и версию «оригинального текста». Описанное задание обычно используется на стадии «вызова» при обобщении имеющихся у ученика знаний, однако на стадии «рефлексии», обобщения и осмысления полученной информации, целесообразно вернуться к ключевым терминам и обсудить обнаруженные совпадения и выявленные разногласия. Использование данной формы развивает воображение, фантазию, способствует активизации внимания при знакомстве с текстом оригинала.

Прием «Зигзаг». Или метод пилы. Учащиеся организуются в группы по 4-5 человек для работы над учебным материалом, который разбит на фрагменты.

Затем ребята, изучающие один и тот же вопрос, но состоящие в разных группах, встречаются и обмениваются информацией как эксперты по данному вопросу. Это называется «встречей экспертов». Затем они возвращаются в свои группы и обучают всему новому, что узнали сами, других членов группы. Те, в свою очередь, докладывают о своей части задания (как зубцы одной пилы).