

МИНИСТЕРСТВО ПРОСВЕЩЕНИЯ РОССИЙСКОЙ ФЕДЕРАЦИИ

Министерство образования и молодежной политики Свердловской области

Департамент образования администрации города Екатеринбурга

МАОУ Лицей № 110

РАССМОТРЕНО

Педагогическим советом
МАОУ лицея № 110
им. Л.К. Гришиной
протокол № 1 от 26.08.2024

УТВЕРЖДЕНО

Приказом №_121-К от 30.08.24
МАОУ лицей №110
им. Л. К. Гришиной
директор _____ И. И. Сметанин

РАБОЧАЯ ПРОГРАММА КУРСА ВНЕУРОЧНОЙ ДЕЯТЕЛЬНОСТИ

«Алгоритмизация и программирование»

для основного общего образования

Срок освоения программы 2 года (8-9 класс)

Место курса в образовательном процессе

Изменение взглядов на предмет информатики как науки, её место в системе научного знания требует существенных изменений в содержании образования по информатике. В связи с этим особую актуальность приобретают раскрытие личностных резервов учащихся, формирование информационной культуры школьника и создание соответствующей среды.

Никакая система задач, какой бы хорошей она ни была, никакие тренинги памяти, внимания и т. п. не дают того эффекта, который возникает в случае, если учащиеся осознают необходимость решения тех или иных задач, если у них появляется острая необходимость к преодолению интеллектуальных трудностей, связанных с познанием, если они видят смысл в сотрудничестве с одноклассниками и учителем.

Содержание обучения, представленное в программе элективного курса «Алгоритмизация и программирование», позволяет вести обучение школьников в режиме актуального познания. Практическая направленность курса на создание внешних образовательных продуктов – блок-схем, алгоритмов, исполняемых файлов – способствует выявлению фактов, которые невозможно объяснить на основе имеющихся у школьников знаний. Возникающие при этом познавательные переживания обуславливают сознательное отношение к изучению основных теоретических положений информатики.

Проявления трудолюбия, целеустремленности и одухотворённости, возникающие при воплощении замыслов учащихся в рамках программы «Алгоритмизация и программирование», стимулируют развитие индивидуально-личностных качеств школьников.

Активизация познавательного процесса позволяет учащимся более полно выражать свой творческий потенциал и реализовывать собственные идеи в изучаемой области знаний, создаёт предпосылки по применению освоенных навыков программирования в других учебных курсах, а также способствует возникновению дальнейшей мотивации, направленной на освоение профессий, связанных с разработкой программного обеспечения.

Концепция курса

Ключевой особенностью курса является его направленность на формирование у учащихся навыков поиска собственного решения поставленной задачи, составления алгоритма решения и реализации алгоритма с помощью среда Кумир.

Общепедагогическая направленность занятий – гармонизация индивидуальных и социальных аспектов обучения по отношению к информационным технологиям. Умение составлять алгоритмы решения и навыки программирования на алгоритмическом языке являются элементами информационной компетенции – одной из ключевых компетенций современной школы. Умение находить решение, составлять алгоритм решения и реализовать его с помощью алгоритмических языков - необходимое условие подготовки современных школьников, воспитания их информационной культуры. Особая роль отводится широко представленной в курсе системе рефлексивных заданий. Освоение рефлексии направлено на осознание учащимися того важного обстоятельства, что наряду с разрабатываемыми ими продуктами в виде программ на компьютере рождается основополагающий образовательный продукт: освоенный инструментарий. Именно этот образовательный продукт станет базой для творческого самовыражения учащихся в форме различных программ

Пояснительная записка

Модифицированная программа элективного курса «Алгоритмизация и программирование» составлена на основе программы курса информатики А.Г.Кушниренко, Г.В.Лебедева, Я.Н.Зайдельман, авторской программы А.В. Горячева, в соответствии с примерной программой элективных курсов по информатике и информационным технологиям, рекомендованной Департаментом образовательных программ и стандартов общего образования Министерства образования Российской Федерации, от 2010 года.

Программа «Алгоритмизация и программирование» предназначена для изучения алгоритмизации и программирования учащимися 8 и 9 классов общеобразовательной школы, для изучения курса отводится 1 час в неделю, 34 часа в год, 68 за два года. Его содержание соответствует общему уровню развития и подготовки учащихся данного возраста.

Тип программы: составительская.

Направленность программы: научно-техническая.

Формы организации учебных занятий

Организация учебного процесса с использованием учебно-методического комплекта предусматривает наличие следующих взаимосвязанных и взаимодействующих форм:

урочная форма, когда учитель во время урока объясняет новый материал и консультирует учащихся в процессе выполнения ими практических заданий на компьютере;

внеурочная форма, когда учащийся вне уроков самостоятельно выполняет на компьютере практические задания.

индивидуальные консультации.

Состав учебно-методического комплекта

Учебно-методический комплект по курсу включает учебные методические разработки уроков, содержащее необходимый теоретический материал, раздаточный материал, в который входят алгоритмы выполнения практических работ.

Курс, имея собственную доминантную направленность, предполагает интеграцию с другими учебными предметами. Информационная составляющая этих предметов может использоваться школьниками в процессе разработки алгоритмов.

Актуальность обучения программированию состоит в следующем:

в связи с введением нового федерального государственного стандарта нам необходимо воспитать новое поколение, отвечающего по своему уровню развития и образу жизни условиям информационного общества, обладающего информационной культурой;

необходимость введения курса обусловлена интересом обучающихся, а так же образовательными запросами родителей, несоответствием действующих учебных планов и требований ФГОС, по подготовке обучающихся к ГИА и ЕГЭ.

Курс поддержан программным обеспечением КуМир (Комплект Учебных МИРов). Особенности системы КуМир:

в системе КуМир используется школьный алгоритмический язык с русской лексикой и встроенными исполнителями Робот и Чертёжник;

при вводе программы КуМир осуществляет постоянный полный контроль ее правильности, сообщая на полях программы об всех обнаруженных ошибках;

при выполнении программы в пошаговом режиме КуМир выводит на поля результаты операций присваивания и значения логических выражений. Это позволяет ускорить процесс освоения азов программирования;

КуМир работает в операционных системах Windows или Linux.

Основные цели

создание условий для формирования и развития у обучающихся интереса к изучению информатики и информационных технологий;

развитие алгоритмического мышления учащихся;

расширение спектра посильных учащимся задач из различных областей знаний, решаемых с помощью формального исполнителя;

ознакомление со спецификой профессии программиста.

В ходе ее достижения решаются задачи

Обучающие:

освоение первоначальных навыков в работе на компьютере с использованием интегрированной графической среды «Исполнители»;

обучение основам алгоритмизации и программирования, приобщении к проектно-творческой деятельности.

Воспитательные:

- воспитание целеустремленности и результативности в процессе решения учебных задач.

Развивающие:

способствование формированию информационной культуры школьника;

способствование формированию представления о роли компьютерного программирования в развитии общества;

способствование развитию логического мышления и памяти ребенка;

способствование развитию навыков проектно-творческой деятельности.

Основные принципы

Основными принципами, заложенными в программу, являются следующие:

1. Индивидуальное обучение

Одним из важнейших элементов дополнительного образования является способствование формированию информационной культуры школьника, возможность овладевать знаниями с индивидуальной скоростью и в индивидуальном объёме, что предполагает отдельную работу с каждым учащимся. Поэтому занятия делятся на лекционные (лекционно-практические), на которых тема изучается всей группой, и индивидуальные, на которых и осваивается основная часть тем. Для физической и моральной разгрузки детей, а также в качестве поощрения проводятся игровые занятия.

2. Обучение в активной деятельности

Все темы программы воспитанники изучают на практике, решая большое количество задач по каждой теме.

3. Преемственность

Программа обучения построена так, что каждая новая тема логически связана с предыдущей, то есть при изучении новой темы используются все знания и навыки, полученные на предыдущих этапах обучения. В результате, к концу учебного года подростки не только не забывают всё, что проходили в начале, но даже, наоборот, помнят и понимают программу первых занятий лучше, чем прежде. Такой принцип способствует не только успешному освоению программы, но и позволяет учащимся понять важность уже изученного материала, значимость каждого отдельного занятия. Основной целью обучения является не освоение определенного языка программирования, а закладывание основ для дальнейшего изучения компьютерных языков. Знания, полученные учащимися, помогут им при изучении любого языка программирования.

Весь курс построен на основе дифференцированного подхода и модульного принципа. Каждый обучающийся может выбрать стратегию своего обучения, т. е. создать алгоритм аналогичный разобранному, либо сделать дополнительные задания, вносящие усовершенствования в итоговый программный продукт. Каждый модуль содержит теоретический блок и практические задания с указаниями учителя. Модули представляют собой цепочку постепенно усложняющихся задач для решения, которых учащимся требуется освоить все новые и новые приемы алгоритмизации. Все этапы алгоритма тщательно разбираются учителем совместно с

детьми. Изучение каждого модуля завершается разработкой полностью законченного алгоритма.

Система оценки достижений обучающихся

Для проверки знаний и умений учащихся осуществляется как текущий, так и итоговый контроль. Текущий контроль уровня усвоения материала осуществляется по результатам выполнения учащимися практикума по каждой теме курса. Итоговый контроль реализуется в форме итогового практикума.

Предметом диагностики и контроля являются внешние образовательные продукты учеников (созданные программы и проекты.), а также их внутренние личностные качества (освоенные способы деятельности, знания, умения), которые относятся к целям и задачам курса.

Основой для оценивания деятельности учеников являются результаты анализа его продукции и деятельности по ее созданию. Оценка имеет различные способы выражения – устные суждения педагога, письменные качественные характеристики, систематизированные по заданным параметрам аналитические данные, в том числе и рейтинги.

Ученик выступает полноправным субъектом оценивания. Одна из задач педагога – обучение детей навыкам самооценки. С этой целью учитель выделяет и поясняет критерии оценки, учит детей формулировать эти критерии в зависимости от поставленных целей и особенностей образовательного продукта – программы и творческого проекта. Проверка достигаемых учениками образовательных результатов производится в следующих формах:

1. Текущий рефлексивный самоанализ, контроль и самооценка учащимися выполняемых заданий.
2. Взаимооценка учащимися работ друг друга или работ, выполненных в группах.
3. Публичная защита выполненных учащимися творческих работ (индивидуальных и групповых).
4. Текущая диагностика и оценка учителем деятельности школьников.
5. Итоговая оценка деятельности и образовательной продукции ученика в соответствии с его индивидуальной образовательной программой по курсу.

6. Итоговая оценка индивидуальной деятельности учащихся учителем, выполняемая в форме образовательной характеристики.

Предметом контроля и оценки являются внешние образовательные продукты учеников. Качество ученической программы оценивается следующими критериями:

последовательность действий при разработке программ: постановка задачи, выбор метода решения, составление алгоритма, составление программы, запись программы в компьютер, отладка программы, тестирование программы;

«Правила хорошего тона» при разработке программ: читаемость и корректность программ, защита от неправильного ввода, понятия хорошего и плохого «стиля программирования».

Выполненные учащимися работы включаются в их «портфель достижений». Уровень развития у учащихся личностных качеств определяется на основе сравнения результатов их диагностики в начале и конце курса. С помощью методики, включающей наблюдение, тестирование, анализ образовательной продукции учеников, учитель оценивает уровень развития личностных качеств учеников по параметрам, сгруппированным в определенные блоки: технические качества, коммуникативные, когнитивные, рефлексивные. Итоговый контроль проводится в конце всего курса. Он может иметь форму зачета олимпиады или защиты творческих работ. Данный тип контроля предполагает комплексную проверку образовательных результатов по всем заявленным целям и направлениям курса. Формой итоговой оценки каждого ученика выступает образовательная характеристика, в которой указывается уровень освоения им каждой из целей курса и каждого из направлений индивидуальной программы ученика по курсу. Оценке подлежит в первую очередь уровень достижения учеником минимально необходимых результатов, обозначенных в целях и задачах курса. Оцениванию подлежат также те направления и результаты деятельности учеников, которые определены в рабочей программе учителя и в индивидуальных образовательных программах учеников.

Планируемые результаты изучения курса АиП

Личностные, метапредметные и предметные результаты освоения предмета

Личностные результаты

1. Готовность и способность обучающихся к саморазвитию и самообразованию на основе мотивации к обучению и познанию; готовность и способность осознанному выбору и построению дальнейшей индивидуальной траектории образования на базе ориентировки в мире профессий и профессиональных предпочтений, с учетом устойчивых познавательных интересов.

2. Сформированность целостного мировоззрения, соответствующего современному уровню развития науки и общественной практики, учитывающего социальное, культурное, языковое, духовное многообразие современного мира.

Требования к результатам образования в соответствии с рабочей программой воспитания

В воспитании обучающихся подросткового возраста приоритетом является создание благоприятных условий для развития социально значимых отношений обучающихся, и, прежде всего, ценностных отношений:

к труду как основному способу достижения жизненного благополучия человека, залогом его успешного профессионального самоопределения и ощущения уверенности в завтрашнем дне;

к своему отечеству, своей малой и большой Родине как месту, в котором человек вырос и познал первые радости и неудачи, которая завещана ему предками и которую нужно оберегать;

к природе как источнику жизни на Земле, основе самого ее существования, нуждающейся в защите и постоянном внимании со стороны человека;

к знаниям как интеллектуальному ресурсу, обеспечивающему будущее человека, как результату кропотливого, но увлекательного учебного труда;

□ к здоровью как залогом долгой и активной жизни человека, его хорошего настроения и оптимистичного взгляда на мир;

□ к окружающим людям как безусловной и абсолютной ценности,

как равноправным социальным партнерам, с которыми необходимо выстраивать доброжелательные и взаимоподдерживающие отношения, дающие человеку радость общения и позволяющие избегать чувства одиночества;

□ к самим себе как хозяевам своей судьбы, самоопределяющимся

и самореализующимся личностям, отвечающим за свое собственное будущее.

Метапредметные результаты

Регулятивные УУД

1. Умение самостоятельно определять цели обучения, ставить и формулировать новые задачи в учебе и познавательной деятельности, развивать мотивы и интересы своей познавательной деятельности. Обучающийся сможет:

- анализировать существующие и планировать будущие образовательные результаты;
- идентифицировать собственные проблемы и определять главную проблему;
- выдвигать версии решения проблемы, формулировать гипотезы, предвосхищать конечный результат;
- ставить цель деятельности на основе определенной проблемы и существующих возможностей;

2. Умение самостоятельно планировать пути достижения целей, в том числе альтернативные, осознанно выбирать наиболее эффективные способы решения учебных и познавательных задач. Обучающийся сможет:

- определять необходимые действие(я) в соответствии с учебной и познавательной задачей и составлять алгоритм их выполнения;
- обосновывать и осуществлять выбор наиболее эффективных способов решения учебных и познавательных задач;

- определять/находить, в том числе из предложенных вариантов, условия для выполнения учебной и познавательной задачи;
- выбирать из предложенных вариантов и самостоятельно искать средства/ресурсы для решения задачи/достижения цели;
- определять потенциальные затруднения при решении учебной и познавательной задачи и находить средства для их устранения;

3. Умение соотносить свои действия с планируемыми результатами, осуществлять контроль своей деятельности в процессе достижения результата, определять способы действий в рамках предложенных условий и требований, корректировать свои действия в соответствии с изменяющейся ситуацией. Обучающийся сможет:

- оценивать свою деятельность, аргументируя причины достижения или отсутствия планируемого результата;
- находить достаточные средства для выполнения учебных действий в изменяющейся ситуации и/или при отсутствии планируемого результата;
- работая по своему плану, вносить коррективы в текущую деятельность на основе анализа изменений ситуации для получения запланированных характеристик продукта/результата;
- устанавливать связь между полученными характеристиками продукта и характеристиками процесса деятельности и по завершении деятельности предлагать изменение характеристик процесса для получения улучшенных характеристик продукта;
- сверять свои действия с целью и, при необходимости, исправлять ошибки самостоятельно.

4. Умение оценивать правильность выполнения учебной задачи, собственные возможности ее решения. Обучающийся сможет:

- определять критерии правильности (корректности) выполнения учебной задачи;
- анализировать и обосновывать применение соответствующего инструментария для выполнения учебной задачи;
- оценивать продукт своей деятельности по заданным и/или самостоятельно определенным критериям в соответствии с целью деятельности;

5. Владение основами самоконтроля, самооценки, принятия решений и осуществления осознанного выбора в учебной и познавательной. Обучающийся сможет:

- наблюдать и анализировать собственную учебную и познавательную деятельность и деятельность других обучающихся в процессе взаимопроверки;
- соотносить реальные и планируемые результаты индивидуальной образовательной деятельности и делать выводы;

Познавательные УУД

6. Умение создавать, применять и преобразовывать знаки и символы, модели и схемы для решения учебных и познавательных задач. Обучающийся сможет:

- переводить сложную по составу (многоаспектную) информацию из графического или формализованного (символьного) представления в текстовое, и наоборот;
- строить схему, алгоритм действия, исправлять или восстанавливать неизвестный ранее алгоритм на основе имеющегося знания об объекте, к которому применяется алгоритм;

Коммуникативные УУД

7. Умение организовывать учебное сотрудничество и совместную деятельность с учителем и сверстниками; работать индивидуально и в группе: находить общее решение и разрешать конфликты на основе согласования позиций и учета интересов; формулировать, аргументировать и отстаивать свое мнение. Обучающийся сможет:

- определять возможные роли в совместной деятельности;
- играть определенную роль в совместной деятельности;
- предлагать альтернативное решение в конфликтной ситуации;
- выделять общую точку зрения в дискуссии;

- договариваться о правилах и вопросах для обсуждения в соответствии с поставленной перед группой задачей;
- организовывать учебное взаимодействие в группе (определять общие цели, распределять роли, договариваться друг с другом и т. д.);

Предметные результаты

Алгоритмы и элементы программирования

- владение навыками алгоритмического мышления и понимание необходимости формального описания алгоритмов;
- овладение понятием сложности алгоритма, знание основных алгоритмов;
- владение стандартными приёмами написания на алгоритмическом языке программы для решения стандартной задачи с использованием основных конструкций программирования и отладки программ;
- владение умением понимать программы, написанные на выбранном для изучения универсальном алгоритмическом языке;
- умением анализировать алгоритмы с использованием таблиц;
- владение навыками и опытом разработки программ в выбранной среде программирования, включая тестирование и отладку программ.

Весь курс построен на основе дифференцированного подхода и модульного принципа. Каждый обучающийся может выбрать стратегию своего обучения, т. е. создать алгоритм аналогичный разобранному, либо сделать дополнительные задания, вносящие усовершенствования в итоговый программный продукт. Каждый модуль содержит теоретический блок и практические задания с указаниями учителя. Модули представляют собой цепочку постепенно усложняющихся задач для решения, которых учащимся требуется освоить все новые и новые приемы алгоритмизации. Все этапы алгоритма тщательно

разбираются учителем совместно с детьми. Изучение каждого модуля завершается разработкой полностью законченного алгоритма.

Выпускник научится:

- составлять алгоритмы для решения учебных задач различных типов;
- выражать алгоритм решения задачи различными способами (словесным, графическим, в том числе и в виде блок-схемы, с помощью формальных языков и др.);
- определять наиболее оптимальный способ выражения алгоритма для решения конкретных задач (словесный, графический, с помощью формальных языков);
- определять результат выполнения заданного алгоритма или его фрагмента;
- использовать термины «исполнитель», «алгоритм», «программа», а также понимать разницу между употреблением этих терминов в обыденной речи и в информатике;
- выполнять без использования компьютера («вручную») несложные алгоритмы управления исполнителями и анализа числовых и текстовых данных, записанные на конкретном языке программирования с использованием основных управляющих конструкций последовательного программирования (линейная программа, ветвление, повторение, вспомогательные алгоритмы);
- составлять несложные алгоритмы управления исполнителями и анализа числовых и текстовых данных с использованием основных управляющих конструкций последовательного программирования и записывать их в виде программ на выбранном языке программирования; выполнять эти программы на компьютере;
- использовать величины (переменные) различных типов, табличные величины (массивы), а также выражения, составленные из этих величин; использовать оператор присваивания;
- анализировать предложенный алгоритм, например, определять какие результаты возможны при заданном множестве исходных значений;
- использовать логические значения, операции и выражения с ними;
- записывать на выбранном языке программирования арифметические и логические выражения и вычислять их значения.

Выпускник получит возможность:

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

Содержание учебного предмета

В содержании предмета «Основы программирования» для 8–9 классов выделено два модуля: Алгоритмизации в среде исполнителя на примере программы Кумир и Основы программирования на языке Паскаль

1. Алгоритмизации в среде исполнителя
 - 1.1. Алгоритм, программа. Освоение среды
 - 1.2. Команды повтора
 - 1.3. Программа и подпрограмма
 - 1.4. Команды ветвления «ЕСЛИ», «ВЫБОР»»
2. Основы программирования

Алгоритмы и элементы программирования

Исполнители и алгоритмы. Управление исполнителями

Исполнители. Состояния, возможные обстановки и система команд исполнителя; команды-приказы и команды-запросы; отказ исполнителя. Необходимость формального описания исполнителя. Ручное управление исполнителем.

Алгоритм как план управления исполнителем (исполнителями). Алгоритмический язык (язык программирования) – формальный язык для записи алгоритмов. Программа – запись алгоритма на конкретном алгоритмическом языке. Компьютер – автоматическое устройство, способное управлять по заранее составленной программе исполнителями, выполняющими команды. Программное управление исполнителем. Программное управление самодвижущимся роботом.

Словесное описание алгоритмов. Описание алгоритма с помощью блок-схем. Отличие словесного описания алгоритма, от описания на формальном алгоритмическом языке.

Системы программирования. Средства создания и выполнения программ.

Понятие об этапах разработки программ и приемах отладки программ.

Управление. Сигнал. Обратная связь. Примеры: компьютер и управляемый им исполнитель (в том числе робот); компьютер, получающий сигналы от цифровых датчиков в ходе наблюдений и экспериментов, и управляющий реальными (в том числе движущимися) устройствами.

Алгоритмические конструкции

Конструкция «следование». Линейный алгоритм. Ограниченность линейных алгоритмов: невозможность предусмотреть зависимость последовательности выполняемых действий от исходных данных.

Конструкция «ветвление». Условный оператор: полная и неполная формы.

Выполнение и невыполнение условия (истинность и ложность высказывания). Простые и составные условия. Запись составных условий.

Конструкция «повторения»: циклы с заданным числом повторений, с условием выполнения, с переменной цикла. Проверка условия выполнения цикла до начала выполнения тела цикла и после выполнения тела цикла: постусловие и предусловие цикла. Инвариант цикла.

Запись алгоритмических конструкций в выбранном языке программирования.

Примеры записи команд ветвления и повторения и других конструкций в различных алгоритмических языках.

Разработка алгоритмов и программ

Оператор присваивания. Представление о структурах данных.

Константы и переменные. Переменная: имя и значение. Типы переменных: целые, вещественные, символьные, строковые, логические. Табличные величины (массивы). Одномерные массивы. Двумерные массивы.

Примеры задач обработки данных:

нахождение минимального и максимального числа из двух, трех, четырех данных чисел;

нахождение всех корней заданного квадратного уравнения;

заполнение числового массива в соответствии с формулой или путем ввода чисел;
нахождение суммы элементов данной конечной числовой последовательности или массива;

нахождение минимального (максимального) элемента массива.

Знакомство с алгоритмами решения этих задач. Реализации этих алгоритмов в выбранной среде программирования.

Составление алгоритмов и программ по управлению исполнителями Робот, Черепашка, Чертежник и др.

Знакомство с постановками более сложных задач обработки данных и алгоритмами их решения: сортировка массива, выполнение поэлементных операций с массивами; обработка целых чисел, представленных записями в десятичной и двоичной системах счисления, нахождение наибольшего общего делителя (алгоритм Евклида).

Понятие об этапах разработки программ: составление требований к программе, выбор алгоритма и его реализация в виде программы на выбранном алгоритмическом языке, отладка программы с помощью выбранной системы программирования, тестирование.

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

Знакомство с документированием программ. Составление описание программы по образцу.

Анализ алгоритмов

Сложность вычисления: количество выполненных операций, размер используемой памяти; их зависимость от размера исходных данных. Примеры коротких программ, выполняющих много шагов по обработке небольшого объема данных; примеры коротких программ, выполняющих обработку большого объема данных.

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

Тематическое планирование

Предмет <i>Алгоритмизация и программирование</i>		Уровень образования (классы) <u>ООО (8-9класс)</u>	
8 класс			
Тема/ Электронный ресурс	Количество часов, отводимое на освоение темы	Темы уроков	Содержание примерной программы
Тема 1. Алгоритм, программа. Освоение среды https://www.kpolyakov.spb.ru/download/robkumppt.zip	7	1. Понятие алгоритма и программы.	<ul style="list-style-type: none"> • понятие алгоритм, программа • свойства алгоритма • пример программы в среде исполнителя Кумир
		2. Переменные. Оператор присваивания.	<ul style="list-style-type: none"> • имя переменной, тип, описание. • оператор присваивания. • блок-схема • решение линейных задач в среде исполнителя Кумир
		3. Команды ввода-вывода.	<ul style="list-style-type: none"> • команды ввода-вывода. • арифметические операции. • порядок операций • нахождение ошибок в программе. • ручная прокрутка • решение линейных задач в среде исполнителя Кумир
		4. Арифметические операции.	
		5. Решение линейных задач в среде исполнителя Кумир.	
		6. Решение линейных задач в среде исполнителя Кумир.	
		7. Итоговая работа.	
Тема 2: Команды ветвления https://www.kpolyakov.spb.ru/download/robkumppt.zip	8	1. Команды ветвления	<ul style="list-style-type: none"> • общий вид команды «если»; • блок - схема • выполнения команды «если»; • краткая и полная форма записи; • решение задач на составление алгоритмов с командой ветвления.
		2. Формы записи команды ветвления	
		3. Решение задач на ветвление в среде исполнителя Кумир	

		4. Сложные условия	<ul style="list-style-type: none"> • сложные условия и логические операции
		5. Сложное условие И	<ul style="list-style-type: none"> • общий вид записи сложного условия "И", • блок-схема, • правило работы; • решение задач с условием в среде исполнителя Кумир
		6. Сложное условие ИЛИ	<ul style="list-style-type: none"> • общий вид записи сложного условия "ИЛИ", • блок-схема, • правило работы; • решение задач с условием в среде исполнителя Кумир
		7. Решение задач с условием в среде исполнителя Кумир	<ul style="list-style-type: none"> • решение задач на ветвление с условием в среде исполнителя Кумир
8. Итоговая работа			
Тема 3: Команды повтора. Циклы. https://www.kpolyakov.spb.ru/download/robkumppt.zip	8	1. Команды повтора.	<ul style="list-style-type: none"> • команда повтора "N раз" - общий вид записи, • блок-схема, • правило работы; • решение задач с повтором в среде исполнителя Кумир
		2. Решение задач с повтором в среде исполнителя Кумир	
		3. Цикл с условием	<ul style="list-style-type: none"> • команда повтора с условием - общий вид записи, • блок-схема, • правило работы • примеры задач
		4. Цикл с условием	
		5. Решение задач с циклами в среде исполнителя Кумир	<ul style="list-style-type: none"> • решение задач с циклами в среде исполнителя Кумир
		6. Решение задач с циклами в среде исполнителя Кумир	
		7. Решение задач с циклами в среде исполнителя Кумир	
		8. Итоговая работа	
Тема 4: Программа и подпрограмма	8	1. Программа и подпрограмма. Основной и вспомогательные алгоритмы.	<ul style="list-style-type: none"> • понятия основного и вспомогательного алгоритмов; • вызов вспомогательного алгоритма;
		2. Программа и подпрограмма. Основной и	

https://www.kpolyakov.spb.ru/download/robkumppt.zip		вспомогательные алгоритмы.	<ul style="list-style-type: none"> • выполнение вспомогательного алгоритма с аргументами
		3. Метод последовательного уточнения	<ul style="list-style-type: none"> • суть метода последовательного уточнения
		4. Решение задач со вспомогательными алгоритмами в среде исполнителя Кумир	<ul style="list-style-type: none"> • решение задач со вспомогательными алгоритмами в среде исполнителя Кумир
		5. Решение задач со вспомогательными алгоритмами в среде исполнителя Кумир	
		6. Итоговая работа	
		7. Фестиваль программ	<ul style="list-style-type: none"> • дифференцированные задания на выбор, творческая работа на составление задач
		8. Фестиваль программ	
	Тема 5. Повторение	3	1. Повторение темы «Обработка информации»
Резерв	1		
Итого	35		

9 класс

Тема	Количество часов, отводимое на освоение темы	Темы уроков	Содержание примерной программы
<p>Тема 1. Среда программирования PascalABC.Net.</p>	<p align="center">1</p>	<p>1. Место Pascal в семействе языков высокого уровня. Редактор кода. Запуск программы и отладка.</p>	<p>История развития языков программирование. Необходимость в высокоуровневых языках. Pascal – язык обучения программированию. Настройка параметров среды. Анализ сообщений компилятора. Отладка программы в пошаговом режиме, отслеживание значений переменных. Запуск программы на выполнение.</p>
<p>Тема 2: Основные конструкции языка Pascal</p> <p>https://www.kpolyakov.spb.ru/download/pasppt1.zip</p>	<p align="center">12</p>	<p>1. Характеристика и особенности языка. Лексическая структура языка. Общая структура программы.</p> <p>2. Величина и ее характеристики. Выражения. Оператор присваивания. Ввод-вывод данных.</p> <p>3. Практикум. Разработка простейших программ. (1 час)</p> <p>4. Оператор ветвления if-then-else.</p> <p>5. Практикум. Разработка программ с ветвлением.</p> <p>6. Оператор выбора case.</p> <p>7. Практикум. Организация выбора в программе.</p>	<p>Состав и алфавит языка. Элементарные конструкции. Комментарии в программе. Основные разделы программы. Назначение и структура блока описаний. Исполняемый блок. Идентификаторы. Правила записи идентификаторов.</p> <p>Данные в программе. Концепция типов. Стандартные типы. Структура выражения и порядок выполнения операций. Структура оператора присваивания. Процедуры Read, Readln, Write, Writeln. Форматирование данных вывода.</p> <p>Разработка и выполнение программ линейной структуры. Ввод-вывод данных.</p> <p>Оператор ветвления в полной и краткой форме. Составной оператор. Отношения. Реализация условий через логические выражения. Логические операции. Разработка и выполнение программ с ветвлением. Задача о зеркале и дверном проеме.</p> <p>Разработка и выполнение программ с ветвлением. Решение квадратного уравнения.</p> <p>Структура оператора Case. Особенности построения переключателя. Использование группы else,</p> <p>Разработка и выполнение программ с множественным ветвлением. Определение времени года по номеру месяца.</p>

		8. Цикл While-do. Решение задач с предусловным циклом.	Структура цикла. Проблема заикливания. Разработка и выполнение программ с предусловным циклом.
		9. Цикл for-to-do.	Цикл с увеличивающимся и уменьшающимся счетчиком. Количество итераций. Нахождение суммы, минимального, максимального на последовательности.
		10. Практикум. Решение задач с арифметическим циклом (1 час).	Решение задач на нахождение суммы, количества, среднего арифметического последовательности.
		11. Практикум. Решение задач с арифметическим циклом (2 час).	Решение задач на нахождение минимального, максимального на последовательности.
		12. Итоговая работа «Операторы языка Pascal» Практикум.	
Тема 3: Модульное программирование https://www.kpolyakov.spb.ru/download/pasppt1.zip	6	1. Программа и подпрограмма. Особенности реализации в Pascal.	Описание подпрограмм в Pascal. Отличия процедур от функций. Модульное программирование.
		2. Практикум. Задачи с процедурами.	Решение задач на выделение цифр из числа.
		3. Параметры подпрограмм.	Формальные и фактические параметры. Локальные и глобальные объекты. Правило видимости.
		4. Практикум. Задачи с функциями.	Делители числа. Решение задач по алгоритму Эвклида.
		5. Практикум. Использование параметров.	Решение задачи сокращения простой дроби.
		6. Итоговая работа «Подпрограммы»	
Тема 4: Составные данные. https://www.kpolyakov.spb.ru/download/pasppt2.zip	12	1. Простые и составные данные. Анализ потока данных.	Необходимость составных данных. Классификация составных данных.
		2. Линейные массивы. Генерация и вывод массива.	Описание массивов. Индексная адресация. Ввод и вывод массива. Генерация данных случайным образом.
		3. Практикум. Отбор данных в массиве по условию.	Решение задач на кратность.
		4. Практикум. Поиск наибольшего, наименьшего на массиве.	Решение задачи о поиске максимума-минимума.
		5. Сортировка массива. Простые сортировки.	Метод пузырька. Сортировка выбором.
		6. Практикум. Сортировка пузырьком.	Реализация сортировки массива методом пузырька.
		7. Практикум. Сортировка выбором.	Реализация сортировки массива методом выбора.
		8. Практикум. Сортировка массива.	Сравнение эффективности методов сортировки (пузырек и выбор)
		9. Практикум. Задача о медиане.	
		10. Поиск второго минимума-максимума.	Подходы к определению второго (и последующих) минимумов-максимумов.
		11. Практикум. Поиск второго минимума-	

		максимума.	
		12. Итоговая работа «Линейные массивы»	
Тема 5. Повторение	2	1. Повторение темы «Обработка информации»	
Резерв	1		
Итого	34		

**ДОКУМЕНТ ПОДПИСАН
ЭЛЕКТРОННОЙ ПОДПИСЬЮ**

СВЕДЕНИЯ О СЕРТИФИКАТЕ ЭП

Сертификат 628517657829497081401818755133691667472907604416

Владелец Сметанин Игорь Иванович

Действителен с 12.08.2024 по 12.08.2025