

**Муниципальное бюджетное общеобразовательное учреждение
ЗАО г. Североморск «Средняя общеобразовательная школа № 7
имени Героя России Марка Евтюхина»**

**Приложение к ООП НОО
МБОУ ЗАО г. Североморск «СОШ №7»**

**Утверждено приказом директора
МБОУ ЗАО г. Североморск «СОШ №7»
от 30.08.2023г. №453**

**Рабочая программа
курса внеурочной деятельности
по информатике
«Основы алгоритмизации и
программирования»**

**среднее общее образование
10-11 класс**

**Принята
на педагогическом совете
протокол №1**

Пояснительная записка

Дополнительная общеразвивающая программа «Основы алгоритмизации и программирования» разработана в соответствии с **нормативно-правовыми актами**:

1. Федеральный закон от 29 декабря 2012 года № 273-ФЗ «Об образовании в Российской Федерации».
2. Постановление Главного государственного санитарного врача Российской Федерации от 28.09.2020 г. № 28 "Об утверждении санитарных правил СП 2.4. 3648-20 "Санитарно-эпидемиологические требования к организациям воспитания и обучения, отдыха и оздоровления детей и молодежи";
3. Постановление Главного государственного санитарного врача Российской Федерации от 28.01.2021 № 2 "Об утверждении санитарных правил и норм СанПиН 1.2.3685-21 "Гигиенические нормативы и требования к обеспечению безопасности и (или) безвредности для человека факторов среды обитания".
4. Приказ Министерства образования и науки Российской Федерации от 9 ноября 2018 г. № 196 (с изменениями на 30 сентября 2020 г.) «Об утверждении порядка организации и осуществления образовательной деятельности по дополнительным общеобразовательным программам».
5. Письмо Минобрнауки России от 11 декабря 2006 г. № 06-1844 «Примерные требования к образовательным программам дополнительного образования детей».
6. Распоряжение Правительства РФ от 25 сентября 2017 г. № 2039-р Об утверждении Стратегии повышения финансовой грамотности в Российской Федерации на 2017 - 2023 гг.

Программа разработана на основе авторской программы Е. М. Гутник, А.В. Перышкина «Физика, 7-9 класс».

Программа «Основы алгоритмизации и программирования» предназначена для изучения алгоритмизации и программирования учащимися 9 классов, для изучения курса отводится 1 час в неделю, 34 часа в год в 9 классе. Его содержание соответствует общему уровню развития и подготовки учащихся данного возраста.

Модифицированная программа 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.

Содержание обучения, представленное в программе «Основы алгоритмизации и программирования», 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 «Основы алгоритмизации и программирования», 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.

Формы организации учебных занятий

Организация учебного процесса с использованием учебно-методического комплекта предусматривает наличие следующих взаимосвязанных и взаимодополняющих 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.
- *дистанционная форма* с использованием on-line технологий;
- *индивидуальные консультации*.

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.

А так же средой программирования PascalABC.Net, реализующей язык программирования Pascal нового поколения, сочетающий простоту классического языка Паскаль, ряд современных расширений и огромные возможности платформы .NET.

Свободная лицензия LGPLv3.

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.

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:

- обучение основам алгоритмизации и программирования, приобщении к проектно-творческой деятельности;
- освоение первоначальных навыков в работе на компьютере с использованием интегрированной графической среды «Исполнители»;
- изучение основ языка Pascal, освоение среды программирования PascalABC.Net.

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

Планируемые результаты

Универсальные учебные действия:

регулятивные:

- учитывать правило в планировании и контроле способа решения;
- осуществлять итоговый и пошаговый контроль по результату;

- оценивать правильность выполнения действий на уровне адекватной ретроспективной оценки;
- вносить необходимые коррективы в действие после его завершения на основе учета характера сделанных ошибок;
- различать способ и результат действия;

познавательные:

- осуществлять поиск необходимой информации для выполнения учебных заданий с использованием учебной литературы;
- строить речевые высказывания в устной и письменной форме;
- проводить сравнение, сериацию и классификацию по заданным критериям;
- владеть общим приемом решения задач;
- ориентироваться на разнообразие способов решения задач;

коммуникативные:

- учитывать разные мнения и стремиться к координации различных позиций в сотрудничестве.

предметные:

- 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 блок-схем;
- 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.
- Использование программных систем и сервисов

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 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.

Содержание программы

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

Алгоритмизации в среде исполнителя

1. Модель, алгоритм, программа. Освоение среды.
2. Команды повтора.
3. Программа и подпрограмма.
4. Команды ветвления «ЕСЛИ», «ВЫБОР».

Основы программирования

1. Среда программирования PascalABC.Net.
2. Основные конструкции языка Pascal.
3. Модульное программирование.
4. Составные данные.

Алгоритмизация в среде исполнителя

Тема	Содержание	Кол-во часов
Тема 1: «Модель, алгоритм, программа. Освоение среды»	Этапы решения задач на ПК. Понятие информационной модели. Простейший пример модели – модель исполнителя. Алгоритм – виды алгоритмов, способы записи алгоритмов, понятие оптимизации алгоритмов. Программа. Ошибки, типы ошибок. Система команд исполнителя. Команды с аргументами. Написание простейших программ в среде исполнителя Робот и в среде исполнителя Чертежник.	3
Тема 2: «Команды повтора»	Команда повтора с условием, команда повтора «N раз», команда повтора с параметром. Общий вид записи, правила работы команд, графическое представление работы команд (блок-схема). Вложенные циклы. Переменные величины: имя, тип, значение.	4
Тема 3: «Программа и подпрограмма»	Основной и вспомогательные алгоритмы. Метод последовательного уточнения. Алгоритмы с аргументами. Арифметические выражения: линейный вид записи, правило записи арифметических выражений. Моделирование диалоговых программ.	5
Тема 4: «Команды ветвления «ЕСЛИ», «ВЫБОР»»	Команды ветвления: «если», «выбор». Общий вид записи, правило работы команд, графическое представление. Сложные условия,	5

Основы программирования

Тема	Содержание	Кол-во часов
Тема 1: Среда программирования PascalABC.Net.	История развития языков программирования. Необходимость в высокоуровневых языках. Место Pascal в семействе языков высокого уровня. Pascal – язык обучения программированию.	1
Тема 2: Основные конструкции	Состав и алфавит языка. Элементарные конструкции. Комментарии в программе. Основные разделы	6

языка Pascal

программы. Назначение и структура блока описаний. Исполняемый блок. Идентификаторы. Правила записи идентификаторов.

Данные в программе. Концепция типов. Стандартные типы.

Структура выражения и порядок выполнения операций. Структура оператора присваивания.

Процедуры Read, Readln, Write, Writeln. Форматирование данных вывода. Разработка и выполнение программ линейной структуры.

Оператор ветвления в полной и краткой форме. Составной оператор. Отношения. Реализация условий через логические выражения. Логические операции.

Разработка и выполнение программ с ветвлением.

Оператор Case. Особенности построения переключателя.

Разработка и выполнение программ с множественным ветвлением.

Структура цикла While-do. Проблема зацикливания. Разработка и выполнение программ с предусловным циклом.

Цикл For-do с увеличивающимся и уменьшающимся счетчиком. Количество итераций. Нахождение суммы, минимального, максимального на последовательности.

Организация контроля данных на вводе.

Тема 3:
Модульное
программирование

Программа и подпрограмма. Особенности реализации подпрограмм в Pascal.

4

Описание подпрограмм в Pascal. Отличия процедур от функций. Параметры подпрограмм. Формальные и фактические параметры. Локальные и глобальные объекты. Правило видимости.

Тема 4:
Составные данные.

Простые и составные данные. Классификация составных данных.

6

Линейные массивы. Описание массивов. Индексная адресация. Ввод и вывод массива. Генерация данных случайным образом.

Отбор данных в массиве по условию. Поиск наибольшего, наименьшего на массиве. Сортировка массива. Простые сортировки. Метод пузырька. Сортировка выбором. Задача о медиане. Поиск второго минимума-максимума.

При реализации программы «Основы алгоритмизации и программирования» у учащихся формируется информационная и алгоритмическая культура; умение формализации и структурирования информации, учащиеся овладевают способами представления данных в соответствии с поставленной задачей, с использованием соответствующих программных средств обработки данных; у учащихся формируется представление о компьютере как универсальном устройстве обработки информации; представление об алгоритмических конструкциях 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.

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.

Список рекомендуемой литературы для обучающихся и родителей

1. Агафонова И.Н. Учимся думать. Сборник занимательных задач, тестов и упражнений. СПб. М.: М – Экспресс, 1996.
2. Семакин И.Г. Информатика 10 класс, М.: БИНОМ. Лаборатория знаний, 2014.
3. Горячев А., Ю. Шафрин. Задачник-практикум по информатике, М.: БИНОМ. Лаборатория знаний, 2004.
4. Михайлова З.А. Игровые занимательные задачи для дошкольников. М.: Просвещение, 1990.
5. Субботина Л.Ю. Развитие воображения у детей. Популярное пособие для детей и педагогов. Ярославль: Академия развития, 1993.
6. Тихомирова Л.Ф. Развитие интеллектуальных способностей школьника. Популярное пособие для родителей и педагогов. Ярославль: Академия развития, 1996.
7. Угринович Н., Л. Босова. Н. Михайлова, Практикум по информационным технологиям (5-11 класс), М.: БИНОМ. Лаборатория знаний, 2004.
8. Кушниренко А. Г., Г.В.Лебедев, Р.А.Сворень "Основы информатики и вычислительной техники", Москва, "Просвещение", 1992 год.
9. Поляков К.Е. Исполнитель "Робот". [Текст] / К.Е. Поляков. – СПб, 2009.
10. Поляков К.Е. Алгоритмы и исполнители. [Текст] / К.Е. Поляков. – СПб, 2007. – 74с.

Список рекомендуемой литературы для учителя

1. Рабочие программы по информатике и ИКТ 5-11 классы, М.: «Глобус», 2009.
2. Босова Л. Л. Подготовка младших школьников в области информатики и ИКТ: опыт, современное состояние и перспективы – М.: БИНОМ. Лаборатория знаний, 2009.
3. Волина В.В. Занимательное азбуковедение. Книга для учителя. М.: Просвещение, 1991.
4. Гильбух Ю.З. Как учиться и работать эффективно. Киев, 1993.
5. Горячев А.В., Волков Т.О., Горина К.И., Лобачева Л.Л., Спиридонова Т.Ю. Информатика в играх и задачах / Под ред. А.В. Горячева. М.: Экспресс, 1996.
6. Житкова О.А., Е.К. Кудрявцева. Тематический контроль по информатике. Основы информатики и вычислительной техники, М.: «Интеллект - центр», 2002.
7. Первин Ю.А., Дувалов А.А., Зайдельман Я.Н., Гольцман М.А. Роботландия. Книга для школьника. М., 1991.
8. Субботина Л.Ю. Развитие воображения у детей. Популярное пособие для детей и педагогов. Ярославль: Академия развития, 1993.

9. Бородин М. Н. – 2-е изд. – М.: БИНОМ. Лаборатория знаний, 2012.
10. Прищепа Т.А. Преподавание программирования в среде КуМир Методическое пособие Томск – 2002 г.
11. Кушниренко А.Г., Леонов А.Г. Методика преподавания основ алгоритмизации на базе системы КуМир (<http://edu.1september.ru/courses/07/010/>).
12. Учебники А.Г. Кушниренко (http://publ.lib.ru/ARCHIVES/K/KUSHNIRENKO_Anatoliy_Georgievich/Kushnirenko_A.G..html).
13. Прищепа Т.А. Преподавание программирования в среде КуМир (http://ido.tsu.ru/other_res/school2/osn/metod/prog/index.html).
14. Крепышев О.Ю. Уроки в среде КуМир, включая “Робот” http://itdo.my1.ru/index/materialy_dlja_skachivanij_a/0-6
Решение задач в системе КуМир (<http://test.kumir.su/>).
Кириенко Д.П. Курс алгоритмизации с использованием исполнителей системы КуМир и автоматического тестирования (<http://server.179.ru/wiki/wakka.php?wakka=DenisKirienko/Kumir>).
Зайдельман Я.Н. Курс “Алгоритмизация и программирование: от первых шагов до подготовки к ЕГЭ”

Перечень программного обеспечения, образовательных ресурсов

1. Матвеева Н., Челак Е., Конопатова Н. Набор цифровых образовательных ресурсов «Информатика и ИКТ 2-11 класс». – М.: БИНОМ. Лаборатория знаний, 2007.
2. Ресурсы Единой коллекции цифровых образовательных ресурсов (<http://school-collection.edu.ru/>).
3. Материалы авторской мастерской Матвеева Н., Челак Е., Конопатова Н. (<http://metodist.lbz.ru/authors/informatika/4>).
4. Материалы авторской мастерской Босова Л. Л. <http://metodist.lbz.ru/authors/informatika/3/>
5. ЭОР Единой коллекции «Виртуальные лаборатории» (<http://school-collection.edu.ru/catalog/rubr>);
6. Лекторий «ИКТ в начальной школе» (<http://metodist.lbz.ru/iections/8/>);
7. Операционная система Windows XP.
8. Пакет офисных приложений MS Office 2007.
9. <http://www.webcont.blg.ru>.
10. www.klyaksa.net.
11. <http://nic-snail.ru>.
12. <http://www.selevko.net/1osnov.php>.
13. Арутюнян Лилит Эдуардовна, http://nsportal.ru/sites/default/files/2013/7/arutyunyan_el.kurs_kumir.docx
14. Евтина Марина Геннадиевна, <http://www.openclass.ru/node/285724>
15. Соловьёва Жанна Николаевна, <http://www.pandia.ru/text/78/150/18425.php>.
16. <http://www.niisi.ru/kumir/>.
17. <http://kpolyakov.spb.ru>