

МИНИСТЕРСТВО ПРОСВЕЩЕНИЯ РОССИЙСКОЙ ФЕДЕРАЦИИ

Министерство образования и науки Пермского края

МАОУ "Гимназия"

УТВЕРЖДЕНА

Директор

МАОУ



РАБОЧАЯ ПРОГРАММА

учебного курса «Основы программирования»

для учащихся 7 класса

Чернушка, 2025 год

Пояснительная записка

Программа «Основы программирования» предназначена для организации внеурочной деятельности по нескольким взаимосвязанным направлениям развития личности, таким как общеинтеллектуальное и общекультурное. Программа предполагает ее реализацию в 7 классе основной школы.

Программа курса способствует развитию творческих способностей, логического мышления, углубления знаний в области алгоритмизации и программирования, расширению общего кругозора учащихся. Курс позволяет успешно готовиться к участию в олимпиадах, конкурсах и к итоговой аттестации по информатике.

Курс поддержан программным обеспечением КуМир (Комплект Учебных МИРов). КуМир - система программирования, предназначенная для поддержки начальных курсов информатики и программирования в основной школе.

КуМир — это свободно распространяемая кроссплатформенная русскоязычная система программирования, предназначенная для начального обучения основам алгоритмизации. Изучая программирование в среде КуМир с исполнителями Робот, Чертежник, Черепаха, Кузнечик, учащиеся прочнее усваивают основы алгоритмизации, приобщаются к алгоритмической культуре, познают азы профессии программиста.

Паскаль (англ. *Pascal*) — один из наиболее известных языков программирования, используется для обучения программированию в старших классах и на первых курсах вузов, является основой для ряда других языков.

Основная цель программы – формирование у учащихся навыков операционного и логического стиля мышления, представления о приемах и методах программирования через составление алгоритмов и программ.

В соответствии с поставленной целью можно выделить следующие **задачи**:

образовательные:

- ✓ способствовать формированию учебно-интеллектуальных умений, приёмов
- ✓ мыслительной деятельности, освоению рациональных способов её осуществления на основе учета индивидуальных особенностей учащихся;
- ✓ способствовать формированию активного, самостоятельного, креативного мышления;
- ✓ научить основным приемам и методам программирования.

развивающие:

- ✓ развивать психические познавательные процессы: мышление, восприятие, память, воображение у учащихся;
- ✓ развивать представление учащихся о практическом значении информатики.

воспитательные:

- ✓ воспитывать культуру алгоритмического мышления;

- ✓ воспитывать у учащихся усидчивость, терпение, трудолюбие.

Общая характеристика учебного курса

Актуальность данной образовательной программы состоит в том, что современные профессии становятся все более интеллектоёмкими, требующими развитого логического мышления. Опоздание с развитием мышления – это опоздание навсегда. Поэтому для подготовки детей к жизни в современном информационном обществе в первую очередь необходимо развивать логическое мышление, способности к анализу и синтезу. Алгоритмическое мышление является необходимой частью научного взгляда на мир. В то же время оно включает и некоторые общие мыслительные навыки, способствует формированию научного мировоззрения, стиля жизни современного человека

Новизна программы основана на раннем изучении азов алгоритмизации и программирования. Программа предполагает раннее знакомство учащихся с основными понятиями, используемыми в языках программирования высокого уровня. Большинство заданий встречаются в разных темах для того, чтобы показать возможности решения одной и той же задачи или проблемы различными средствами, обеспечивающими достижение требуемого результата, что в итоге приведет к способности выбирать оптимальное решение данной задачи или проблемы.

Место курса в учебном плане

Учебный курс «Основы программирования» в 7 классе реализуется за счет вариативного компонента, формируемого участниками образовательного процесса. Используется время, отведенное на внеурочную деятельность. Программа рассчитана на 34 часа в год, 1 час в неделю (одно занятие в неделю по 40 мин).

Формы организации учебного процесса: индивидуальная (самостоятельное усвоение знаний, формирование умений и навыков, развитие самооценки учеников, познавательной самостоятельности), групповая (взаимопомощь, распределение обязанностей, развитие чувства ответственности за результат совместной деятельности, стимул творческого соревнования), парная.

Формы контроля

Предметом контроля и оценки являются составленные алгоритмы и программы учащимися к предложенным задачам в среде «Кумир» и «Паскаль». 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». Оценке подлежит в

первую очередь уровень достижения учеников минимально необходимых результатов, обозначенных в целях и задачах курса.

Контроль за усвоением качества знаний должен проводиться на трех уровнях:

1-й уровень – воспроизводящий (репродуктивный) – предполагает воспроизведение знаний и способов деятельности. Учащийся воспроизводит учебную информацию, выполняет задания по образцу.

2-й уровень – конструктивный предполагает преобразование имеющихся знаний. Ученик может переносить знания в измененную ситуацию, в которой он видит элементы, аналогичные усвоенным;

3-й уровень – творческий предполагает овладение приемами и способами действия. Ученик осуществляет перенос знаний в незнакомую ситуацию, создает новые нестандартные алгоритмы познавательной деятельности.

Качество знаний и умений ученика оценивается следующими характеристиками:

- знание основных алгоритмических конструкций;
- умение составить и записать алгоритм с использованием соответствующей алгоритмической конструкции;
- умение найти более эффективный способ решения задачи;
- умение тестировать программу.

Текущий контроль знаний осуществляется по результатам выполнения учащимися практических заданий.

Выполненные учащимися работы включаются в их «портфель достижений».

Итоговый контроль реализуется в форме защиты собственных программ-проектов учащихся или группы учащихся.

Личностные, метапредметные и предметные результаты освоения учебного курса

В результате изучения курса получают дальнейшее развитие личностные, регулятивные, коммуникативные и познавательные универсальные учебные действия, учебная (общая и предметная) и общепользовательская ИКТ-компетентность обучающихся.

В основном формируются и получают развитие **метапредметные результаты**, такие как:

умение самостоятельно планировать пути достижения целей, осознанно выбирать наиболее эффективные способы решения учебных и познавательных задач;

умения соотносить свои действия с планируемыми результатами, осуществлять контроль своей деятельности в процессе достижения результата;

умение оценивать правильность выполнения учебной задачи, собственные возможности ее решения;

формирование и развитие компетентности в области использования информационно-коммуникационных технологий (далее ИКТ-компетентности).

Вместе с тем вносится существенный вклад в развитие личностных результатов, таких как:

формирование ответственного отношения к учению;

формирование коммуникативной компетентности в общении и сотрудничестве со сверстниками, детьми старшего и младшего возраста, взрослыми в процессе образовательной, творческой и других видов деятельности.

формирование способности обучающихся к саморазвитию и личностному самоопределению, мотивации к целенаправленной познавательной деятельности с целью приобретения профессиональных навыков в ИТ-сфере;

способность и готовность к принятию ценностей здорового образа жизни за счет знания основных гигиенических, эргономических и технических условий безопасной эксплуатации средств ИКТ.

В части развития предметных **результатов** наибольшее влияние изучение курса оказывает:

умение использовать термины понятий «алгоритм», «данные», «программа» через

призму практического опыта в ходе создания программных кодов; понимание различий между употреблением этих терминов в обиходной речи и в информатике;

умение создавать и выполнять программы для решения несложных алгоритмических задач в среде КУМИР и ПАСКАЛЬ;

практические навыки создания линейных алгоритмов управления исполнителями;

умение формально выполнять алгоритмы, описанные с использованием конструкций ветвления (условные операторы) и повторения (циклы), вспомогательных алгоритмов;

умение создавать и выполнять программы для решения несложных алгоритмических задач в выбранной среде программирования.

Регулятивные универсальные учебные действия

Обучающийся научится:

целеполаганию, включая постановку новых целей, преобразование практической задачи в познавательную;

самостоятельно анализировать условия достижения цели на основе учета выделенных учителем ориентиров действия в новом учебном материале;

планировать пути достижения целей; уметь самостоятельно контролировать свое время и управлять им.

Коммуникативные универсальные учебные действия

Обучающийся научится:

устанавливать и сравнивать разные точки зрения, прежде чем принимать решения и делать выбор;

аргументировать свою точку зрения, спорить и отстаивать свою позицию не враждебным для оппонентов образом;

задавать вопросы, необходимые для организации собственной деятельности и сотрудничества с партнером;

осуществлять взаимный контроль и оказывать в сотрудничестве необходимую взаимопомощь.

Познавательные универсальные учебные действия

Обучающийся научится:

создавать и преобразовывать модели и схемы для решения задачи;

осуществлять выбор наиболее эффективных способов решения задач в зависимости от конкретных условий.

Характеристика деятельности ученика

Аналитическая деятельность:

приводить примеры формальных и неформальных исполнителей;

придумывать задачи по управлению учебными исполнителями;

выделять примеры ситуаций, которые могут быть описаны с помощью линейных алгоритмов, алгоритмов с ветвлениями и циклами.

Практическая деятельность:

составлять линейные алгоритмы и программы по управлению учебным исполнителем;

составлять циклические алгоритмы по управлению учебными исполнителями;

составлять алгоритмы с ветвлением по управлению учебным исполнителем;

составлять вспомогательные алгоритмы для управления учебными исполнителями.

Содержание программы курса внеурочной деятельности «Основы программирования»

Раздел 1. Алгоритмы и исполнители (4 часа)

Понятие исполнителя. Неформальные и формальные исполнители. Исполнители алгоритмов. Система команд исполнителя. Понятие алгоритма. Способы записи алгоритмов. Основные алгоритмические конструкции. Понятие оптимизации алгоритмов. Программа, ошибки. Выбор необходимой алгоритмической конструкции для решения поставленной задачи.

Раздел 2. Компьютерные исполнители алгоритмов в среде Кумир (16 часов)

Знакомство со средой Кумир. Учебные исполнители: Кузнечик, Водолей, Черепаха, Чертежник, Робот как примеры формальных исполнителей. Их назначение, среда, режим работы, система команд. Управление исполнителями с помощью команд и их последовательностей. Составление линейных, с ветвлениями и циклами алгоритмов и программ для управления исполнителями Кузнечик, Водолей, Черепаха, Чертежник, Робот в среде Кумир.

Раздел 3. Программирование с использованием языка Паскаль. (12 часов)

Знакомство со средой программирования Паскаль. Его назначением, средой, режимами работы, системой команд. Составление алгоритмов и программ с использованием линейных, циклических, разветвляющихся конструкций.

Раздел 4. Свободное проектирование. Итоговое повторение (2 часа)

Итоговое повторение. Интеллектуальный марафон «Основы программирования». Компьютерный практикум.

Практические работы на каждом уроке по соответствующей теме.

КАЛЕНДАРНО-ТЕМАТИЧЕСКОЕ ПЛАНИРОВАНИЕ

№ п/п	Дата	№	Тема	Примечание
Алгоритмы и исполнители (4 часа)				
1		1.1	Вводный урок. Инструктаж по технике безопасности. Исполнители вокруг нас. Игра «Что мы знаем про алгоритмы?»	Игра «Что мы знаем про алгоритмы?»
2		1.2	Понятие исполнителя. Формальные и неформальные исполнители. Система команд исполнителя.	
3		1.3	Исполнители алгоритмов. Понятие алгоритма. Способы записи алгоритмов (нумерованный список, таблица, блок-схема)	Примеры алгоритмов с ветвлениями и циклами в повседневной жизни
4		1.4	Основные алгоритмические конструкции. Программа, ошибки, типы ошибок.	
Управление компьютерными исполнителями алгоритмов в среде Кумир (16 часов)				
5		2.1	Знакомство со средой Кумир. Учебные исполнители (Водолей, Кузнечик, Черепаха, Чертежник, Робот) как примеры формальных исполнителей.	
6		2.2	Исполнитель Кузнечик. Среда обитания, СКИ. Решение задач и разработка программ для исполнителя Кузнечик.	
7		2.3	Составление линейных программ для исполнителя Кузнечик.	Выполнение проекта «Кузнечик»
8		2.4	Исполнитель Водолей. Среда обитания, СКИ. Решение задач и разработка	

			программ для исполнителя Водолей.	
9		2.5	Решение задач и составление программ для исполнителя Водолей.	Выполнение проекта «Водолей»
10		2.6	Исполнитель Черепаха. Среда обитания, СКИ. Решение задач и разработка программ для исполнителя Черепаха.	
11		2.7	Построение геометрических фигур и орнаментов с помощью исполнителя Черепаха.	Выполнение проекта «Черепаха»
12		2.8	Исполнитель Чертежник. Среда обитания, СКИ. Решение задач и разработка программ для исполнителя Чертежник.	Различать команды переместиться в точку и сместиться на вектор
13		2.9	Основные и вспомогательные программы. Составление линейных программ для управления исполнителем Чертежник.	Выполнение проекта «Чертежник»
14		2.10	Исполнитель Робот. Среда обитания, СКИ. Решение задач и разработка программ для исполнителя Робот.	
15		2.11	Составление линейных алгоритмов для исполнителя Робот	
16		2.12	Основные базовые алгоритмические конструкции (ветвление) и их реализация в среде исполнителя Робот.	
17		2.13	Основные базовые алгоритмические конструкции (цикл со счетчиком) и их реализация в среде исполнителя Робот.	Цикл «n раз»
18		2.14	Основные базовые алгоритмические конструкции (цикл с условием) и их реализация в среде исполнителя Робот.	Использовать повторение фрагментов алгоритма при создании орнамента
19		2.15	Решение задач и составление программ для исполнителя Робот.	Выполнение проекта «Робот»
20		2.16	Обобщение раздела Управление компьютерными исполнителями алгоритмов в среде Кумир	Игра «Чему мы научились»

Программирование с использованием языка Паскаль (12 часов)				
21		3.1	Возникновение и назначение языка Pascal. Структура программы на языке PascalABC.	
22		3.2	Алгоритмы работы с величинами: константы, переменные, основные типы, присваивание, ввод и вывод данных	
23		3.3	Линейный алгоритм	
24		3.4	Создание программ линейной структуры. Операторы ввода и вывода.	
25		3.5	Выполнение проекта « <i>Линейная программа</i> »	Создание программ линейной структуры самостоятельно
26		3.6	Разветвляющийся алгоритм	
27		3.7	Создание программ разветвляющейся структуры.	
28		3.8	Выполнение проекта « <i>IfThenElse</i> »	Создание программ с ветвлением самостоятельно
29		3.9	Алгоритм циклической структуры	
30		3.10	Знакомство с видами циклов « <i>For</i> », « <i>While do</i> », « <i>Repeat until</i> »	
31		3.11	Создание программ циклической структуры	
32		3.12	Выполнение проекта « <i>For, While do, Repeat until</i> »	Создание программ с циклом самостоятельно
Свободное проектирование. Итоговое повторение. (2 часа)				
33		4.1	Интеллектуальный марафон «Нескучное программирование»	
34		4.2	Итоговое занятие. Игра «Битва Титанов»	

Требования к подготовке учащихся

Учащиеся должны:

- Иметь представление о системах программирования, средах и языках программирования.

Знать:

- Что такое формальные языки как способ представления алгоритмов;
- Что такое программирование и этапы решения задачи на ПК;
- Понятие программы и ее структуры;
- Понятие входных параметров;
- Понятие команды в программе и правила записи команд в программе;
- Назначение среды «Кумир» и «Pascal»

Уметь:

- Приводить примеры представления алгоритмов на формальном языке;
- Представлять алгоритмы на формальном языке (в блок-схеме, графически, с помощью пиктограмм);
- Запускать программы «Кумир» и «Pascal»;
- Выбирать нужного исполнителя;
- Запускать программу на исполнение;
- Находить ошибки в программе и исправлять их;
- Владеть основными понятиями:
 - Алгоритм
 - Свойства алгоритма
 - Исполнитель, среда исполнителя, СИИ
 - Переменная – имя, тип, значение
 - Алгоритмическая конструкция – следование, ветвление, цикл
 - Блок-схема, алгоритмический язык.

Учебно-методическое обеспечение

Литература для педагога:

1. Анеликова Л.А., Гусева О.Б. Программирование на алгоритмическом языке КУМИР. – Москва, СОЛОН-ПРЕСС, 2013.
2. Босова Л.Л., Сорокина Т.Е. Методика применения интерактивных сред для обучения младших школьников программированию: Информатика и образование №7(256) сентябрь 2014 г.
3. Костюк Ю.Л. Информатика для начинающих программистов. – Томск, Издательство Томского университета, 1997.
4. Информатика. Методическое пособие для 7-9 классов. Босова Л.Л., Босова А.Ю. .– М.: БИНОМ. Лаборатория знаний, 2015
5. Окулов, С.М. Программирование в алгоритмах/ С.М. Окулов. – М.:БИНОМ, Лаборатория знаний, 2013.
6. Программы внеурочной деятельности для основной школы. 7-9 классы. Цветкова М.С., Богомолова О.Б. . – М.: БИНОМ. Лаборатория знаний, 2013.
7. Поляков К.Е. Алгоритмы и исполнители. Учебник по алгоритмизации. (Доступ: <https://docs.google.com/file/d/0BxInd4PRGJMmNEViWDVtbVp6Rlk/edit?pli=1>)
8. Прищепа Т.А. Преподавание программирования в среде КуМир. Методическое пособие (Доступ: http://ido.tsu.ru/other_res/school2/osn/metod/prog/index.html)
9. Удалова Т.Л., Ануфриева М.Н. Информатика. КуМир. – Саратов: Лицей, 2012.

Литература для учащихся:

1. Анеликова Л.А., Гусева О.Б. Программирование на алгоритмическом языке КУМИР. – Москва, СОЛОН-ПРЕСС, 2013.
2. Поляков К.Е. Алгоритмы и исполнители. Учебник по алгоритмизации. (Доступ: <https://docs.google.com/file/d/0BxInd4PRGJMmNEViWDVtbVp6Rlk/edit?pli=1>)
3. Прищепа Т.А. Преподавание программирования в среде КуМир. Методическое пособие (Доступ: http://ido.tsu.ru/other_res/school2/osn/metod/prog/index.html)
4. Удалова Т.Л., Ануфриева М.Н. Информатика. КуМир. – Саратов: Лицей, 2012.

Интернет ресурсы

- ▲ <https://www.niisi.ru/kumir/> – Сайт НИИСИ РАН
- ▲ <http://kpolyakov.spb.ru/school/kumir.htm> – Сайт Константина Полякова
- ▲ <http://kpolyakov.spb.ru/download/kumkurs.pdf> – Практикумы в КуМир. К.Ю. Поляков.
- ▲ <https://docs.google.com/file/d/0BxInd4PRGJMmNEViWDVtbVp6Rlk/edit?pli=1> – Поляков К.Е. Алгоритмы и исполнители. Учебник по алгоритмизации
- ▲ <https://sites.google.com/site/fakultativinformatika/home/zanatie-1-znakomstvo-s-cerepahoj> – Программирование в системе КУМИР
- ▲ <http://edusar.soiro.ru/course/view.php?id=475> – Алгоритмизация в среде КУМир
- ▲ <http://www.klyaksa.net/htm/konspektsch/kumir/index.htm> – сайт Клякс@.net: Информатика в школе. Компьютер на уроках
- ▲ <http://ftl1.ru/udalova-tl.html> – Электронные образовательные ресурсы, разработанные учителем информатики и ИКТ Удаловой Т.Л.
- ▲ http://licey.net/free/18-programmirovanie_na_yazyke_kumir.html – Программирование на языке КуМир
- ▲ <https://books.google.ru/books?id=CdAqevFDPa4C&printsec=frontcover&hl=ru#v=onepage&q&f=false> – Удалова Т.Л., Ануфриева М.Н. Информатика. КуМир (в электронном доступе).

**ДОКУМЕНТ ПОДПИСАН
ЭЛЕКТРОННОЙ ПОДПИСЬЮ**

СВЕДЕНИЯ О СЕРТИФИКАТЕ ЭП

Сертификат 143507986500560089701835989304833372774460075108

Владелец Усанина Наталья Леонидовна

Действителен с 03.04.2025 по 03.04.2026