

муниципальное бюджетное общеобразовательное учреждение  
«Средняя общеобразовательная школа № 5  
имени Героя Российской Федерации Клещенко Василия Петровича»



«Утверждена»  
решением педсовета  
№8 от «31» августа 2023 г.  
и.о. директора школы

Воронина Е. В.

Образовательная программа  
дополнительного образования  
кружок «Лаборатория прототипирования»  
2023-2024 учебный год  
(для обучающихся 3-11 классов,  
минимальный возраст – 9 лет,  
срок реализации – 1 год)

Направление: техническое

Руководитель: Сорокин М.М.

г. Торжок, 2023

## ПОЯСНИТЕЛЬНАЯ ЗАПИСКА

Данная программа рассчитана на 5 лет реализации и включает в себя 5 модулей:

- Основы программирования на Кумире
- Робототехника
- Основы программирования на Паскале
- 3D моделирование
- Олимпиадное программирование.

Эти модули учащиеся могут проходить, последовательно начиная с 3 класса, либо отдельные модули по их желаниям и способностям. Каждый отдельный модуль рассчитан на изучение в течение одного учебного года.

Такое построение программы позволяет мультипредметно использовать имеющееся в лаборатории оборудование. Роботов для демонстрации принципов программирования, 3D принтеров для создания частей к роботам, программных алгоритмов для написания скетчей.

Примерный возраст учеников, по модулям

- 1 модуль – 9-10 лет
- 2 модуль – 11-15 лет
- 3 модуль – 13-15 лет
- 4 модуль – 14-16 лет
- 5 модуль – 16-17 лет

Начиная со второго модуля, учащиеся оканчивают курс, с выполнением и защитой творческих проектов в области ИКТ технологии. Что предполагает возможность их участия в конкурсах различных уровней.

## МОДУЛЬ «ОСНОВЫ ПРОГРАММИРОВАНИЯ НА КУМИРЕ»

### Пояснительная записка

Рабочая программа внеурочной деятельности составлена на основе федерального государственного образовательного стандарта основного общего образования и соответствует требованиям к результатам освоения образовательной программы (личностным, метапредметным, предметным); основным подходам к развитию и формированию универсальных учебных действий (УУД) для основного общего образования. В ней учитываются возрастные и психологические особенности школьников, обучающихся на ступени основного общего образования, учитываются межпредметные связи.

Рабочая программа разработана для организации внеурочной деятельности для учащихся с 11 до 13 лет, рассчитана на 34 часов в год (1 час в неделю). Изучение курса предполагает наличие в школе компьютерного класса, предусмотрено проведение практических работ.

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

Программа реализует общеинтеллектуальное направление во внеурочной деятельности.

#### **Цель программы:**

Помочь учащимся заинтересоваться программированием, сформировать у школьников знания, умения и навыки решения задач по программированию и алгоритмизации.

В ходе ее достижения решаются **задачи:**

#### **Обучающие:**

- Обучение основным базовым алгоритмическим конструкциям.
- Освоение основных этапов решения задачи.
- Обучение навыкам разработки, тестирования и отладки несложных программ.
- Обучение навыкам разработки проекта, определения его структуры, дизайна.

#### **Развивающие:**

- Развивать познавательный интерес школьников.
- Развивать творческое воображение, математическое и образное мышление учащихся.
- Развивать умение работать с компьютерными программами и дополнительными источниками информации.
- Развивать навыки планирования проекта, умение работать в группе

#### **Воспитывающие:**

- Воспитывать интерес к занятиям информатикой.
- Воспитывать культуру общения между учащимися.
- Воспитывать культуру безопасного труда при работе за компьютером.
- Воспитывать культуру работы в глобальной сети.

## ***Личностные и метапредметные результаты освоения курса внеурочной деятельности***

В результате изучения данной программы обучающиеся получают возможность формирования:

### **Личностных результатов:**

- готовность и способность обучающихся к саморазвитию и личностному самоопределению, мотивация к целенаправленной познавательной деятельности с целью приобретения профессиональных навыков в ИТ-сфере.

### **Метапредметных результатов**

#### **Регулятивные УУД:**

- умение самостоятельно планировать пути достижения целей, соотносить свои действия с планируемыми результатами, осуществлять контроль и коррекцию своей деятельности в процессе достижения результата.

#### **Коммуникативные УУД:**

- умения организовывать продуктивное сотрудничество и совместную деятельность с учителем и сверстниками.

#### **Познавательные УУД:**

- поиск и выделение необходимой информации;
- построение логической цепи рассуждений;
- самостоятельное создание способов решения проблем творческого и поискового характера.

### **Предметных результатов:**

- освоение понятий «алгоритм», «программа» через призму практического опыта в ходе создания программных кодов;
- 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

## **Учебно-тематический план**

№	Название разделов	Общее количество часов	В том числе	
			теория	практика
1	Понятие исполнителя	1	1	
2	Учебные исполнители	4	2	2
3	Что такое алгоритм.	30	9	20
	Итого часов	34	12	22

## Содержание программы

### Раздел 1.

Понятие исполнителя. Неформальные и формальные исполнители. Среда Кумир.

### Раздел 2.

Учебные исполнители (Кузнечик, Водолей, Робот, Черепаха) как примеры формальных исполнителей. Их назначение, среда, режим работы, система команд. Управление исполнителями с помощью команд и их последовательностей.

### Раздел 3.

Что такое алгоритм. Различные формы записи алгоритмов (нумерованный список, таблица, блок-схема). Примеры линейных алгоритмов, алгоритмов с ветвлениями и повторениями (в повседневной жизни, в литературных произведениях, на уроках математики и т.д.). Составление алгоритмов и программ (линейных, с ветвлениями и циклами) для управления исполнителями Кузнечик, Водолей, Робот, Черепаха.

## Методическое обеспечение программы

### *Формы организации учебных занятий*

Беседы, игры, практические занятия, самостоятельная работа, викторины и проекты. Использование метода проектов позволяет обеспечить условия для развития у ребят навыков самостоятельной постановки задачи выбора оптимального варианта их решения, самостоятельного достижения цели, анализа полученных результатов с точки зрения решения поставленной задачи. Для контроля знаний учащихся проводится входной, промежуточный и итоговый мониторинг образовательных результатов. *Приложение №1.*

Программой предусмотрены **методы обучения**: объяснительно-иллюстративные, частично-поисковые (вариативные задания), творческие, практические.

### *Виды учебной деятельности*

#### *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.

## Календарно-тематическое планирование

Дата	№	Тема	Количество часов	
			Теория	Практик
Понятие алгоритма (1 час)				
		Инструктаж по технике безопасности. Алгоритмы и исполнители. Кумир. <b>Входной мониторинг</b>	1	
Учебные исполнители (4 часа)				
		Система команд исполнителя. Решение задач для исполнителя Кузнечик.	0,5	0,5
		Исполнитель Черепаха. Среда обитания, система команд.	0,5	0,5
		Исполнитель Водолей. Среда обитания, система команд.	0,5	0,5
		Исполнитель Робот. Среда обитания, система команд.	0,5	0,5
Что такое алгоритм (30 часов)				
		Способы записи алгоритмов. Решение задач для исполнителя Кузнечик.	0,5	0,5
		Виды алгоритмов. Составление линейных алгоритмов для исполнителя Кузнечик	1	
		Составление циклических алгоритмов для исполнителя Кузнечик	1	
		Решение задач для исполнителя Кузнечик		1
		Решение задач для исполнителя Водолей.		1
		Решение задач для исполнителя Водолей.		1
		Составление линейных алгоритмов для исполнителя Черепаха.		1
		Цикл со счетчиком. Решение задач для исполнителя Черепаха.	0,5	0,5
		Цикл со счетчиком. Решение задач для исполнителя Черепаха.	0,5	0,5
		Вложенные циклы. Решение задач для исполнителя Черепаха.	0,5	0,5
		Построение геометрических фигур с помощью исполнителя Черепа		1
		<b>Промежуточный мониторинг: Построение орнаментов с помощью исполнителя Черепаха.</b>		1
		Составление линейных алгоритмов для исполнителя Робот. Лабиринт	1	
		Цикл со счетчиком. Решение задач для исполнителя Робот.	0,5	0,5
		Цикл со счетчиком. Решение задач для исполнителя Робот.		1
		Вложенные циклы. Решение задач для исполнителя Робот.	0,5	0,5
		Цикл с условием. Решение задач для исполнителя Робот.		1
		Цикл с условием. Решение задач для исполнителя Робот.		1
		Ветвления. Решение задач для исполнителя Робот.	0,5	0,5
		Ветвления. Решение задач для исполнителя Робот.		1
		Сложные условия. Решение задач для исполнителя Робот.	0,5	0,5
		Сложные условия. Решение задач для исполнителя Робот.		1
		Вспомогательные алгоритмы. Решение задач для исполнителя Робо	0,5	0,5
		Вспомогательные алгоритмы. Решение задач для исполнителя Робо	0,5	0,5
		Переменные. Решение задач для исполнителя Робот.		1

		Циклы с переменной. Решение задач для исполнителя Робот.		1
		Циклы с переменной. Решение задач для исполнителя Робот.		1
		Решение задач для исполнителя Робот.		1
		<b>Итоговый мониторинг</b>		1
		<b>Итог</b>	12	22

## Приложения

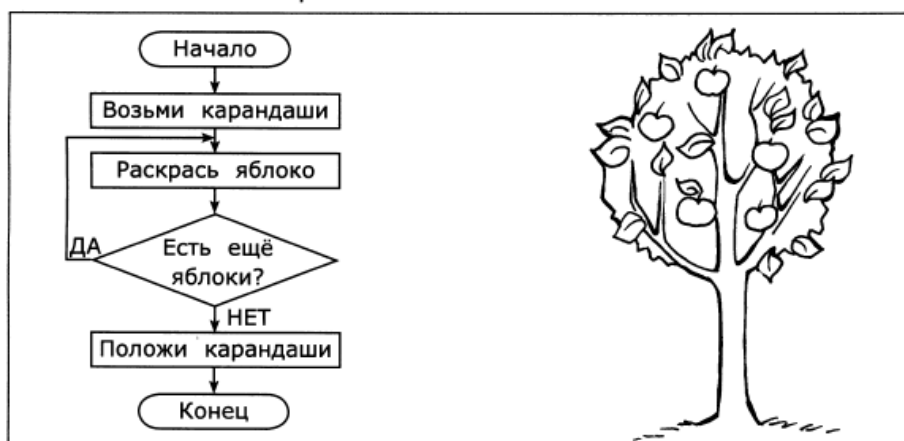
### Приложение №1

#### Мониторинг образовательных результатов

##### 1. Входной мониторинг

**Цель:** Выявить расположенность обучающихся к логическому, алгоритмическому мышлению.

3. Выполни задание по алгоритму «РАСКРАСЬ ЯБЛОКИ». Какая команда выполнялась несколько раз?



##### 2. Промежуточный мониторинг

**Цель:** повторить и закрепить основные понятия, изученные в первом полугодии.

Промежуточный мониторинг проводится в виде усложненной практической работы:  
«Построение орнаментов с помощью исполнителя Черепаха».

##### 3. Итоговый мониторинг

Итоговый мониторинг проводится в виде подготовки обучающимися проектных работ.  
Темы проектов:

## МОДУЛЬ «РОБОТОТЕХНИКА»

### Пояснительная записка

Программа включает 34 часа аудиторных занятий.

Курс предполагает знакомство с основами программирования на языке высокого уровня. Предметом изучения являются принципы и методы разработки, конструирования и программирования управляемых электронных устройств на базе вычислительной платформы (контроллера) Ардуино или её клона, а также создание робототехнических устройств в рамках небольших проектов.

Целесообразность изучения данного курса определяется:

- востребованностью специалистов в области программируемой микроэлектроники в современном мире
- возможностью развить и применить на практике знания, полученные на уроках математики, физики, информатики
- возможностью предоставить ученику образовательную среду, развивающую его творческие способности и амбиции, формирующую интерес к обучению, поддерживающую самостоятельность в поиске и принятии решений.

### Цели курса

- познакомить учащихся с принципами и методами разработки, конструирования и программирования управляемых электронных устройств на базе вычислительной платформы Ардуино;
- развить навыки программирования в современной среде программирования углубить знания, повысить мотивацию к обучению путем практического интегрированного применения знаний, полученных в различных образовательных областях (математика, физика, информатика);
- развить интерес к научно-техническому, инженерно-конструкторскому творчеству развить творческие способности учащихся.

### Задачи курса

Первый уровень - репродуктивный (ученик понимает, может воспроизвести без ошибок).

Второй уровень - «интерпретация» (ученик понимает, может применить с изменениями в похожей ситуации).

Третий уровень - «изобретение» (ученик может самостоятельно спроектировать, сконструировать и запрограммировать устройство, решающее поставленную перед ним практическую задачу)

**Первый уровень:** на базе Ардуино с использованием макетной платы и набора электронных элементов научить учащихся

- понимать заданные схемы («схема на макетке») электронных устройств и воспроизводить их на макетной плате
- понимать назначение элементов, их функцию
- понимать правила соединения деталей в единую электрическую цепь
- понимать ограничения и правила техники безопасности функционирования цепи
- понимать написанный программный код управления устройством, вносить незначительные изменения, не затрагивающие структуру программы (например, значения констант)



- записывать отлаженный программный код на плату Ардуино, наблюдать и анализировать результат работы
- использовать монитор последовательного порта для отладки программы, наблюдения за показателями датчиков и изменением значений переменных
- организовывать беспроводную передачу данных на ПК
- создавать графические интерфейсы для взаимодействия с устройством на Ардуино.

**Второй уровень:** на базе Ардуино с использованием макетной платы и набора электронных элементов научить учащихся:

- понимать заданные схемы («принципиальная схема» и «схема на макетке»)
- электронных устройств и воспроизводить их на макетной плате
- понимать назначение элементов, их функцию
- понимать правила соединения деталей в единую электрическую цепь
- понимать ограничения и правила техники безопасности функционирования цепи
- модифицировать заданные схемы для измененных условий задачи
- понимать написанный программный код управления устройством и модифицировать его для измененных условий задачи
- самостоятельно отлаживать программный код, используя, в частности, такие средства как мониторинг показаний датчиков, значений переменных и т. п.
- записывать отлаженный программный код на плату Ардуино, наблюдать и анализировать результат работы, самостоятельно находить ошибки и исправлять их

**Третий уровень** предполагает достижение результатов второго уровня и, кроме того, умение учащихся самостоятельно проектировать, конструировать и программировать устройство, которое решает практическую задачу, сформулированную учителем или самостоятельно.

Основной формой обучения является практическая работа, которая выполняется малыми (2 человека) группами. Для работы необходим персональный компьютер (один на каждую группу), установленное программное обеспечение (может быть установлено с сайта <http://arduino.cc/en/Main/Software>), контроллер Arduino Uno или его клон (1 на каждую группу).

### **Требования по аппаратно-программному обеспечению**

*Из расчёта на 10 человек в кружке.*

- Набор «Матрёшка Z» - 5 шт.
- Образовательный набор «Амперка» - 3 шт.
- 3D-принтер - 1 шт.
- Дополнительные механические детали и исполнительные механизмы по мере необходимости в рамках проектов учащихся.
- Компьютеры - 10 шт.

Программное обеспечение на каждом ПК:

- ОС Windows версии XP SP3 и выше.
- Высокоскоростной доступ к сети Интернет.
- MS Visual Studio, либо другая среда для на
- IDE Arduino.
- ПО для симуляции Arduino «123D Circuits».
- ПО для создания 3D-моделей «123D Designer».
- САПР «Компас 3D».
- MS Office версии 2007 и выше.

### ***Использованные материалы***

1. Дистанционный курс на сайте amperka.ru <http://wiki.amperka.ru/КОHcneKT-arduino>
2. «Основы программирования микроконтроллеров» Учебник для образовательного набора «Амперка», Москва 2013
3. Список ссылок на сайте Arduino, do it! <https://sites.google.com/site/arduinoitoit/>

№ занятия	Название темы	Количество часов (теория/практ)	дата
1	<b>Техника безопасности Роботы вокруг нас.</b>	1 час(1/0)	08.09
2-3	<b>Контроллер Ардуино</b> • Микроконтроллеры в нашей жизни (сообщения учеников), контроллер, контролер Ардуино • Среда программирования для Ардуино (IDE Arduino) и язык программирования Processing	2 часа (1/1)	15.09 22.09
4-5	<b>Основы проектирования и моделирования электронного устройства на базе Ардуино</b> •Управление электричеством. Законы электричества. •Как быстро строить схемы: макетная доска (breadboard). •Чтение электрических схем. •Управление светодиодом на макетной доске.	2 часа(1/1)	29.09 06.10
6-7	<b>Широтно-импульсная модуляция</b> •Аналоговые и цифровые сигналы, понятие ШИМ. •Управление устройствами с помощью портов, поддерживающих ШИМ. •Циклические конструкции, датчик случайных чисел.	2 часа(1/1)	13.10 20.10
8-9	<b>Программирование Ардуино. Пользовательские функции</b> •Подпрограммы: назначение, описание и вызов, параметры, локальные и глобальные переменные.	2 часа(1/1)	27.10 10.11
10-11	<b>Сенсоры. Датчики Ардуино</b> •Роль сенсоров в управляемых системах. •Сенсоры и переменные резисторы. •Делитель напряжения. Потенциометр. •Аналоговые сигналы на входе Ардуино, фильтрация сигналов. •Использование монитора последовательного порта для наблюдений за параметрами системы	2 часа(1/1)	17.11 24.11
12-13	<b>Библиотеки, класс, объект</b> •Что такое библиотеки, использование библиотек в программе. •Библиотека math.h, использование математических функций в программе.	2 часа(1/1)	01.12 08.12
14-15	<b>Жидкокристаллический экран</b> •Назначение и устройство жидкокристаллических экранов. •Библиотека LiquidCrystal. Вывод сообщений на экран	2 часа (1/1)	15.12 22.15
16-17	<b>Транзистор - управляющий элемент схемы</b> •Назначение, виды и устройство транзисторов. •Использование транзистора в моделях, управляемых Ардуино.	2 часа (1/1)	19.01 26.01
18-19	<b>Управление двигателями</b> •Разновидности двигателей: постоянные, шаговые, серводвигатели. •Управление коллекторным двигателем. •Управление скоростью коллекторного двигателя.	2 часа (1/1)	02.02 09.02

	•Управление серводвигателем: библиотека Servo.h		
20-21	<b>Индивидуальная творческая работа по изученному материалу.</b>	2 часа (0/2)	16.02 02.03
22-24	<b>Управление Ардуино через USB и беспроводную связь.</b> •Использование Serial Monitor для передачи текстовых сообщений на Ардуино. •Преобразование текстовых сообщений в команды для Ардуино. •Программирование: объекты, объект String, цикл while, оператор выбора case.	3 часа (1/2)	16.03 23.03
25-28	<b>Создание интерфейсов управления на ПК</b> •Знакомство со графической средой программирования для ПК. •Приём и отправка данных через последовательный порт.	4 часа (2/2)	30.03 06.04
29-34	<b>Проект робототехнического устройства. •Этапы проектирования.</b> •САПР. •Прототипирование. •Представление проекта.	6 часов (1/4)	13.04 20.04 27.04 04.05 11.05 18.05

**Всего за год 34 часа**

## **МОДУЛЬ «ОСНОВЫ ПРОГРАММИРОВАНИЯ НА ПАСКАЛЕ»**

### **Пояснительная записка**

Одна из задач профильной школы – содействовать воспитанию нового поколения, отвечающего по своему уровню развития и образу жизни условиям информационного общества. Для этого учащимся необходимо анализировать информацию, выявлять в ней факты и проблемы, самостоятельно ставить задачи, структурировать и преобразовывать информацию, использовать ее для решения учебных и жизненных задач.

Кружок является предметом по выбору для учащихся 7-9 классов старшей школы.

Курс рассчитан на 35 часов, которые проводятся в течение учебного года по 1 часу в неделю. Концентрированное изучение курса позволяет учащимся более полно выявить свои способности в изучаемой области знаний, создать предпосылки по применению освоенных умений в других учебных курсах, подготовить себя к осознанному выбору профессий, предусматривающих программирование.

Курс включает в себя практическое освоение языка программирования, знакомство учащихся с ролью программного обеспечения и его видами; нацелен на формирование целостного представления об организации данных для эффективной алгоритмической обработки; на развитие логического мышления и реализацию математических способностей учащихся в ходе составления программ на языке программирования.

Основа курса – личностная, практическая и продуктивная направленность занятий. Одна из целей обучения информатике – предоставить ученикам возможность личностного самоопределения и самореализации по отношению к стремительно развивающимся информационным технологиям и ресурсам.

#### **Цели курса:**

- Познакомить учащихся с ролью программного обеспечения и его видами.
- Сформировать целостное представление об организации данных для эффективной алгоритмической обработки.
- Развитие логического мышления.
- Реализация математических способностей учащихся в ходе составления программ на языке программирования.

#### **Задачи курса:**

##### **Обучающие:**

- Познакомить учащихся с основными алгоритмическими конструкциями и правилами их записи, с основными способами организации данных.
- Научить учащихся составлять и записывать алгоритмы с использованием соответствующих алгоритмических конструкций.
- Научить распознавать необходимость применения той или иной алгоритмической конструкции при решении задач.
- Научить организовывать данные для эффективной алгоритмической обработки.
- Научить учащихся разрабатывать алгоритмы и реализовывать их на языке программирования Pascal .
- Научить учащихся осуществлять отладку и тестирование программы.

### ***Развивающие:***

- формировать новый тип мышления – операционный, который направлен на выбор оптимальных решений;
- предоставление возможности узнать новое в области компьютерного программирования;
- формирование представления о роли компьютерного программирования в развитии общества, изменении содержания и характера деятельности человека.

### ***Воспитательные:***

- повышение общекультурного уровня учащихся;
- вооружение учащихся правильным методологическим подходом к познавательной и практической деятельности;
- выделение и раскрытие роли информационных технологий и компьютеров в развитии современного общества;
- привитие навыков сознательного и рационального использования компьютера в своей учебной, а затем и профессиональной деятельности;
- формирование эмоционально-ценностного отношения к миру, к себе;
- воспитание у учащихся стремления к овладению техникой исследования;
- воспитание трудолюбия, инициативности и настойчивости в преодолении трудностей.

**Возраст детей:** 14-16 лет.

**Сроки реализации курса** 1 год.

**Формы занятий:** Основными, характерными при реализации данной программы формами являются комбинированные занятия. Занятия состоят из теоретической и практической частей, причём большее количество времени занимает практическая часть.

В теоретической части рассматриваются основные понятия языка программирования Pascal, основные алгоритмические конструкции. В практической части предлагаются практические работы, направленные на отработку основных алгоритмических конструкций, на развитие логического мышления, на реализацию математических способностей учащихся в ходе составления программ. Практическая часть предполагает использование школьного компьютерного класса.

При проведении занятий традиционно используются три формы работы:

- демонстрационная, когда обучающиеся слушают объяснения педагога и наблюдают за демонстрационным экраном или экранами компьютеров на ученических рабочих местах;
- фронтальная, когда обучающиеся синхронно работают под управлением педагога;
- самостоятельная, когда обучающиеся выполняют индивидуальные задания в течение части занятия или нескольких занятий.

**Технологии и формы обучения:**

- теоретические занятия;
- практические занятия.

**Режим занятий:**

Занятия проводятся: 1 раз в неделю по 1 часу (итого 1 час в неделю, 34 часа в год).

Элективный курс предусматривает классно-урочную систему обучения.

## **Межпредметные связи**

Знания, полученные при изучении курса «Программирование на языке Pascal », учащиеся могут использовать при создании собственных программ по определенной тематике, для решения задач из различных областей знаний – математике, физике, химии, биологии и др. Знания и умения, приобретенные в результате освоения данного курса, являются фундаментом для дальнейшего мастерства в области программирования.

## **МЕТОДИЧЕСКАЯ ЧАСТЬ**

Основной тип занятий – практикум. Большинство заданий курса выполняется с помощью персонального компьютера и необходимых программных средств.

Занятия включают лекционную и практическую часть. Практическая часть курса реализуется через классно-урочную систему. Важной составляющей каждого урока является самостоятельная работа учащихся. Тема урока определяется приобретаемыми навыками. В каждом уроке материал излагается следующим образом: повторение основных понятий и методов работы с ними, разбор новой темы, основные приемы работы (самостоятельное выполнение заданий для получения основных навыков работы), упражнения для самостоятельного выполнения.

Теоретическая и практическая части курса изучаются параллельно, чтобы сразу же закреплять теоретические вопросы на практике.

В ходе обучения учащимся периодически предлагаются короткие (5-10 мин) контрольные работы на проверку освоения изученных способов действий. Проводятся краткие срезовые работы (тесты, творческая работа) по определению уровня знаний учеников по данной теме. Выполнение контрольных способствует быстрой мобилизации и переключению внимания на осмысливание материала изучаемой темы. Кроме того, такая деятельность ведет к закреплению знаний и служит регулярным индикатором успешности образовательного процесса.

Регулярное повторение способствует закреплению изученного материала. Возвращение к ранее изученным темам и использование их при изучении новых тем способствуют устранению весьма распространенного недостатка – формализма в знаниях учащихся – и формируют научное мировоззрение учеников.

### **Планируемые результаты элективного курса**

В рамках данного курса учащиеся овладевают следующими знаниями, умениями и способами деятельности:

- знают роль программного обеспечения и его виды;
- у учащихся сформировано целостное представление об организации данных для эффективной алгоритмической обработки;
- знают основные алгоритмические конструкции и правила их записи, знакомы с основными способами организации данных;
- умеют составлять и записывать алгоритмы с использованием соответствующих алгоритмических конструкций;
- умеют распознавать необходимость применения той или иной алгоритмической конструкции при решении задачи;
- умеют организовывать данные для эффективной алгоритмической обработки;
- умеют разрабатывать алгоритмы и реализовывать их на языке программирования Pascal;



- умеют осуществлять отладку и тестирование программы.

### **Форма контроля за уровнем достижения учащихся**

Предметом диагностики и контроля являются составленные алгоритмы и программы на языке программирования Pascal к предложенным задачам.

Оценка имеет различные способы выражения – устные суждения педагога, письменные качественные характеристики, систематизированные по заданным параметрам аналитические данные, в том числе и рейтинги. Оценке подлежит в первую очередь уровень достижения учеников минимально необходимых результатов, обозначенных в целях и задачах курса.

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.

В течение всего курса проводятся контрольные срезы и выставляются баллы за решенные задачи. По окончании курса **зачет** получают те учащиеся, у которых сумма баллов за учебный курс составляет не менее 60% от всей суммы баллов.

## **СОДЕРЖАНИЕ**

### **Часть 1. Теоретическая**

#### **1. Язык программирования Pascal**

Текстовый редактор языка. Основы языка. Структура программы. Основные математические функции.

#### **2. Ввод и вывод данных. Линейный алгоритм**

Форматы вывода. Составление линейных алгоритмов.

#### **3. Условный оператор**

Условный оператор. Оператор выбора.

#### **4. Алгоритмы с повторениями**

Цикл с параметром FOR . Циклы While и Repeat. Вложенные циклы.

#### **5. Массивы**

Понятие массива. Двумерные массивы.

#### **6. Графика в Pascal**

Графика.

### **Часть 2. Практическая**

1. Язык программирования Pascal. Текстовый редактор языка.
2. Структура программы. Разделы описания.
3. Основные математические функции. Моя первая программа.

4. Ввод и вывод данных. Форматы вывода.
5. Составление линейных алгоритмов. Комментарии в программе.
6. Составление линейных алгоритмов с использованием арифметических операций.
7. Составление линейных алгоритмов с использованием основных функций.
8. Условный оператор. Структура условного оператора.
9. Условный оператор. Простые условия.
10. Условный оператор. Составные условия.
11. Операторные скобки.
12. Составной оператор.
13. Оператор выбора.
14. Алгоритмы с повторениями.
15. Цикл с параметром.
16. Цикл с предусловием.
17. Цикл с постусловием.
18. Вложенные циклы.
19. Понятие массива. Ввод и вывод элементов массива.
20. Задание массива в разделе констант.
21. Поиск в массиве элементов с заданными свойствами.
22. Поиск максимального (минимального) элемента массива.
23. Двумерные массивы: описание, ввод и вывод массивов по строкам.
24. Графика. Графические примитивы.
25. Структура графической программы. Прямые линии. Прямоугольники.
26. Графика. Окружность. Эллипс. Дуга. Сектор.
27. Графика. Закрашивание.
28. Построение графиков на экране.
29. Построение поверхностей.

#### Учебно-тематическое планирование

№ п/п	Тема	Количество часов
1	<b>Язык программирования Pascal</b>	<b>5</b>
	Техника безопасности. Текстовый редактор языка. Основы языка. Структура программы. Основные математические функции. <i>Практическая работа №1.</i> Язык программирования Pascal . Текстовый редактор языка. <i>Практическая работа №2 .</i> Структура программы. Разделы описания.	1
		1
		1
		1

	Практическая работа №3. Основные математические функции. Первая программа. Практическая работа №4. Ввод и вывод данных. Форматы вывода.	1
<b>2</b>	<b>Ввод и вывод данных. Линейный алгоритм</b>	<b>4</b>
	Форматы вывода. Составление линейных алгоритмов. Практическая работа №5 . Составление линейных алгоритмов. Комментарии в программе. Практическая работа №6. Составление линейных алгоритмов с использованием арифметических операций. Практическая работа №7. Составление линейных алгоритмов с использованием основных функций.	1
		1
		1
		1
<b>3</b>	<b>Условный оператор</b>	<b>7</b>
	Условный оператор. Оператор выбора. Практическая работа №8. Условный оператор. Структура условного оператора. Практическая работа №9. Условный оператор. Простые условия. Практическая работа №10. Условный оператор. Составные условия. Практическая работа №11. Операторные скобки. Практическая работа №12. Составной оператор. Практическая работа №13. Оператор выбора.	1
		1
		1
		1
		1
		1
		1
<b>4</b>	<b>Алгоритмы с повторениями</b>	<b>6</b>
	Цикл с параметром FOR. Циклы While и Repeat. Вложенные циклы. Практическая работа №14. Алгоритмы с повторениями. Практическая работа №15. Цикл с параметром. Практическая работа №16. Цикл с предусловием. Практическая работа №17. Цикл с постусловием. Практическая работа №18. Вложенные циклы.	1
		1
		1
		1
		1
		1
<b>5</b>	<b>Массивы</b>	<b>6</b>
	Понятие массива. Двумерные массивы. Практическая работа №19. Понятие массива. Ввод и вывод элементов массива. Практическая работа №20. Задание массива в разделе констант. Практическая работа №21. Поиск в массиве элементов с заданными свойствами. Практическая работа №22. Поиск максимального (минимального) элемента массива. Практическая работа №23. Двумерные массивы: описание, ввод и вывод массивов по строкам.	1
		1
		1
		1
		1
		1
<b>6</b>	<b>Графика в Pascal</b>	<b>7</b>
	Графика.	1

<i>Практическая работа №24.</i> Графика. Графические примитивы. <i>Практическая работа №25.</i> Структура графической программы. Прямые линии. Прямоугольники. <i>Практическая работа №26.</i> Графика. Окружность. Эллипс. Дуга. Сектор. <i>Практическая работа №27.</i> Графика. Закрашивание. <i>Практическая работа №28.</i> Построение графиков на экране. <i>Практическая работа №29.</i> Построение поверхностей.	1
	1
	1
	1
	1
<b>ИТОГО:</b>	<b>34</b>

#### **Материально-техническое обеспечение :**

1. Компьютерный класс из 13 персональных компьютеров с операционной системой Windows-2007 и программным обеспечением Microsoft Office, Pascal.
2. Локальная компьютерная сеть;
3. Глобальная сеть Интернет;
4. Видео-проектор, экран.

#### **СПИСОК ЛИТЕРАТУРЫ**

1. <http://books.kulichki.ru/data/pascal/pas1/> Полный обучающий курс Турбо Паскаль.
2. <http://ips.ifmo.ru/courses/pascal/> Курс лекций «Язык программирования Pascal».
3. [http://www.gmcit.murmansk.ru/text/information\\_science/profile/methodic/pascal/pascal.html](http://www.gmcit.murmansk.ru/text/information_science/profile/methodic/pascal/pascal.html)  
- 40 уроков по Pascal .
4. Ушаков Д.М., Юркова Т.А. Паскаль для школьников. – СПб.: Питер, 2006. – 256 с.: ил.
5. Житкова О.А., Кудрявцева Е.К. Алгоритмы и основы программирования. М.: «Интеллект-центр», 2001.
6. Житкова О.А., Кудрявцева Е.К. Справочные материалы по программированию на языке Pascal . М.: «Интеллект-центр», 2001.
7. Златопольский Д.М. Я иду на урок информатики. М.: «Первое сентября», 2001.
8. Тимофеевская М. Изучаем программирование. Санкт-Петербург, «Питер», 2002.

## МОДУЛЬ «3D МОДЕЛИРОВАНИЕ»

### ПОЯСНИТЕЛЬНАЯ ЗАПИСКА

Учебный курс «3D-моделирование и анимация» входит в образовательную область «информатика». Он включает 18 часов аудиторных занятий и (при возможности) самостоятельную работу учащихся. Курс может быть использован для профильной подготовки учащихся в классах физико-математического, информационно-технологического, социально-гуманитарного и др. профилей.

Предметом изучения являются принципы и методы создания и анимации трехмерных моделей с помощью программы *Gmax*, которая является упрощенной версией профессиональной программы *3ds max*.

Целесообразность изучения данного курса определяется быстрым внедрением цифровой техники в повседневную жизнь и переходом к новым технологиям обработки информации. Учащиеся получают начальные навыки трехмерного моделирования и анимации, которые повышают их подготовленность к жизни в современном мире.

#### Цели курса:

- познакомить учащихся с современными принципами и методами создания 3D-моделей, основанных на использовании векторной графики;
- развить творческие и дизайнерские способности учащихся.

#### Задачи курса: научить школьников

- создавать трехмерные модели в *Gmax*;
- использовать программу *Gmax* для создания анимационных роликов.

Данный курс имеет выраженную практическую направленность, которая и определяет логику построения материала учебных занятий.

Основной формой обучения является практикум. Для нормальной работы необходим персональный компьютер (один на каждого ученика) и программа *Gmax*, которая распространяется бесплатно и может быть загружена с сайта <http://www.turbosquid.com>.

Для поддержки курса автором разработано электронное учебное пособие в формате СНМ, которое содержит теоретический материал и задания для выполнения практических работ. Оно используется во время уроков для самостоятельной работы и в качестве справочника. Это позволяет успешно организовывать занятия в группах, в которых есть ученики с разным темпом усвоения материала.

Знания, полученные при изучении курса «3D-моделирование и анимация», учащиеся могут применить для подготовки качественных иллюстраций к докладам и мультимедийным разработкам по различным предметам — математике, физике, химии, биологии и др. Анимационные ролики могут быть также использованы при создании *Web*-страниц. Трехмерное моделирование служит основой для изучения систем виртуальной реальности.

#### Ожидаемые результаты

В рамках данного курса учащиеся получают следующие знания и умения:

- понимают принципы создания и редактирования трехмерных моделей;
- понимают принципы работы с временной шкалой;
- умеют создавать анимационные ролики;
- умеют использовать звуковые файлы для сопровождения анимации.

#### Формы подведения итогов

Текущий контроль уровня усвоения материала осуществляется по результатам выполнения учащимися практических заданий на каждом уроке. В конце курса каждый учащийся выполняет индивидуальный проект в качестве зачетной работы. На последнем занятии проводится конференция, на которой учащиеся представляют свои работы и обсуждают их.

## **СОДЕРЖАНИЕ ИЗУЧАЕМОГО КУРСА**

### **Раздел 1. Введение**

#### **Тема 1. Знакомство с *Gmax***

Изучается расположение панелей программы *Gmax* и основные приемы работы с готовой сценой (режимы просмотра, рендеринг, просмотр анимации).

#### **Тема 2. Простейшие объекты (примитивы)**

Рассматриваются 3D-примитивы (куб, сфера, цилиндр и т.д.) и методы их перемещения, вращения, масштабирования, клонирования.

#### **Тема 3. Преобразования**

Изучаются сплайны и построение 3D-фигур на основе сплайнов (вращение, лофтинг). Вводится понятие модификатора и стека модификаторов.

### **Раздел 2. Сеточные модели**

#### **Тема 1. Сетки и их элементы**

Учащиеся знакомятся с понятием сеточной модели и видами сеточных моделей (грани, полигоны, лоскуты).

#### **Тема 2. Полигональные сетки**

Изучаются методы работы с полигональными сетками на уровне подобъектов (вершины, ребра, границы, полигоны, элементы).

#### **Тема 3. Особенности других типов сеток**

Рассматриваются особенности других типов сеток (грани, лоскуты) и методы работы с ними.

### **Раздел 3. Материалы и рендеринг**

#### **Тема 1. Материалы**

Изучаются методы создания и редактирования материалов (простые и многокомпонентные материалы, свойства материалов, текстурные карты). Учащиеся знакомятся с настройкой наложения текстур на криволинейные объекты (*UVW*-развертки).

#### **Тема 2. Освещение и рендеринг**

Изучаются типы источников света, камеры и настройка параметров рендеринга с помощью программы *YafRay*.

### **Раздел 4. Анимация**

#### **Тема 1. Ключевые кадры**

Изучаются методы создания 3D-анимации на основе автоматической расстановки ключевых кадров. Учащиеся знакомятся с понятиями контроллера и ограничителя.

#### **Тема 2. Связанные цепочки**

Рассматриваются приемы анимации на основе связанных цепочек объектов (методы прямой и обратной кинематики). Изучается анимация сеточных моделей с помощью скелетов (*bones*).

#### **Тема 3. *MAXScript***

Учащиеся знакомятся с языком программирования *MAXScript* и примерами его эффективного использования при построении сложных моделей и анимации.

### **Раздел 5. Выполнение проекта**

В течение 4-х занятий учащиеся выполняют проект на выбранную тему.

На последнем занятии учащиеся обсуждают все выполненные работы на конференции.

### УЧЕБНО-ТЕМАТИЧЕСКИЙ ПЛАН

№	Раздел (тема)	Кол-во часов	Виды занятий	
			Теория	Практика
1. Введение				
1.1	Знакомство с Gmax	1	1	
1.2	Примитивы	1		1
1.3	Преобразования	2		2
2. Сеточные модели				
2.1	Сетки и их элементы	1		1
2.2	Полигональные сетки	2		2
2.3	Особенности других типов сеток	1		1
3. Материалы и рендеринг				
3.1	Материалы	1		1
3.2	Освещение и рендеринг	1		1
4. Анимация				
4.1	Ключевые кадры	2		2
4.2	Связанные цепочки	1		1
4.3	MAXScript	1		1
5. Выполнение проекта		4		4
Всего часов:		18	1	17

ИН

### ФОРМАЦИОННОЕ ОБЕСПЕЧЕНИЕ КУРСА

#### Интернет-ресурсы

Программы, утилиты для Gmax

1. [www.turbosquid.com/gmax](http://www.turbosquid.com/gmax) — официальный сайт поддержки *Gmax*;
2. [www.fsalpha.com/gmax\\_tips](http://www.fsalpha.com/gmax_tips) — отсюда также можно скачать саму программу *Gmax* и всю документацию к ней;
3. [www.yafRay.org](http://www.yafRay.org) — сайт разработчиков бесплатной программы рендеринга **YafRay**;
4. [www.geocities.com/kam\\_b\\_lai](http://www.geocities.com/kam_b_lai) — сайт автора программы-скрипта **YAGMax** — *Yet Another GMax Exporter for YafRay*;
5. [www.windyweather.net/WW/max/gmax](http://www.windyweather.net/WW/max/gmax) — инсталляторы и уроки по *Gmax*;
6. [www.scenery.org/design\\_utilities\\_gmax.htm](http://www.scenery.org/design_utilities_gmax.htm) — набор утилит *Gmax Render Kit* для рендеринга изображений в *Gmax*;
7. [www.yayap.com/rendertut.html](http://www.yayap.com/rendertut.html) — программа-скрипт от Xzzy для преобразования моделей *Gmax* в формат *YafRay*;
8. [www.knochlet-development.de.tt](http://www.knochlet-development.de.tt) — *Gmax Render Suite* — пакет для рендеринга в *Gmax*;
9. [www.davidhsmith.net/gmaxindex.htm](http://www.davidhsmith.net/gmaxindex.htm) — страница *David Smith*, посвященная *Gmax* (программы, утилиты, уроки).

## Уроки по Gmax

1. [www.scenery.org/tutorials.htm](http://www.scenery.org/tutorials.htm) — уроки по *Gmax*;
2. <http://www.halomods.com> — основы *Gmax* от *SlashOx*;
3. [www.fileplanet.com/122467/120000/fileinfo/Gmax-Tutorial-Introduction](http://www.fileplanet.com/122467/120000/fileinfo/Gmax-Tutorial-Introduction) — введение в *Gmax*;
4. [www.windyweather.net/WW/max/gmax](http://www.windyweather.net/WW/max/gmax) — инсталляторы и уроки по *Gmax*;
5. [www.vmbollig.de/msts/tut\\_en/index.html](http://www.vmbollig.de/msts/tut_en/index.html) — уроки по *Gmax* от *Volker*;
6. [takeoff.to/landing](http://takeoff.to/landing) — уроки по *Gmax* от *Cris* (самолеты);
7. [www.fred-hsu.com/there/developer/index.html](http://www.fred-hsu.com/there/developer/index.html) — уроки от *Freddie*;
8. [www.simviation.com/gryphon/tutorials/gMax\\_00.htm](http://www.simviation.com/gryphon/tutorials/gMax_00.htm) — руководство по *Gmax* «для полных идиотов»;
9. [airspace.uhk.cz/mlk/msfs/tutorials/gmax](http://airspace.uhk.cz/mlk/msfs/tutorials/gmax) — применение текстур в *Gmax*, руководство от *Milan Lisner*;
10. [www.oregon-coast.net/Tutorials/index.htm](http://www.oregon-coast.net/Tutorials/index.htm) — уроки от *Oregon Coast*;
11. [cloud.prohosting.com/talone/gmax/tute/tutorials.html](http://cloud.prohosting.com/talone/gmax/tute/tutorials.html) — уроки от *Taelon*;
12. [members.fortunecity.com/foot\\_fall/the\\_basics.htm](http://members.fortunecity.com/foot_fall/the_basics.htm) — уроки от *LeJohn*;
13. [www.aerodynamika.com/gmaxtut/gmax000.htm](http://www.aerodynamika.com/gmaxtut/gmax000.htm) — разработка модели самолета;
14. [www.auran.com/TRS2004/learning.htm](http://www.auran.com/TRS2004/learning.htm) — уроки по созданию трехмерной модели паровоза для игры *Trainz Railroad Simulator*.
15. <http://home.sprynet.com/~drawlins> — анимация самолета;
16. <http://world-editor-tutorials.thehelper.net/magos.php> — уроки моделирования от *Magos*;
17. <http://www.studio-erebus.com/studio/tutorials/2003/uvw/uvw-dice.html> — использование текстур, модификатор *UVW mapping*;
18. [www.44090digitalmodels.co.uk/](http://www.44090digitalmodels.co.uk/) — уроки с сайта *44090 Digital Models*;
19. [www.angelfire.com/ma4/molkien/Gmax/Home.html](http://www.angelfire.com/ma4/molkien/Gmax/Home.html) — моделирование каменной скалы;
20. [www.modport.co.uk/index.php?showtopic=4752](http://www.modport.co.uk/index.php?showtopic=4752) — создание ландшафта;
21. [www.worldoftrainz-downloads.com/~garyp/gmaxtutorials.htm](http://www.worldoftrainz-downloads.com/~garyp/gmaxtutorials.htm) — уроки от *Garry* (мультитекстурные материалы, дом с окнами).

## Видеоуроки (на английском языке)

1. [www.quia.com/pages/scivistutorials.html](http://www.quia.com/pages/scivistutorials.html) — подборка видеоуроков по *Gmax* от *Roy Kimmins*;
2. [www.fsalpha.com/tutorials/tutorials.shtml](http://www.fsalpha.com/tutorials/tutorials.shtml) — видеоуроки от *FS Alpha Group*;
3. [youtube.com/watch?v=jY2yCX6yJac](http://youtube.com/watch?v=jY2yCX6yJac) — моделирование резервуара;
4. [youtube.com/watch?v=nE25b6DTgDk](http://youtube.com/watch?v=nE25b6DTgDk) — модель машины;
5. [youtube.com/watch?v=GVMSAtVX1u4](http://youtube.com/watch?v=GVMSAtVX1u4) — модель фюзеляжа самолета;
6. [youtube.com/watch?v=kgahFqcVFqU](http://youtube.com/watch?v=kgahFqcVFqU) — анимация ракеты;
7. [youtube.com/watch?v=tOtyd0jtGlg](http://youtube.com/watch?v=tOtyd0jtGlg) — работа с текстурами (на немецком языке);
8. [youtube.com/watch?v=kDboWZP\\_6Bg](http://youtube.com/watch?v=kDboWZP_6Bg) — модель космического корабля;
9. [youtube.com/watch?v=aUV3tl0-nJY](http://youtube.com/watch?v=aUV3tl0-nJY) — модель поршня;
10. [youtube.com/watch?v=HPUCcjZwEng](http://youtube.com/watch?v=HPUCcjZwEng) — модель лодки, часть 1а;  
часть 1b: [youtube.com/watch?v=VY1n0YYatpM](http://youtube.com/watch?v=VY1n0YYatpM);  
часть 2: [youtube.com/watch?v=Y25TJ9biGPI](http://youtube.com/watch?v=Y25TJ9biGPI);
11. [youtube.com/watch?v=7GyCBpVaOKA](http://youtube.com/watch?v=7GyCBpVaOKA) — еще один вводный урок по *Gmax*.

## Основная литература

1. Клейтон Е. Крукс II, *Gmax: настольная книга*, М.: — Кудиц-Образ, 2004.

## Дополнительная литература



1. **Миловская О.С.** Самоучитель 3ds Max 9, СПб: - БХВ, 2007.
2. **Бондаренко С., Бондаренко М.** Видеосамоучитель 3ds Max. – СПб: Питер, 2007.
3. **Козин М.** 3ds Max 9 для начинающих, , СПб: - БХВ, 2007.
4. **Мааров М.** Эффективная работа в 3ds Max 9. – СПб: Питер, 2007.
5. **Фокс Б.** Анимация в 3ds max 6: от замысла до создания мультфильма. СПб: Вильямс, 2005.

#### **Литература для школьников**

1. **Поляков К.Ю.** Уроки по 3D Gmax. Электронное учебное пособие, 2008.

## **МОДУЛЬ «ОЛИМПИАДНОЕ ПРОГРАММИРОВАНИЕ»**

### **ПОЯСНИТЕЛЬНАЯ ЗАПИСКА**

Программирование занимает значительное место в системе подготовки профессионалов в области информационных технологий. Уникальная ценность программирования как учебного вида деятельности состоит в том, что программирование - это борьба со сложностью. Как создать сложную систему, как довести её до работоспособного состояния? Эта борьба имеет свои особенности, свою специфику, которых нет ни в одном предмете. Она (борьба со сложностью) требует от интеллекта развитой аналитики, особой интуиции, умения предвидения совершаемых действий, рациональности и строгости мышления.

Школьники, умеющие составить алгоритм и написать по нему программу, обычно значительно легче овладевают и пользовательскими навыками, так как понимают механизм управления компьютером. Они лучше успевают и по другим предметам, поскольку культура их мышления выше, а ассортимент методов выполнения различных работ богаче. В связи с этим представляется достаточно важным привить учащимся навыки алгоритмического мышления.

### **Направленность программы**

Программа имеет техническую направленность, в связи с этим рассматриваются два актуальных аспекта изучения.

1. Технологический. Содержание программы рассматривается как средство формирования образовательного потенциала, позволяющего развивать наиболее передовые на сегодняшний день технологии - информационные, интегрирующие в себе науку, технологию, инженерное дело, математику (Science Technology Engineering Mathematics - STEM)
2. Общеобразовательный. Содержание программы рассматривается как средство развития основных познавательных процессов, умения анализировать, выявлять сущности и отношения, описывать планы действий и делать логические выводы, опираясь на такие дисциплины, как механика, теория управления, схемотехника, программирование, теория информации.

### **Актуальность программы**

Совершенствование технологических и программных средств привело к снижению количества часов, отводимых для изучения программирования в Программе среднего общего образования по информатике. Современные визуальные и

мультимедийные пользовательские среды являются теми конкурентами, которые вытесняют разработку программ из сферы интересов школьников. Для работы за компьютером для поиска информации в сети пользователь имеет простые инструменты, не требующие мыслительных усилий при применении. Как следствие, в большинстве школ отсутствует системная работа по подготовке обучающихся 7-11 классов к олимпиадам высокого уровня по программированию.

Дополнительная общеобразовательная общеразвивающая программа «Олимпиадное программирование» обеспечивает углубленное изучение языков программирования, нацелена на формирование математического аппарата описания и построения процессов обработки информации, в том числе человеком и технологическим устройством, создания и исследования числовых и нечисловых математических моделей.

**Педагогическая целесообразность программы** состоит в том, что индустрия программирования остаётся важнейшей в мире. Качество деятельности предприятий, их устойчивость зависят от программного обеспечения, и здесь никаких изменений не предвидится, разве что программные продукты будут играть всё большую роль.

### **Новизна программы**

Программой предусмотрены новые методики преподавания, в том числе - гибридное обучение; нововведения в формах диагностики и подведения итогов реализации программы, выполняемые в формате Всероссийских и международных олимпиад по программированию.

### **Цели программы**

Поиск новых эффективных средств развития у одаренных учащихся алгоритмического и операционального мышления.

Углубление и расширение знаний, относящихся к построению и описанию объектов и процессов, позволяющих осуществить их программное моделирование.

Подготовка учащихся к успешному участию в конкурсах и олимпиадах по программированию высокого уровня.

Подготовка базы для последующего профессионального изучения программирования в высшей школе.

Освоение навыков высокой эффективности деятельности.

Развитие интеллектуального, эмоционального и духовного интеллектов.

Обучение учащихся современным психологическим методам и стратегиям развития личностных качеств.

### **Задачи программы**

#### **1. Обучающие:**

- формирование умений в области создания текстов программ для персональных компьютеров на основе глубоких знаний среды программирования и языков программирования;
- расширение и углубление представлений об автоматизированной обработке информации;
- освоение методов программирования: выдвижение и обоснование идеи решения задачи, структурирование этой идеи, формализация элементов полученной структуры средствами выбранного языка, анализ результатов решения задачи при различных значениях исходных данных;
- овладение навыками публичного выступления;
- овладение приёмами аутогенной тренировки.

#### **2. Воспитывающие:**

- формирование определенного мировоззрения, противодействующего терроризму и экстремизму, связанного с устоями и обычаями, национальными и культурными традициями, историей региона, межнациональной и межрелигиозной толерантностью;
- восприятие системы ценностей, принципов, правил, стереотипов информационного общества;
- освоение информационной культуры: ответственного отношения к информации с учетом правовых и этических аспектов ее распространения, избирательного отношения к полученной информации;
- освоение психологических основ эффективного общения;
- формирование потребности в самостоятельном приобретении и применении знаний из дополнительных источников.

#### **3. Развивающие:**

- формирование операционного мышления, направленного на выбор оптимальных решений;

- развитие способностей по самостоятельному приобретению знаний, умений, навыков, ускорение процесса перехода от обучения к научению, самообучению - наивысшей ступени образовательного процесса;
- развитие способностей эффективной работы в условиях ограничений;
- развитие умений эффективного использования возможностей информационной среды, защиты от негативных воздействий;
- формирование способностей выдвигать и доказывать гипотезы опытным путем, разрабатывать стратегию решения, прогнозировать результаты своей деятельности, анализировать и находить рациональные способы решения задачи путем оптимизации, детализации созданного алгоритма;
- развитие способности к самоанализу, самопознанию;
- освоение психологических технологий, направленных на развитие оптимизма, вдохновения, стремления к победе;
- формирование навыка рефлексивной деятельности.
- **Ожидаемые результаты**
- Основным результатом обучения является достижение высокой информационно-коммуникационной компетентности учащегося в области программирования на языке
- Обязательные результаты изучения курса приведены в разделе «Содержание программы».
- Рубрика «Знать/понимать» включает требования к учебному материалу, который усваивается и воспроизводится учащимися. Выпускники должны понимать смысл изучаемых понятий, принципов и закономерностей.
- Рубрика «Уметь» включает требования, основанные на более сложных видах деятельности, в том числе творческой: создавать объекты, оперировать ими, оценивать числовые параметры процессов, приводить примеры практического использования полученных знаний, осуществлять самостоятельный поиск учебной информации.

Уровень предъявления материала обеспечивает учащимся возможность изучить дискретные математические структуры, такие, как графы и утверждения в логике.

#### **Учащиеся должны знать:**

- логику предикатов, бинарные отношения и их виды;
- элементы теории отображений и алгебры подстановок;
- основы алгебры вычетов и их приложение к простейшим криптографическим шифрам;
- метод математической индукции;
- алгоритмическое перечисление основных комбинаторных объектов;

- основы теории графов;
- элементы теории автоматов.

#### **Учащиеся должны уметь:**

- выполнять операции над предикатами;
- исследовать бинарные отношения на заданные свойства;
- выполнять операции над отображениями и подстановками;
- выполнять операции в алгебре вычетов;
- применять простейшие криптографические шифры для шифрования текстов;
- генерировать основные комбинаторные объекты на языке программирования
- находить характеристики графов;

#### **Формы занятий, используемые при изучении данного курса:**

- лекционная;
- индивидуальная работа;
- групповая работа;
- индивидуальная консультация;
- групповая консультация;

- контрольная работа.

#### **Тема .1. Теория множеств.**

*Теория.* Вводное занятие. Линейное представление НОД. Основная теорема арифметики. Исследование алгоритма факторизации на эффективность. Алгоритм Ферма. Формулы для нахождения простых чисел. Экспоненциальная формула простых чисел Мерсена. Числа Ферма. Исследование составных чисел Ферма  $F(1945)$  и  $F(5)$ . Исследование чисел Ферма вида  $F(6)$ ,  $F(7)$ ,  $F(13)$ ,  $F(16)$ . Теорема Морхеда и Вестерна. Сравнения и их свойства. Кольцо классов вычетов по модулю. Сравнение первой степени. Теорема Эйлера и Ферма. Полная и приведенная формула вычетов. Применение теорем Эйлера и Ферма к решению задач. Понятие о системе остаточных классов. Китайская теорема об остатках.

*Практика.* Решение задач на теорию множеств.

*Электронное обучение.* Дистанционная подготовка по теме «Теория множеств». Дистанционное решение задач по теме «Теория множеств».

*Форма подведения итогов:* контрольная работа.

## **Тема .2. Основы теории графов.**

*Теория.* Комбинаторные задачи. Решение комбинаторных уравнений и доказательство тождеств. Бином Ньютона. Рекурсивные и рекуррентные отношения. Матричное представление графов. Связность графа, свойства связности. Виды графов, их характеристика. Деревья, их виды и свойства. Алгоритмы оптимизации путей на графе: поиск в глубину и в ширину. Алгоритм Дейкстры. Метод Беллмана-Форда. Минимальные покрывающие деревья. Понятие векторного пространства. Алгоритм Крускала.

*Практика.* Решение задач на нахождение кратчайших путей. Нахождение фундаментального множества циклов.

*Электронное обучение.* Дистанционная подготовка по теме «Основы теории графов». Дистанционное решение задач по теме «Основы теории графов».

*Форма подведения итогов:* переводной экзамен.

## **РАЗДЕЛ . ПРАКТИКУМ ПО РЕШЕНИЮ ОЛИМПИАДНЫХ ЗАДАЧ**

**Учащиеся должны знать:**

- алгоритмы решения основных видов олимпиадных задач по программированию.

**Учащиеся должны уметь:**

- применять методы комбинаторики и дискретной математики при решении олимпиадных задач по программированию.

**Формы занятий,** используемые при изучении данного курса:

- индивидуальная работа;
- групповая работа;
- индивидуальная консультация;
- групповая консультация;
- контрольная работа.

### **Тема .1. Комбинаторика.**

*Теория.* Комбинаторные задачи.

*Практика.* Решение олимпиадных задач на комбинаторику.

*Электронное обучение.* Дистанционная подготовка по теме «Комбинаторика». Дистанционное решение задач по теме «Комбинаторика».

*Форма подведения итогов:* контрольная работа.

### **Тема .2. Алгоритмы на графах.**

*Теория.* Теория графов.

*Практика.* Решение олимпиадных задач на графы.

*Электронное обучение.* Дистанционная подготовка по теме «Алгоритмы на графах». Дистанционное решение задач по теме «Алгоритмы на графах».

*Форма подведения итогов:* контрольная работа.

## РАЗДЕЛ . СОВРЕМЕННОЕ ПРОГРАММИРОВАНИЕ

Уровень предъявления материала знакомит учащихся с различными методами решения олимпиадных задач по программированию.

### **Учащиеся должны знать:**

- однопроходные алгоритмы;
- рекурсивные алгоритмы;
- алгоритмы нестандартной обработки чисел;
- бинарный поиск, слияние и сортировку;
- вычислительную геометрию на плоскости;
- графы;
- комбинаторику;
- жадные алгоритмы.

### **Учащиеся должны уметь:**

- реализовывать вычислительные алгоритмы;
- оценивать сложность алгоритма и время работы программы
- разрабатывать тесты к задачам;
- обрабатывать исключительные ситуации;
- использовать стандартную библиотеку среды программирования.

### **Формы занятий, используемые при изучении данного курса:**

- лекционная;
- индивидуальная работа;
- групповая работа;
- индивидуальная консультация;
- групповая консультация;
- контрольная работа;
- экзамен.

### **Тема .1. Однопроходные алгоритмы.**

*Теория.* Попутные вычисления. Чтение и обработка символов. Линейный поиск подстроки в тексте.

*Практика.* Решение задач с использованием однопроходных алгоритмов.



*Электронное обучение.* Дистанционная подготовка по теме «Однопроходные алгоритмы». Дистанционное решение задач по теме «Однопроходные алгоритмы».

*Форма подведения итогов:* контрольная работа.

## **Тема .2. Рекурсия.**

*Теория.* Глубина рекурсии и общее количество рекурсивных вызовов. Быстрое возведение в степень. Рисование самоподобных ломаных.

*Практика.* Решение задач на рекурсию.

*Электронное обучение.* Дистанционная подготовка по теме «Рекурсия». Дистанционное решение задач по теме «Рекурсия».

*Форма подведения итогов:* Контрольная работа.

## **Тема .3. Нестандартная обработка чисел.**

*Теория.* Длинная целочисленная арифметика. Остатки от деления. Отслеживание циклических повторений

*Практика.* Решение задач по нестандартной обработке чисел.

*Электронное обучение.* Дистанционная подготовка по теме «Нестандартная обработка чисел». Дистанционное решение задач по теме «Нестандартная обработка чисел».

*Форма подведения итогов:* Контрольная работа.

## **Тема .4. Избранные олимпиадные задачи.**

*Теория.* Бинарный поиск. Слияние упорядоченных последовательностей. Применение сортировок. Вычислительная геометрия на плоскости. Точки, векторы, прямые, отрезки. Вычислительная геометрия на плоскости. Многоугольники. Полигоны. Вычислительная геометрия на плоскости. Окружности и круги. Графы и их применение. Комбинаторика. Перебор вариантов. Множества. Жадные алгоритмы. Игры двух лиц. Японский кроссворд.

*Электронное обучение.* Дистанционное решение олимпиадных задач.

*Практика.* Решение олимпиадных задач высокого уровня сложности.

	РАЗДЕЛ . Дискретная математика.	теория	практика	всего
1	Тема .1. Теория множеств.	3	3	6
2	Тема .2. Основы теории графов.	2	4	6
	РАЗДЕЛ. Практикум по решению олимпиадных задач.			
3	Тема .1. Комбинаторика.	2	4	6
4	Тема .2. Алгоритмы на графах.	2	4	6
	РАЗДЕЛ . Современное программирование.			
5	Тема 1. Однопроходные алгоритмы.	3	3	6
6	Тема 2. Рекурсия.	2	6	8
7	Тема 3. Нестандартная обработка чисел.	2	4	6
8	Тема 4. Избранные олимпиадные задачи.		8	8
	Итого	16	36	52

## СПИСОК ЛИТЕРАТУРЫ

### Список литературы, использованной при написании программы

1. Епанешников А.В. Программирование в среде Turbo Pascal 7.0 - М.: Диалог - МИФИ, 2013 г.
2. Вирт Н. Алгоритмы и структуры данных: Пер. с англ. - М.: Мир, 2013. - 360 с.: ил.
3. Лупанов О. Б. Курс лекций по дискретной математике. - М., 2012.
4. Иванов Б.Н. Дискретная математика. Алгоритмы и программы. Учеб. пособие М., Лаборатория Базовых Знаний, 2013. 288 с.
5. Фиофанова О.А. Психология взросления и воспитательные практики нового поколения: учеб. Пособие / - М.: Флинта: НОУ ВПО «МПСИ», 2012. - 120с.
6. Кови С. «7 навыков высокоэффективных людей. Мощные инструменты развития личности» - Альпина Паблишер, 2015
7. Мастерство коуча. 3D Коучинг Галата Ю. Издательство: Рига 2010 г.

### Список литературы, рекомендованной обучающимся

1. Культин Н.Б. Программирование в Turbo Pascal 7.0 и Delphi 5. СПб.: БХВ - Санкт-Петербург, 2013. 464 с.: ил.
2. Программирование на С и С++. Практикум: Учеб. Пособие для вузов/ А.В. Крячков, И.В. Сухина, В.К. Томшин; Под ред. В.К. Томшина - 2-е изд., исправ. - М.: Горячая линия - Телеком, 2014. - 344 с.:ил.
3. Манфред Кетс де Врис «Мистика лидерства. Развитие эмоционального интеллекта». 4-е издание Альпина Паблишер, 2012 г.
4. Кови Ш. «7 Навыков высокоэффективных подростков». - Добрая книга, 2014 г.

### Список литературы, рекомендованной родителям

1. Щебланова, Е. И. Неуспешные одаренные школьники / Е. И. Щебланова. - Москва: БИНОМ. Лаборатория знаний, 2011. - 245 с.
2. Дэниел Гоулман, Ричард Бояцис, Энни Макки «Эмоциональное лидерство: Искусство управления людьми на основе эмоционального интеллекта» Альпина Паблишер, 2013 г.
3. Ричард Темплар Правила самоорганизации: Как всё успевать, не напрягаясь Альпина Паблишер, 2013 г.

#### **СПИСОК ЭЛЕКТРОННЫХ ИСТОЧНИКОВ ИНФОРМАЦИИ**

8. Сайт дистанционной подготовки по информатике *informatics.mccme.ru*
9. Контеет по программированию «Poisk» *contest.stavpoisk.ru*
10. Электронная школа программиста *асmp. ru*
11. Белова Е.С., Щебланова Е.И. Социометрический статус и особенности общения школьников с высоким творческим потенциалом // Психологические исследования: электронный научный журнал. 2011. №2(16). С.12. URL: <http://psystudy.ru>