

**УТВЕРЖДАЮ**

Директор  
МКОУ «СОШ № 17»  
*А.А.А. / А.А.А. /*  
« 31 » 08 2022г.



**СОГЛАСОВАНО**

Руководитель центра  
образования цифрового и  
гуманитарного профилей  
«Точка роста»  
*А.А.А. / А.А.А. /*  
« 31 » 08 2022г.

**Дополнительная общеразвивающая программа**

## **«КИБЕРНЕТИКА»**

Составитель: Нога Сергей Алексеевич  
Направленность: информационно-технологическая  
Уровень: базовый  
Возраст обучающихся: 10–17 лет

МКОУ «СОШ №17»  
ст. Новомарьевская  
2022-2023 учебный год



## **Пояснительная записка**

### **Цель и задачи обучения курса дополнительного образования**

Целью изучения предмета «Информатика» является получение теоретических и практических знаний, умений и навыков в области современной информатики; формирование целостного мировоззрения, соответствующего современному уровню развития науки и общественной практики, учитывающего социальное, культурное, языковое, духовное многообразие современного мира.

Для достижения поставленной цели необходимо решение следующих задач:

- создание условий для развития интеллектуальных и творческих способностей учащихся, необходимых для успешной социализации и самореализации личности;
- формирование информационной и алгоритмической культуры;
- развитие алгоритмического мышления, необходимого для профессиональной деятельности в современном обществе; развитие умений составить и записать алгоритм;
- формирование умений формализации и структурирования информации, умения выбирать способ представления данных в соответствии с поставленной задачей;
- овладение важнейшими общеучебными умениями и универсальными учебными действиями (формулировать цели деятельности, планировать ее, находить и обрабатывать необходимую информацию из различных источников, включая Интернет и др.).

### **Место курса в учебном плане курса дополнительного образования**

Данная программа рассчитана на 36 учебных недель, 4 час в неделю, общее количество часов — 144. Рабочая программа может реализовываться с использованием электронного обучения (ЭО) и дистанционных образовательных технологий (ДОТ).

### **Предметные результаты освоения курса дополнительного образования**

Сформулированная цель реализуется через достижение образовательных результатов. Эти результаты структурированы по ключевым задачам дополнительного общего образования, отражающим индивидуальные, общественные и государственные потребности, и включают в себя предметные результаты:

- формирование представления об основных изучаемых понятиях курса;

- формирование информационной и алгоритмической культуры; формирование представления о компьютере как универсальном устройстве обработки информации; развитие основных навыков и умений использования компьютерных устройств;
- развитие алгоритмического мышления, необходимого для профессиональной деятельности в современном обществе; развитие умений составить и записать алгоритм для решения конкретной задачи;
- формирование умений формализации и структурирования информации, умения выбирать способ представления данных в соответствии с поставленной задачей, с использованием соответствующих программных средств обработки данных;
- формирование представления о том, что значит “программировать” на примере языка Scratch, формирование умения составлять сценарии проектов среды Scratch;
- знакомство с основными алгоритмическими структурами — линейной, условной и циклической;
- формирование умения тестировать и оптимизировать алгоритмы исполнителей.

## **Содержание курса дополнительного образования**

### **Программирование в Scratch (36 часов)**

Сегодня наука и технология развиваются столь стремительно, что образование зачастую не успевает за ними. Одним из самых известных неформальных способов организации внеучебной образовательной деятельности является метод проектов. Самым подходящим инструментом для организации такой деятельности является среда Scratch, которая есть серьезное и современное направление компьютерного дизайна и анимации. Овладев даже минимальным набором операций, самый неискущённый пользователь может создавать законченные проекты. Scratch - это самая новая среда, которая позволяет детям создавать собственные анимированные и интерактивные истории, презентации, модели, игры и другие произведения. Работа в среде Scratch позволяет, с одной стороны, организовать среду для самореализации и самоутверждения учащихся, и, с другой стороны, сформировать у них тягу к творчеству и знаниям и дать подходящие средства её реализации. Быть успешным в такой среде становится проще.

Scratch можно рассматривать как инструмент для творчества, оставив программирование на втором плане. Школьники могут сочинять истории, рисовать и оживлять на экране придуманных ими персонажей, учиться работать с графикой и звуком. Применений возможностям Scratch можно найти множество: в этой среде легко создавать анимированные открытки, мини-игры, мультфильмы. В результате выполнения простых команд может складываться сложная модель, в которой будут взаимодействовать множество объектов, наделенных различными свойствами. Начальный уровень

программирования настолько прост и доступен, что Scratch рассматривается в качестве средства обучения не только старших, но и младших школьников.

Когда ученики создают проекты в Scratch, они осваивают множество навыков 21 века, которые будут необходимы для успеха:

- творческое мышление,
- ясное общение,
- системный анализ,
- беглое использование технологий,
- эффективное взаимодействие,
- проектирование,
- умение обучаться и самообучаться,
- самостоятельное принятие решений.

Изучение Scratch может серьезно помочь школьникам освоить азы алгоритмизации и программирования, а полученные знания пригодятся для дальнейшего и более серьезного изучения программирования. Работа в среде Scratch ведется так же как средство подготовки учащихся к всевозможным конкурсам и выставкам по данной тематике, которые в настоящее время набирают большие обороты.

Изучение данного курса направлено на достижение *следующих целей*:

- ✓ формирование общеучебных умений и навыков на основе средств и методов информатики и ИКТ, в том числе овладение умениями работать с различными видами информации, самостоятельно планировать и осуществлять индивидуальную и коллективную информационную деятельность, представлять и оценивать ее результаты;
- ✓ пропедевтическое изучение понятий основного курса школьной информатики;
- ✓ воспитание ответственного и избирательного отношения к информации; развитие познавательных, интеллектуальных и творческих способностей учащихся,
- ✓ развитие способности к решению творческих задач,
- ✓ развитие самостоятельности.

Ключевым в программе является понятие «проектная научно-познавательная деятельность школьника», как совместная (с другими субъектами), так и самостоятельная деятельность с использованием методов научного исследования, ведущим мотивом которой является познавательный интерес. Проектная научно-познавательная деятельность не является самоцелью, но рассматривается как среда, в которой наиболее естественным образом раскрывается личностный потенциал школьника.

В этой связи **целями проектной научно-познавательной деятельности школьника** являются:

- ✓ развитие интеллектуальных, познавательных и творческих способностей школьника;
- ✓ развитие метапредметных умений (личностных, познавательных, коммуникативных, регулятивных);
- ✓ развитие способов мыслительной деятельности;
- ✓ формирование целостной картины мира и системного мышления на основе межпредметных связей.

Следует иметь в виду, что возрастные особенности младшего школьника не позволяют в полной мере реализовать проведение полноценных научных исследований. В то же время раннее включение в организованную специальным образом проектную деятельность творческого характера позволяет сформировать у школьника познавательный интерес и исследовательские навыки, которые в старшем возрасте пригодятся им для выполнения научно-познавательных проектов.

**Основная задача курса** – способствовать формированию у школьников информационной и функциональной компетентности, развитие алгоритмического мышления. Назначение курса – помочь детям узнать основные возможности компьютера и научиться им пользоваться в повседневной жизни. А так же:

- ✓ дать учащимся представление о современном подходе к изучению реального мира, о широком использовании алгоритмов и вычислительной техники в научных исследованиях;
- ✓ сформировать у учащихся умения владеть компьютером как средством решения практических задач;
- ✓ подготовка учеников к активной полноценной жизни и работе в условиях технологически развитого общества;
- ✓ создание условий для внедрения новых информационных технологий в учебно-воспитательный процесс школы;
- ✓ реализовать в наиболее полной мере возрастающий интерес учащихся к углубленному изучению программирования через совершенствование их алгоритмического и логического мышления;
- ✓ формирование знаний о роли информационных процессов в живой природе, технике, обществе;
- ✓ формирование знаний о значении информатики и вычислительной техники в развитии общества и в изменении характера труда человека;
- ✓ формирование знаний об основных принципах работы компьютера, способах передачи информации;
- ✓ формирование знаний об основных этапах информационной технологии решения задач в широком смысле;

- ✓ формирование умений моделирования и применения его в разных предметных областях;
- ✓ формирование умений и навыков самостоятельного использования компьютера в качестве средства для решения практических задач.

Реализация этих задач будет способствовать дальнейшему формированию взгляда школьников на мир, раскрытию роли информатики в формировании естественнонаучной картины мира, развитию мышления, в том числе формированию алгоритмического стиля мышления, подготовке учеников к жизни в информационном обществе.

### **Программирование в Python (58 часов)**

Элективный курс по информатике «Язык Python: основы программирования, реализация известных алгоритмов» Основы программирования на Python" представляет собой вводный курс по программированию, дающий представление о базовых понятиях структурного программирования (данных, операциях, переменных, ветвлениях в программе, циклах и функциях, массивах, файлах), о реализации с помощью Python наиболее известных в математическом обеспечении программирования алгоритмов:

- Алгоритм Евклида (нахождение наибольшего общего делителя)
- Вычисление факториала на языке программирования Python
- Двоичный (бинарный) поиск элемента в массиве
- Замена элементов в списке
- Перевод чисел из десятичной системы счисления в двоичную
- Решето Эратосфена - алгоритм определения простых чисел
- Сортировка выбором (поиск минимума и перестановка)
- Сортировка методом пузырька
- Сумма и произведение цифр числа
- Тестирование простоты числа методом перебора делителей
- Числа Фибоначчи (вычисление с помощью цикла while и рекурсии)

### **Основы нейротехнологий (50 часов)**

Раздел «Основы нейротехнологий» технической направленности базового уровня позволяет обучающимся овладеть основами нейротехнологий, созданием и конструированием человеко-машинных интерфейсов – технических устройств, управляемых биосигналами человека. Раздел предусматривает развитие творческих способностей обучающихся, их технических знаний, навыков, умений по двум направлениям: электроника и электрофизиология, помогает сориентироваться в выборе будущей профессии.

Актуальность раздела обусловлена развитием современных биологических, медицинских и инженерных технологий в области нейробиологии, нейрохирургии и нейроуправления, что требует воспитания технически грамотных людей.

Из вышеизложенного вытекает педагогическая целесообразность раздела – ориентация обучающихся на техническое творчество, профессиональное самоопределение, дальнейшее применение полученных начальных знаний, умений и навыков во время обучения в учреждениях среднего профессионального и высшего образования.

Новизна Программы заключается в знакомстве обучающихся с современными технологиями, приемами и способами работы с современным нейротехнологическим оборудованием, позволяющим исследовать и моделировать различные объекты и системы.

Данный раздел позволяет обучающимся познакомиться с основами нейротехнологий не только через теоретическое изучение, но, прежде всего, через практическую работу с приборами, улавливающими биосигналы человеческого тела и способными обрабатывать и трансформировать полученные данные в информацию для дальнейшего использования. Для реализации программы раздела используется набор-конструктор «Юный нейромоделист» ViTronics Lab, позволяющий считывать и визуализировать биосигналы человека посредством электромиограммы, электроэнцефалограммы, кожногальванической реакции пульса, благодаря этому у обучающихся есть возможность заниматься проектной работой в области медицины и инженерии.

**Цель раздела** – создание условий для развития инженернотехнических способностей обучающихся через изучение нейротехнологий, программирования, электроники.

#### **Задачи раздела**

##### **Обучающие:**

- ознакомить с историей развития отечественной и мировой техники, ее создателях, о различных направлениях изучения нейротехнологий,
- программирования;
- формировать целостную научную картину мира;
- обучать эффективной работе с технической литературой, интернетисточниками;
- ознакомить с основными принципами работы нейрокомпьютерных интерфейсов;
- формировать представления об основах электроники;
- обучать основам программирования.

##### **Развивающие:**

- развивать инженерное мышление, изобретательность, образное и пространственное восприятие;
- развивать мыслительные, творческие, коммуникативные способности обучающихся;

- развивать интерес к техническим знаниям;
- развивать интеллектуальные и практические умения, самостоятельно приобретать и применять на практике полученные знания;
- развивать умение осознанно ставить перед собой конкретную задачу и добиваться её выполнения.

#### **Воспитательные:**

- воспитывать устойчивый интерес к нейротехнологиям;
- воспитывать информационную культуру как составляющую общей культуры современного человека;
- формировать потребность в творческой деятельности, стремление к самовыражению через техническое творчество;
- формировать новаторское отношение ко всем сферам жизнедеятельности человека.

### **Планируемые результаты освоения Программы**

#### **По итогам обучающиеся будут знать:**

- технику безопасности и требования, предъявляемые к организации рабочего места;
- терминологию нейромоделирования;
- оборудование и инструменты, используемые в области нейротехнологий;
- основные сферы применения нейротехнологий;
- основные направления развития нейротехнологий;
- основы нейробиологии и нейрофармтехнологии;
- основные принципы работы нейрокомпьютерных интерфейсов;
- основные принципы работы электронных схем и систем управления объектами;
- основы электроники и программирования микроконтроллеров для решения задач из области нейротехнологий;
- основы прикладной математики и программирования человеко-машинных интерфейсов;
- основы робототехники и управления роботами с помощью нейроинтерфейсов и датчиков биосигналов;

#### **будут уметь:**

- пользоваться инструментами и оборудованием, используемыми в области нейромоделирования;
- грамотно использовать технические термины;
- составлять простые программы для решения задач из области нейротехнологий;
- читать технические рисунки, эскизы, чертежи, схемы;
- конструировать простейшие электронные схемы, использующие интерфейс «мозг-компьютер»;



- разрабатывать простейшие системы с использованием электронных компонентов и нейрокомпьютерных интерфейсов;
- программировать человеко-машинные интерфейсы;
- управлять роботами с помощью нейроинтерфейсов и датчиков биосигналов.

### Тематическое планирование

№	Тема	Кол – во часов
1.	Программирование в Scratch	36
2.	Программирование в Python	58
3.	Основы нейротехнологий	50
<b>Всего</b>		<b>144</b>

### Календарно-тематический план

№	Тема	Кол – во часов	Дата
1.	Введение. Что такое Scratch. Основные алгоритмические конструкции. Знакомство с интерфейсом программы Scratch.	1	
2.	Исполнитель Scratch, цвет и размер пера. Кейс 2. Рисуем разноцветные лужи и облака для прогулки кота. Сцена. Редактирование фона. Добавление фона из файла.	1	
3.	Основные инструменты встроенного графического редактора программной среды SCRATCH. Кейс 3. Свободное рисование Понятие спрайтов. Добавление новых спрайтов. Рисование новых объектов.	1	
4.	Алгоритм. Линейный алгоритм. Создание блок-схемы. Кейс 4. Запиши мой алгоритм! Синий ящик – команды движения. Темно-зеленый ящик – команды рисования.	1	
5.	Линейный алгоритм. Рисование линий исполнителем Scratch. Кейс 5. Что бывает полосатое? Фиолетовый ящик – внешний вид объекта. Оживление объекта с помощью добавления костюмов.	1	

6.	Линейный алгоритм. Исполнитель Scratch рисует квадраты и прямоугольники линейно. Желтый ящик – контроль. Лиловый ящик – добавление звуков.	1	
7.	Конечный цикл. Scratch рисует квадраты, линии. Использование в программах условных операторов.	1	
8.	Конечный цикл. Scratch рисует несколько линий и фигур. Копирование фрагментов программы. Кейс 6. Создай картинку из квадратов. Функциональность работы циклов. Цикличность выполнения действий в зависимости от поставленных условий.	1	
9.	Циклический алгоритм. Цикл в цикле. Зеленый ящик – операторы. Использование арифметических и логических блоков вместе с блоками управления.	1	
10.	Цикл в цикле. Повторение пунктирной линии с поворотом. Блок-схема цикла. Кейс 7. Мой необычный дом. События. Оранжевый ящик – переменные.	1	
11.	Бесконечный цикл. Анимация исполнителя Scratch на основе готовых костюмов. Списки.	1	
12.	Сцена как исполнитель. Создаем модель таймера. Голубой ящик – сенсоры. Ввод-вывод данных.	1	
13.	Одинаковые программы для нескольких исполнителей. Последовательность и параллельность выполнения скриптов.	1	
14.	Параллельное выполнение действий несколькими исполнителями Взаимодействие между спрайтами. Управление через обмен сообщениями.	1	
15.	Разбиение программы на части для параллельного выполнения исполнителями. Таймер. Кейс 8. Мини-проект «Смена времени суток».	1	

16.	Два исполнителя со своими программами. Мини-проект «Часы с кукушкой». Кейс 9. Мини-проект «Часы с кукушкой».	1	
17.	Алгоритмы с ветвлением. Условие ЕСЛИ.	1	
18.	Цикл с условием. Мини-проект «Шарики в лабиринте» Кейс 10. Мини-проект «Шарики в лабиринте»	1	
19.	Цикл с условием. Исполнитель определяет цвет. Сенсор «касается цвета»	1	
20.	Оператор случайных чисел.	1	
21.	Перемещение исполнителей между слоями.	1	
22.	Действия исполнителей в разных слоях. Кейс 10. Мини-проект «Дорога».	1	
23.	Взаимодействие исполнителей.	1	
24.	Последовательное выполнение команд исполнителями.	1	
25.	Программирование клавиш. Мини-проект «Лабиринт» Кейс 10. Игра «Лабиринт»	1	
26.	Управление событиями.	1	
27.	Координатная плоскость. Геометрические фигуры. Кейс 11. Исследование времени выполнения программ.	1	
28.	Координатная плоскость. Переменные.	1	
29.	Создание списков.	1	
30.	Кейс 12. Мини-проект «Викторина».	1	
31.	Использование подпрограмм.	1	
32.	Сообщество Scratch.	1	
33.	Отладка программ с ошибками.	1	
34.	Кейс 13. Итоговый проект.	1	
35.	Кейс 13. Итоговый проект.	1	

36.	Кейс 13. Итоговый проект.	1	
37.	История языков программирования.. Язык Python. Простейшие программы. Реализация вычислений и ветвлений.	8	
38.	Реализация циклических, вспомогательных алгоритмов алгоритмов. Рекурсия.	14	
39.	Словари. Массивы. Обработка массивов	16	
40.	Символьные строки. Обработка символьных строк.	10	
41.	Матрицы. Ввод, вывод, обработка матриц. Чтение и запись текстовых файлов.	8	
42.	Основы нейротехнологий	2	
43.	Электромиография в современной нейротехнологии	2	
44.	Электроэнцефалография в современной нейротехнологии	2	
45.	Фотоплетизмография в современной нейротехнологии	2	
46.	Электрокардиография в современной нейротехнологии	2	
47.	Диагностика электрической активности кожи в современной нейротехнологии	2	
48.	Программирование в Arduino IDE	2	
49.	Синтаксис и программа кода Типы данных и переменные	3	
50.	Математические операции. Массивы	2	
51.	Сравнения, условия и выбор. Циклы. Строки и массивы символов Функции. Объекты и классы	3	
52.	Основы робототехники и управления роботами с помощью нейроинтерфейсов и датчиков биосигналов	28	

## Содержание программы

История языков программирования. Компиляция и интерпретация.  
Знакомство с Python и средами программирования.  
Типы данных в программировании. Определение переменной.  
Ввод данных с клавиатуры.  
Логические выражения.  
Условный оператор. Инструкция if.  
Множественное ветвление.  
Цикл While и For.  
Строки как последовательности символов.  
Списки — изменяемые последовательности. Массивы.  
Основные задачи обработки массивов: поиск, сортировка, реверс, ...  
Введение в словари.  
Функции в программировании. Параметры и аргументы функций.  
Локальные и глобальные переменные. Процедуры.  
Файлы. Чтение текстового файла. Запись в файл.  
Алгоритм Евклида (нахождение наибольшего общего делителя)  
Вычисление факториала на языке программирования Python  
Двоичный (бинарный) поиск элемента в массиве  
Замена элементов в списке  
Перевод чисел из десятичной системы счисления в двоичную  
Решето Эратосфена - алгоритм определения простых чисел  
Сортировка выбором (поиск минимума и перестановка)  
Сортировка методом пузырька  
Сумма и произведение цифр числа  
Тестирование простоты числа методом перебора делителей  
Числа Фибоначчи (вычисление с помощью цикла while и рекурсии)

## Планируемые результаты обучения курса дополнительного образования

- отдельные способы планирования деятельности;
- составление плана предстоящего проекта в виде рисунка, схемы;
- составление плана предстоящего проекта в виде таблицы объектов, их свойств и взаимодействий;
- разбиение задачи на подзадачи;
- о требованиях к организации компьютерного рабочего места, соблюдать требования безопасности и гигиены в работе со средствами ИКТ;
- устройства компьютера;
- примеры информационных носителей.

### *уметь:*

- приводить простые жизненные примеры передачи, хранения и обработки информации в деятельности человека, в живой природе, обществе, технике;
  - различать программное и аппаратное обеспечение компьютера;
  - запускать программы из меню Пуск;
  - изменять размеры и перемещать окна, реагировать на диалоговые окна;
  - вводить информацию в компьютер с помощью клавиатуры и мыши;
  - применять текстовый редактор для набора, редактирования и форматирования простейших текстов;
  - иметь представление о способах кодирования информации;
  - применять простейший графический редактор для создания и редактирования рисунков;
  - выполнять вычисления с помощью приложения Калькулятор;
  - составлять план проекта, включая:
    - выбор темы;
    - анализ предметной области;
    - разбиение задачи на подзадачи;
  - проанализировать результат и сделать выводы;
  - найти и исправить ошибки;
  - подготовить небольшой отчет о работе;
  - публично выступить с докладом;
  - наметить дальнейшие пути развития проекта.
- 
- Важнейшими умениями/знаниями являются следующие:
  - умение пользоваться персональным компьютером и его периферийным оборудованием;
    - умение следовать требованиям техники безопасности, гигиены, эргономики и ресурсосбережения при работе со средствами информационных и коммуникационных технологий;
    - умение осуществлять взаимодействие посредством электронной почты, чата, форума;
    - умение искать информацию с применением правил поиска (построения запросов), в компьютерных сетях, некомпьютерных источниках информации (справочниках и словарях, каталогах, библиотеках) при выполнении заданий и проектов по различным учебным дисциплинам;
    - умение составлять сценарии проектов среды Scratch;
    - умение составлять алгоритмы, определять последовательность выполнения команд; использовать обширную библиотеку готовых сцен и исполнителей;
    - умение создавать линейные алгоритмы для исполнителя; умение создавать циклические и ветвящиеся алгоритмы;

- умение управлять одновременной работой нескольких исполнителей; умение передавать сообщения между исполнителями;
- умение тестировать и оптимизировать алгоритмы исполнителей.
- знать место языка Python среди языков программирования высокого уровня,
- знать особенности структуры программы, представленной на языке Python,
- иметь представление о модулях, входящих в состав среды Python?
- знать возможности и ограничения использования готовых модулей,
- иметь представление о величине, ее характеристиках,
- знать, что такое операция, операнд и их характеристики,
- знать принципиальные отличия величин, структурированных и не структурированных,
- иметь представление о таких структурах данных, как число, текст, кортеж, список, словарь,
- иметь представление о составе арифметического выражения;
- знать математические функции, входящие в Python,
- иметь представление о логических выражениях и входящих в них операндах, операциях и функциях,
- уметь записывать примеры арифметических и логических выражений всех атрибутов, которые могут в них входить,
- знать основные операторы языка Python, их синтаксис,
- иметь представление о процессе исполнения каждого из операторов,
- уметь разрабатывать программы обработки числовой и символьной информации,
- уметь разрабатывать программы (линейные, разветвляющиеся и с циклами),
- иметь представление о значении полноценных процедур и функций для структурно-ориентированного языка высокого уровня,
- знать правила описания процедур и функций в Python и построение вызова процедуры,
- знать принципиальные отличия между формальными, локальными и глобальными переменными,
- знать область действия описаний в процедурах,
- иметь представление о рекурсии, знать ее реализацию на Python,
- владеть основными приемами формирования процедуры и функции,
- знать, как с помощью Списков, определять в программе тип «массив», «матрица»

- знать свойства данных типа «массив», «матрица»
- уметь воспроизводить алгоритмы сортировки массивов и матриц, поиска в упорядоченном массиве, распространять эти алгоритмы на сортировку и поиск в нечисловых массивах
- уметь читать и записывать текстовые файлы в заданном формате.
- решать основные алгоритмические задачи, представленные в пояснительной записке, в среде Python.

## **Планируемые результаты**

### **Личностные результаты**

К личностным результатам освоения информационных и коммуникационных технологий как инструмента в учёбе и повседневной жизни можно отнести:

- критическое отношение к информации и избирательность её восприятия;
- уважение к информации о частной жизни и информационным результатам других людей;
- осмысление мотивов своих действий при выполнении заданий с жизненными ситуациями;
- начало профессионального самоопределения, ознакомление с миром профессий, связанных с информационными и коммуникационными технологиями.

**Метапредметными результатами** изучения курса «Программирование» являются формирование следующих универсальных учебных действий:

### **Регулятивные УУД:**

- планирование последовательности шагов алгоритма для достижения цели;
- поиск ошибок в плане действий и внесение в него изменений.

### **Познавательные УУД:**

- моделирование – преобразование объекта из чувственной формы в модель, где выделены существенные характеристики объекта (пространственно-графическая или знаково-символическая);
- анализ объектов с целью выделения признаков (существенных, несущественных);
- синтез – составление целого из частей, в том числе самостоятельное достраивание с восполнением недостающих компонентов;
- выбор оснований и критериев для сравнения, классификации объектов;



- подведение под понятие;
- установление причинно-следственных связей;
- построение логической цепи рассуждений.

**Коммуникативные УУД:**

- аргументирование своей точки зрения на выбор оснований и критериев при выделении признаков, сравнении и классификации объектов;
- выслушивание собеседника и ведение диалога;
- признание возможности существования различных точек зрения и права каждого иметь свою.