

МБОУ «Средняя общеобразовательная школа №4» городского округа Судак

Рассмотрено и одобрено  
На заседании ШМО  
Протокол № 1  
«31» 08 2022г.

СОГЛАСОВАНО  
Заместитель директора по УВР

«31» 08 2022г

УТВЕРЖДАЮ  
Директор МБОУ  
Ю.А.Собко  
Приказ № 31 от 31 08.2022г



## ДОПОЛНИТЕЛЬНАЯ ОБЩЕОБРАЗОВАТЕЛЬНАЯ ОБЩЕРАЗВИВАЮЩАЯ ПРОГРАММА

### «Программирование на Python» на 2022/2023 учебный год

Вид программы: **Авторская**  
Составитель: учитель информатики  
Козлов Сергей Владимирович

# 1. ПОЯСНИТЕЛЬНАЯ ЗАПИСКА

Увлекательное программирование на одном из самых популярных языков в мире.

Учимся кодить на Python — одном из самых популярных языков программирования в мире и сразу применяем знания на практике, создавая программы для учебы, развлечений и жизни.

Программа составлена в соответствии с нормативными документами:

- Законом Российской Федерации от 29 декабря 2012 г. №273–ФЗ «Об образовании в Российской Федерации»,
- Концепцией развития дополнительного образования детей, утвержденной распоряжением правительства Российской Федерации от 4 сентября 2014 г. № 1726-р,
- Приказом Министерства просвещения РФ от 09 ноября 2018 г. № 196 «Об утверждении Порядка организации и осуществления образовательной деятельности по дополнительным общеобразовательным программам»,
- Постановлением Главного государственного санитарного врача РФ от 4 июля 2014 г. № 41 «Об утверждении СанПин 2.4.4.3172-14 «Санитарноэпидемиологические требования к устройству, содержанию и организации режима работы образовательных организаций дополнительного образования детей»,
- Примерные требования к содержанию и оформлению образовательных программ дополнительного образования детей (Письмо Министерства образования от 18 июня 2003 г. МС 28-02-484\16),
- Типовым Положением об образовательном учреждении дополнительного образования детей, нормативными документами Министерства общего и профессионального образования РФ,
- Образовательная программа МБОУ СОШ№4» городского округа Судак.

**Новизна.** С первых занятий создаем игры и приложения, уже в процессе изучая нужные инструменты и термины. В формате игры проходим путь от собеседования в IT-компани до должности ведущего разработчика.

**Актуальность.** Через 2 месяца научимся писать код на Python, понимать его синтаксис и основные типы данных.

Через 5 месяцев научимся создавать приложения для компьютера и делать их удобными для пользователя.

В конце первого года научимся разрабатывать многоуровневые игры, используя профессиональные библиотеки Python

**Цели:**

- развитие общеучебных умений и навыков на основе средств и методов программирования и ИКТ, в том числе овладению умениями работать с различными видами информации, самостоятельно планировать и осуществлять индивидуальную и коллективную информационную деятельность, представлять и оценивать ее результаты;
- целенаправленному формированию таких общеучебных понятий, как «объект», «система», «модель», «алгоритм» и др.;

**Задачи:**

- создать условия для овладения основными универсальными умениями информационного характера, таких как постановка и формулирование проблемы; поиск и выделение необходимой информации, применение методов информационного поиска; структурирование и визуализация информации; выбор наиболее эффективных способов решения задач в зависимости от конкретных условий; самостоятельное создание алгоритмов деятельности при решении проблем творческого и поискового характера;

**Личностные результаты:**

- способность увязать учебное содержание с собственным жизненным опытом, понять значимость подготовки в области информатики и ИКТ в условиях развития информационного общества;

**Метапредметные результаты:**

- владение информационно-логическими умениями: определять понятия, создавать обобщения, устанавливать аналогии, классифицировать, самостоятельно выбирать основания и критерии для классификации, устанавливать причинно-следственные связи, строить логическое рассуждение, умозаключение (индуктивное, дедуктивное и по аналогии) и делать выводы;
- ИКТ-компетентность — широкий спектр умений и навыков использования средств информационных и коммуникационных технологий для сбора, хранения, преобразования и передачи различных видов информации, навыки создания личного информационного пространства (обращение с устройствами ИКТ; фиксация изображений и звуков; создание письменных сообщений; создание графических объектов; создание музыкальных и звуковых

**Предметные результаты:**

- формирование информационной и алгоритмической культуры; формирование представления о компьютере как универсальном устройстве обработки информации; развитие основных навыков и умений использования компьютерных устройств;
- формирование представления об основных изучаемых понятиях, таких как информация, алгоритм, модель, и их свойствах;
- развитие алгоритмического мышления, необходимого для профессиональной деятельности в современном обществе; развитие умений составлять и записывать алгоритм для конкретного исполнителя; формирование знаний об алгоритмических конструкциях, логических значениях и операциях; знакомство с одним из языков программирования и основными алгоритмическими структурами – линейной, условной и циклической;

## 2. ТЕМАТИЧЕСКОЕ ПЛАНИРОВАНИЕ

№	Тема	Количество часов
1	ОСНОВЫ ЯЗЫКА	4
2	УПРАВЛЯЮЩИЕ КОНСТРУКЦИИ	5
3	ФУНКЦИИ И МОДУЛИ	4
4	МОДУЛЬ TURTLE. МАТЕМАТИКА ДЛЯ РАЗРАБОТЧИКА	4
5	ОБЪЕКТНО-ОРИЕНТИРОВАННОЕ ПРОГРАММИРОВАНИЕ	7
6	ОСНОВЫ РАЗРАБОТКИ ИГР НА PYGAME	8
7	ИТОГОВОЕ ПОВТОРЕНИЕ	2

### 3. СОДЕРЖАНИЕ ПРОГРАММЫ

#### Раздел 1. Основы языка

##### Тема 1. ОСНОВЫ ЯЗЫКА(4 часа)

###### Введение в язык Python

**Знать:** сферы применения языка Python; определения понятий «язык программирования», «алгоритм», «программа», «функция»; смысл основных правил синтаксиса Python: правило порядка, правило начала, правило оформления кода

**Уметь:** печатает данные в форме констант с использованием операторов: +, -, \*, //, %, (, ) с помощью функции print()

**Познавательные:** умеет пользоваться средой программирования Python, понимает, что ошибки в консоли говорят о неисправности программы

**Регулятивные:** определять последовательность промежуточных целей с учетом конечного результата.

**Коммуникативные:** задавать нужные вопросы для организации собственной деятельности и сотрудничества с партнером

###### Переменные

**Знать:** переменная — это элемент данных, имеющий имя, строковый и численный (целые числа и десятичная дробь) типы данных; как меняются значения переменных при присваивании

**Уметь:** запрашивать данные с клавиатуры с помощью функции input(); изменяет тип с помощью int()

**Познавательные:** понимание переменной как инструмента, позволяющего писать универсальный код и раскрыть понятие «тип данных» на трёх примерах: целочисленного типа, численного типа с плавающей точкой и строкового типа.

**Регулятивные:** планировать свое действие в соответствии с поставленной задачей и условиями ее реализации, в том числе и на внутреннем плане.

**Коммуникативные:** использовать речь для регуляции своего действия; с учетом целей коммуникации достаточно точно, последовательно и полно передавать партнеру необходимую информацию как ориентир для построения действия

###### Строки

**Знать:** функции для работы со строками; компьютер понимает только сигналы тока 0 и 1; для распознавания и выполнения компьютером команд языка программирования требуется специальная программа — интерпретатор

**Уметь:** складывать строки, получать символ строки по его номеру, получать часть строки по номерам начала и следующего за концом, искать вхождение одной строки в другую, делать все буквы строки строчными (маленькими)

**Познавательные:** методы работы со строками; как информация представлена в компьютере и как с помощью интерпретатора происходит распознавание и исполнение команд.

**Регулятивные:** принимать

познавательную цель, сохранять её при выполнении учебных действий.

**Коммуникативные:** управлять поведением партнера – убеждать его, контролировать, корректировать и оценивать его действия

###### Вложенные конструкции

**Знать:** важность создания качественного и лаконичного кода; важность создания легко читаемого кода

**Уметь:** делать код короче, вкладывая одну команду в другую (например, int(input())); выполнять действия с целыми числами и десятичным и дробными; выполнять действия со строками; распознавать и исправлять ошибки, вызванные неправильным использованием типа данных

**Познавательные:** способы создания и оптимизации качественного кода с использованием одной конструкции внутри другой

**Регулятивные:** самостоятельно планировать пути достижения целей; соотносить свои действия с планируемыми результатами.

**Коммуникативные:** устанавливать и сравнивать разные точки зрения, прежде чем принимать решение и делать выбор; задавать уточняющие вопросы для получения недостающей информации

## Тема 2 УПРАВЛЯЮЩИЕ КОНСТРУКЦИИ (5 часов)

### Условный оператор

**Знать:** назначение логического типа данных; все возможные значения переменных и выражений, относящиеся к логическому типу; основные логические операции (логическое сложение и умножение) и порядок их выполнения; синтаксис условного оператора; значения элементов блок-схемы и читать записанный с её помощью алгоритм;

**Уметь:** использовать условный оператор для программирования выбора одной команды или блока команд в зависимости от значения логического выражения (условия)

**Познавательные:** сформировать понимание условного оператора как конструкции, выбирающей команду для исполнения в зависимости от значения логического выражения.

**Регулятивные:** вносить коррективы и дополнения в составленные планы; принимать познавательную цель, сохранять её при выполнении учебных действий, регулировать весь процесс их выполнения.

**Коммуникативные:** с достаточной полнотой и точностью выражать свои мысли в соответствии с задачами и условиями коммуникации; вступать в диалог; участвовать в коллективном обсуждении проблем

### Вложенный условный оператор

**Знать:** синтаксис вложенного условного оператора и условного оператора с несколькими ветвями;

**Уметь:** программировать условия разными способами (составное логическое выражение, вложенный условный оператор, несколько ветвей условий); читать алгоритмы, записанные с помощью блок-схем; распознавать ошибки консоли, связанные с неправильным использованием логического типа данных.

**Познавательные:** владеть информационно-логическими умениями: определять понятия, самостоятельно выбирать основания и критерии для классификации, делать выводы.

**Регулятивные:** определять способы действий в рамках предложенных условий; оценивать правильность выполнения учебной задачи.

**Коммуникативные:** планировать учебное сотрудничество с учителем и сверстниками; адекватно использовать речевые средства для аргументации своей позиции

### Циклы

**Знать:** оператор while и оформление цикла; необходимость использования счётчика для подсчёта количества повторов цикла;

**Уметь:** описывать назначение циклической конструкции; задавать условие работы цикла с помощью логического выражения; выбирать подходящую алгоритмическую конструкцию: условную или циклическую; цикл while со счётчиком;

**Познавательные:** изучают циклическую конструкцию и оператор while; узнают синтаксис цикла и цикла со счётчиком.

**Регулятивные:** преобразовывать практическую задачу в познавательную; вносить коррективы и дополнения в составленные планы; адекватно воспринимать оценку учителя.

**Коммуникативные:** задавать вопросы, необходимые для организации собственной деятельности и сотрудничества с партнером; использовать речь для регуляции своего действия

### Циклы. Продолжение

**Знать:** циклические конструкции с использованием операторов while и for; новый аспект вложенности (цикл и ветвление можно вкладывать друг в друга);

**Уметь:** использовать функцию range в цикле for для повтора действий n-раз; использовать цикл for для перебора символов строки; программировать вложенные управляющие конструкции (условный оператор в цикле, цикл в цикле);

**Познавательные:** изучают цикл for для повтора действий n-раз и для перебора символов строки.

**Регулятивные:** проявлять способность к волевому усилию в случае затруднения; осуществлять контроль на уровне произвольного внимания.

**Коммуникативные:** понимать относительность мнений и подходов к решению проблемы; осуществлять взаимный контроль и оказывать в сотрудничестве необходимую помощь

#### **Вложенные управляющие конструкции**

**Знать:** оператор break для прерывания цикла; алгоритмические конструкции

**Уметь:** разбить большую задачу на части и пошагово работать над ней; распознать и исправить ошибки в использовании алгоритмических конструкций и логического типа данных;

**Познавательные:** систематизируют алгоритмические конструкции

**Регулятивные:** принимать взвешенные решения и осуществлять осознанный выбор в учебной и познавательной деятельности.

**Коммуникативные:** продуктивно разрешать конфликты на основе учета интересов и позиций всех его участников; уметь с достаточной полнотой и точностью выразить свои мысли в соответствии с задачами и условиями коммуникации

### **Тема 3 ФУНКЦИИ И МОДУЛИ (4 часа)**

#### **Функции**

**Знать:** функция — это именованный набор команд программы, который может быть вызван из другой части программы; для конкретной функции называет её имя, параметры (аргументы) и возвращаемое значение;

**Уметь:** своими словами объяснять, что такое локальная область видимости; создать функцию, используя оператор def; возвращать значение из функции с помощью оператора return; вызывать в собственной функции встроенные функции Python.

**Познавательные:** понимание функции как фрагмента кода программы (подпрограммы), команды которого выполняются при обращении по имени.

**Регулятивные:** определять способы действий в рамках предложенных условий и оценивать правильность выполнения учебной задачи.

**Коммуникативные:** проявлять инициативу в поиске и сборе информации в сотрудничестве с партнером; владеть диалогической формой речи в соответствии с грамматическими и синтаксическими нормами родного языка

#### **Функции. Продолжение**

**Знать:** к функции можно обращаться не только из основной части программы, но и локально (из другой функции); отладчик в качестве инструмента контроля значений переменных.

**Уметь:** объявлять функцию с помощью оператора def; возвращать значение из функции с помощью оператора return; вызывать в собственной функции встроенные функции Python; вызывать в одной собственной функции другую собственную функцию; правильно выделяли глобальную и локальную области видимости; создавали собственные функции, возвращающие и не возвращающие значения;

**Познавательные:** узнать и применить на практике вызов одной функции в другой

**Регулятивные:** принимать

и сохраняют учебную задачу; соотносить свои действия с планируемыми результатами; осуществлять пошаговый контроль по результату.

**Коммуникативные:** допускать возможность существования у людей различных точек зрения; ориентироваться на позицию партнера в общении и взаимодействии; адекватно использовать речь для планирования и регуляции своей деятельности

#### **Модули random и time**

**Знать:** в стандартной библиотеке Python есть много готовых модулей; команды модулей random и time

**Уметь:** подключать модули к программе; использует команды модуля random для генерации случайных чисел; использовать команды модуля time для работы с системным временем; распознавать и исправлять ошибки, вызванные неправильным использованием модулей.

**Познавательные:** узнать и применить возможности модулей стандартной библиотеки Python: random и time.

**Регулятивные:** учитывать выделенные учителем ориентиры действия в новом учебном материале; осознавать качество и уровень усвоения материала.

**Коммуникативные:** договариваться и приходить к общему решению в совместной деятельности, в том числе в ситуации столкновения интересов; задавать вопросы, необходимые для организации собственной деятельности и сотрудничества с партнером

### **Создание модулей**

**Знать:** зачем разработчики создают собственные модули; разные способы подключения модулей; процесс создания собственных модулей.

**Уметь:** приводить примеры готовых модулей стандартной библиотеки Python; использовать готовые модули других разработчиков (не из стандартной библиотеки Python) в программах; подключать к своей программе как отдельные функции из модулей, так и модуль целиком; создавать собственные модули и использовать их в программах; распознавать и исправлять ошибки, вызванные неправильным использованием модулей;

**Познавательные:** способы подключения модулей к программе и способ создания собственных модулей

**Регулятивные:** преобразовывать практическую задачу в познавательную; самостоятельно оценивать правильность выполнения действия и вносить необходимые коррективы в исполнение как по ходу его реализации, так и в конце действия.

**Коммуникативные:** учитывать разные мнения и стремиться к координации различных позиций в сотрудничестве

## **Тема 4 МОДУЛЬ TURTLE. МАТЕМАТИКА ДЛЯ РАЗРАБОТЧИКА (4 часа)**

### **Turtle. Линейные алгоритмы**

**Знать:** возможности модуля turtle для отрисовки графических примитивов

**Уметь:** своими словами объяснить, что такое исполнитель (пример исполнителя: черепашка); задать угол поворота черепашки в градусах и координаты точки для перемещения на некоторое расстояние; показывать на экране начало координат черепашки; использовать функции forward(), left(), right(), goto() для перемещения черепашки; использовать функции pensize(), color(), penup(), pendown() для отрисовки фигур; программировать линейные алгоритмы для отрисовки простых фигур исполнителем.

**Познавательные:** как устроено изображение в компьютере, и учатся рисовать простейшие геометрические фигуры с помощью команд модуля turtle; что такое система координат, координаты точки и как переместить исполнителя в нужную точку на экране.

**Регулятивные:** самостоятельно формулировать познавательную цель и строить действия в соответствии с ней; оценивать достигнутый результат.

**Коммуникативные:** использовать адекватные языковые средства для отображения своих чувств, мыслей и побуждений; проявлять готовность реагировать на нужды других, оказывать помощь и эмоциональную поддержку партнерам

### **Turtle. Циклы**

**Знать:** циклические алгоритмы для отрисовки сложных фигур

**Уметь:** задать угол поворота в градусах и координаты точки для перемещения черепашки; показывать на экране примерное начало координат черепашки; использовать функции forward(), left(), right(), goto() для перемещения черепашки; использовать функции pensize(), color(), penup(), pendown() для отрисовки фигур; регулировать скорость черепашки и скрывает исполнителя;

**Познавательные:** новые команды модуля turtle и возможности их использования в циклических конструкциях.

**Регулятивные:** в сотрудничестве с учителем ставить новые учебные задачи; принимать познавательную цель, сохранять ее при выполнении учебных действий.

**Коммуникативные:** проявлять готовность к обсуждению разных точек зрения и выработке общей (групповой) позиции; строить понятные для партнера высказывания



### **Turtle. Условный оператор**

**Знать:** условные алгоритмические структуры для отрисовки фигур исполнителем.

**Уметь:** задавать угол поворота черепашки в градусах и координаты точки для перемещения на некоторое расстояние; показывать на экране начало отсчёта системы координат черепашки; использовать функции forward(), left(), right(), goto() для перемещения черепашки; использовать функции pensize(), color(), penup(), pendown(), speed() для задания параметров отрисовки фигур;

**Познавательные:** функции модуля turtle для рисования фигур.

**Регулятивные:** планировать свои действия в соответствии с поставленной задачей и условиями ее реализации, а также во внутреннем плане.

**Коммуникативные:** с учетом целей коммуникации достаточно точно, последовательно и полно передавать партнеру необходимую информацию как ориентир для построения действия

### **Проект "Городская среда"**

**Знать:** алгоритмы для создания геометрических фигур

**Уметь:** создавать функции для создания множества геометрических фигур

**Познавательные:** работа над многоступенчатым проектом

**Регулятивные:** самостоятельно формулировать познавательную цель и строить действия в соответствии с ней; выделять и осознавать то, что уже усвоено и что еще подлежит усвоению, осознавать качество и уровень усвоения.

**Коммуникативные:** вступать в диалог; участвовать в коллективном обсуждении проблем; владеть монологической и диалогической формами речи в соответствии с грамматическими и синтаксическими нормами родного языка

## **Тема 5 ОБЪЕКТНО-ОРИЕНТИРОВАННОЕ ПРОГРАММИРОВАНИЕ(6 часов)**

### **ООП. Объекты и методы**

**Знать:** что такое объект, свойство и метод объекта; идею ООП (исполнение программы представляется как действия разных объектов)

**Уметь:** создавать объекты и обращаться к их свойствам и методам через точку; знать и использовать команды Turtle(), shape(); создавать программы с несколькими объектами.

**Познавательные:** работу с несколькими объектами в одной программе; объект, свойство и метод

**Регулятивные:** проявлять

познавательную инициативу в учебном сотрудничестве; адекватно воспринимать оценку учителя.

**Коммуникативные:** понимать относительность мнений и подходов к решению проблемы; осуществлять инициативное сотрудничество в поиске и сборе информации

### **ООП. События**

**Знать:** что такое событие, подписка на событие и обработчик события; своими словами описывает, как программа выделяет значимые события и реагирует на них;

**Уметь:** использует метод объекта turtle onclick(); обращается к свойствам и методам объекта turtle; создаёт программы с несколькими объектами и обрабатывает клик мышкой по одному из них.

**Познавательные:** изучить и применить на практике обработку событий на примере клика мышкой по черепашке.

**Регулятивные:** соотносить свои действия с планируемыми результатами; осуществлять контроль своей деятельности.

**Коммуникативные:** учиться разрешать конфликты: выявлять, идентифицировать проблемы, искать и оценивать альтернативные способы разрешения конфликта, принимать и реализовывать решение

## **ООП. Проект Simple Paint**

**Знать:** что такое событие, подписка на событие, обработка события, для чего использовать события в программе;

**Уметь:** создаёт подписку на события, связанные с экраном и черепашкой; обрабатывает события, связанные с экраном, на котором расположена черепашка; обрабатывает события, связанные с нажатием на клавиши клавиатуры; знает и применяет методы объекта turtle `ondrag()`, `getscreen()` и методы объекта `screen`: `onclick()`, `onkey()`, `listen()`; применяет знания и умения работы с объектами и событиями в процессе реализации проекта.

**Познавательные:** На занятии ученики проходят полный цикл работы над проектом: знакомятся с техническим заданием, составляют чек-лист с этапами работы, выделяют подзадачи, реализуют проект и обмениваются обратной связью.

**Регулятивные:** принимать и сохранять учебную задачу; учитывать правила в планировании и контроле способа решения.

**Коммуникативные:** контролировать действия партнера; оказывать в сотрудничестве необходимую помощь

## **ООП. Классы**

**Знать:** что такое класс, экземпляр, конструктор; понимает, зачем создавать и использовать классы в программе; своими словами объясняет связь объектов и классов;

**Уметь:** создаёт собственные классы и их экземпляры; программирует конструктор класса и в нём определяет его поля; описывает методы класса.

**Познавательные:** изучить и применить на практике создание классов и взаимодействие с ними через консоль. В первой половине урока ученики узнают, что такое класс, экземпляр класса, конструктор класса, создадут первый класс Него и запрограммируют схватку двух персонажей. Во второй половине урока ученики используют класс Него для создания текстового квеста «Рыцарь и дракон».

**Регулятивные:** принимать и сохранять учебную задачу; осуществлять констатирующий и предвосхищающий контроль по результату и по способу действия.

**Коммуникативные:** адекватно использовать речевые средства для решения различных коммуникативных задач

## **ООП. Наследование**

**Знать:** что такое класс, суперкласс, класс-наследник; понимает, зачем использовать наследование в программах (в частности, в играх); знает два способа создания класса-наследника: без дополнения конструктора суперкласса и с дополнением новыми полями;

**Уметь:** программирует класс-наследник от собственного суперкласса; программирует класс-наследник от готового суперкласса.

**Познавательные:** изучить и применить на практике концепцию наследования классов как от собственного суперкласса, так и от класса стандартной библиотеки Python.

**Регулятивные:** планировать свое действие в соответствии с поставленной задачей и условиями ее реализации, в том числе и на внутреннем плане; самостоятельно оценивать правильность выполнения действия.

**Коммуникативные:** задавать вопросы, необходимые для организации собственной деятельности и сотрудничества с партнером

## **ООП. Наследование. Продолжение**

**Знать:** что такое класс, суперкласс, класс-наследник; понимает, зачем создавать и использовать наследование в программах (в частности, в играх); знает два способа создания класса-наследника: без дополнения конструктора суперкласса и с созданием новых полей;

**Уметь:** использует экземпляры суперкласса и класса-наследника в одной программе.

**Познавательные:** экземпляры суперкласса Turtle и класса-наследника Sprite в создании игры;

**Регулятивные:** определять последовательность промежуточных целей с учетом конечного результата; сличать свой способ действия с эталоном.

**Коммуникативные:** с достаточной полнотой и точностью выражать свои мысли в соответствии с задачами и условиями коммуникации; описывать содержание совершаемых действий

## Тема 6 ОСНОВЫ РАЗРАБОТКИ ИГР НА PYGAME (8 часов)

### Основы создания игр на Pygame

**Знать:** возможности PyGame; своими словами описывает принцип обработки событий клавиш на PyGame; понимает, как задать цвет с помощью палитры цветов RGB;

**Уметь:** создаёт окно игры, заливаает фон цветом; создаёт спрайт-текст, задаёт его свойства и располагает на экране; создаёт игровой цикл, настраивает частоту обновления кадров в секунду; распознаёт и обрабатывает нажатия на клавиши; использует классы в создании собственных игровых объектов.

*Познавательные:* изучить и применить на практике основы создания игр с помощью библиотеки PyGame.

*Регулятивные:* принимать и сохранять учебную задачу; вносить коррективы в свое действие после его завершения на основе оценки и учета характера сделанных ошибок.

*Коммуникативные:* строить понятные для партнера высказывания, учитывающие, что партнер знает и видит, а что – нет; аргументировать свою точку зрения

### Списки

**Знать:** что такое список, нумерация элементов списка, как вернуть элемент по его номеру (индексу); одном списке могут быть объекты разных типов; методы работы со строками и списками.

**Уметь:** использует методы работы со списком: добавление, удаление, вставка, сортировка, подсчёт вхождения элемента; перебирает элементы списка, используя цикл for; перечисляет возможности PyGame; настраивает игровой цикл; использует классы в создании игровых объектов.

*Познавательные:* изучить и применить на практике списки и методы работы с ними.

*Регулятивные:* вносить коррективы и дополнения в способ своих действий в случае расхождения эталона, реального действия и его продукта.

*Коммуникативные:* управлять поведением партнера: убеждать его, контролировать, корректировать и оценивать его действия; допускать возможность существования у людей различных точек зрения, в том числе не совпадающих с собственной

### Игра Fast Clicker. Ч. 1

**Знать:** команды Pygame для создания заготовки игры;

**Уметь:** создаёт собственный класс и наделяет его свойствами и методами; создаёт класс-наследник существующего класса; создаёт экземпляры собственного класса и заполняет ими список; работает со списком объектов, используя методы работы со списками;

*Познавательные:* устанавливать причинно-следственные связи; строить логическое рассуждение; подбирать алгоритмическую конструкцию, соответствующую заданной ситуации.

*Регулятивные:* проявлять

познавательную инициативу в учебном сотрудничестве; самостоятельно оценивать правильность выполнения действия и вносить необходимые коррективы в исполнение как по ходу его реализации, так и в конце действия.

*Коммуникативные:* понимать относительность мнений и подходов к решению проблемы; аргументировать свою позицию и координировать ее с позициями партнеров в сотрудничестве при выработке общего решения в совместной деятельности

### Игра Fast Clicker. Часть 2

**Знать:** команды Pygame для создания заготовки игры;

**Уметь:** создаёт и работает с экземплярами собственного класса; дополняет существующий класс новыми методами; работает со списком объектов, используя методы работы со списками; обрабатывает события мыши инструментами pygame;

**Познавательные:** осуществлять поиск необходимой информации для выполнения учебных заданий с помощью компьютера; анализировать объекты с целью выделения признаков .  
**Регулятивные:** ставить учебную задачу на основе соотнесения того, что уже известно и усвоено, и того, что еще неизвестно; различать способ и результат действия.  
**Коммуникативные:** осуществлять взаимный контроль и оказывать в сотрудничестве необходимую помощь; владеть диалогической формой речи

### **Игра Fast Clicker. Ч. 3**

**Знать:** команды Pygame для создания заготовки игры;

**Уметь:** создаёт и работает с экземплярами собственного класса; дополняет существующий класс новыми методами; работает со списком объектов, используя методы работы со списками; программирует счётчики статистики времени и набранных очков; использует данные счётчиков для описания условий победы и проигрыша;

**Познавательные:** создавать и преобразовывать алгоритмы для решения задач; осуществлять выбор наиболее эффективных способов решения задач в зависимости от конкретных условий.

**Регулятивные:** учитывать выделенные учителем ориентиры действия в новом учебном материале в сотрудничестве с учителем; адекватно воспринимать оценку учителя.

**Коммуникативные:** договариваться и приходить к общему решению в результате совместной деятельности, в том числе в ситуации столкновения интересов

### **Игра «Арканоид». Часть 1**

**Знать:** знает и использует команды Pygame для создания заготовки игры;

**Уметь:** создаёт класс-наследник существующего класса; создаёт экземпляры собственного класса и заполняет ими список; работает со списком объектов, используя методы работы со списками; использует наработки из прошлого проекта;

**Познавательные:** определять основную и второстепенную информацию; составлять алгоритмы и блок-схемы на основе анализа текста задачи; строить логическую цепочку рассуждений.

**Регулятивные:** планировать свое действие в соответствии с поставленной задачей и условиями ее реализации, в том числе и во внутреннем плане.

**Коммуникативные:** с учетом целей коммуникации достаточно точно, последовательно и полно передавать партнеру необходимую информацию как ориентир для построения действия

### **Игра Арканоид. Часть 2**

**Знать:** знает и использует команды Pygame для создания заготовки игры;

**Уметь:** создаёт собственный класс и наделяет его свойствами и методами; создаёт класс-наследник существующего класса; создаёт экземпляры собственного класса и заполняет ими список; работает со списком объектов, используя методы работы со списками; обрабатывает события клавиатуры с помощью инструментов Pygame; программирует автоматическое перемещение спрайта по игровой сцене;

**Познавательные:** анализировать условия и требования задачи; выполнять операции со знаками и символами; составлять целое из частей, самостоятельно достраивая, восполняя недостающие компоненты.

**Регулятивные:** выделять и осознавать то, что уже усвоено и что еще подлежит усвоению, осознавать качество и уровень усвоения; определять последовательность промежуточных целей с учетом конечного результата.

**Коммуникативные:** адекватно использовать речевые средства для аргументации своей позиции; обмениваться знаниями между членами группы для принятия эффективных совместных решений

### **Игра Арканойд. Часть 3**

**Знать:** знает и использует команды Pygame для создания заготовки игры;

**Уметь:** управляет собственной группой спрайтов; дополняет в игровом цикле условия победы и проигрыша;

**Познавательные:** анализировать условия и требования задачи; выбирать знаково-символические средства для построения модели; составлять целое из частей, самостоятельно достраивать, восполняя недостающие компоненты.

**Регулятивные:** сличать способ и результат своих действий с заданным эталоном, обнаруживать отклонения и отличия от эталона.

**Коммуникативные:** проявлять готовность адекватно реагировать на нужды других, оказывать помощь и эмоциональную поддержку партнерам

### **Тема 7. ИТОГОВОЕ ПОВТОРЕНИЕ (2 часа)**

#### **Выполнение и защита итогового проекта Chess Vision**

**Знать:** знает и использует команды Pygame для создания заготовки игры;

**Уметь:** создаёт класс-наследник существующего класса; создаёт экземпляры собственного класса и заполняет ими список; работает со списком объектов, используя методы работы со списками; использует наработки из прошлого проекта;

**Познавательные:** самостоятельно создавать алгоритмы деятельности при решении проблем творческого и поискового характера; выбирать наиболее эффективные способы решения задачи в зависимости от конкретных условий.

**Регулятивные:** определять

последовательность промежуточных целей с учетом конечного результата; оценивать достигнутый результат.

**Коммуникативные:** с достаточной полнотой и точностью выражать свои мысли в соответствии с задачами и условиями коммуникации; уметь представлять конкретное содержание и сообщать его в письменной и устной форме

Календарно- тематическое планирование по информатике на основе:

- Методическое пособие: Алгоритмика Python Start 2020/2021

**Календарно-тематическое планирование кружка программирования на Python**

**Класс:** 6 - 11 класс

**Количество часов за год всего 34 часа, в неделю 1 час.**

**Планирование составлено на основе:**

Методическое пособие: Алгоритмика Python Start 2020/2021

№ урока	Дата проведения		Тема урока
	План	Факт	
<b>Раздел 1. Основы языка</b>			
<b>Тема 1 ОСНОВЫ ЯЗЫКА</b>			
1			Введение в язык Python
2			Переменные
3			Строки
4			Вложенные конструкции
<b>Тема 2 УПРАВЛЯЮЩИЕ КОНСТРУКЦИИ</b>			
5			Условный оператор
6			Вложенный условный оператор
7			Циклы
8			Циклы. Продолжение
9			Вложенные управляющие конструкции
<b>Тема 3 ФУНКЦИИ И МОДУЛИ</b>			
10			Функции
11			Функции. Продолжение
12			Модули random и time
13			Создание модулей
<b>Тема 4 МОДУЛЬ TURTLE. МАТЕМАТИКА ДЛЯ РАЗРАБОТЧИКА</b>			
14			Turtle. Линейные алгоритмы
15			Turtle. Циклы
16			Turtle. Условный оператор
17			Проект "Городская среда"
<b>Тема 5 ОБЪЕКТНО-ОРИЕНТИРОВАННОЕ ПРОГРАММИРОВАНИЕ</b>			

18			ООП. Объекты и методы
19			ООП. События
20			ООП. Проект Simple Paint
21			ООП. Классы
22			ООП. Наследование
23			ООП. Наследование. Продолжение
<b>Тема 6 ОСНОВЫ РАЗРАБОТКИ ИГР НА PYGAME</b>			
24			Основы создания игр на Pygame
25			Списки
26			Игра Fast Clicker. Ч. 1
27			Игра Fast Clicker. Часть 2
28			Игра Fast Clicker. Ч. 3
29			Игра «Арканоид». Часть 1
30			Игра Арканоид. Часть 2
31			Игра Арканоид. Часть 3
<b>Тема 7. ИТОГОВОЕ ПОВТОРЕНИЕ</b>			
32 – 34			Выполнение и защита итогового проекта Chess Vision

Пронумеровано, прошнуровано и скреплено  
печатью \_\_\_\_\_ страниц.

Директор школы \_\_\_\_\_ Ю.А.Собко

