

Муниципальное общеобразовательное учреждение
«Архангельская средняя школа»

Рассмотрена на заседании
педсовета
Протокол от 26.08.2025 № 25

Утверждена
Приказом директора
от 26.08.2025 № 294

Дополнительная общеобразовательная (общеразвивающая)
программа
**«Основы программирования на языке Python на примере
программирования беспилотного летательного аппарата»**

Возраст детей: 15-17 лет
Автор-составитель: Межева М. В.
учитель информатики

с. Архангельское, 2025 г.

ДОКУМЕНТ ПОДПИСАН ЭЛЕКТРОННОЙ ПОДПИСЬЮ

МУНИЦИПАЛЬНОЕ ОБЩЕОБРАЗОВАТЕЛЬНОЕ УЧРЕЖДЕНИЕ
"АРХАНГЕЛЬСКАЯ СРЕДНЯЯ ШКОЛА", Баранова Светлана Николаевна,
Директор

17.04.26 13:54 (MSK)

Сертификат 2DDA96F43C485DE430B4D8DEB64A74F3

1. Пояснительная записка

Дополнительная общеобразовательная (общеразвивающая) программа «Основы программирования на языке Python на примере программирования беспилотного летательного аппарата» технической направленности базового уровня разработана в соответствии с нормативно-правовыми требованиями развития дополнительного образования детей и в соответствии с:

- Концепцией развития дополнительного образования детей от 4 сентября 2014 г. № 1726-р;

Приказом Министерства просвещения Российской Федерации от 09 ноября 2018 №196 (ред. 2020 года) «Об утверждении Порядка организации и осуществления образовательной деятельности по дополнительным общеобразовательным программам».

- Распоряжением Правительства РФ от 29 февраля 2016 г. № 326-р (ред. от 30 марта 2018 г.) «Об утверждении Стратегии государственной культурной политики на период до 2030 года».

Актуальность и отличительные особенности

В настоящее время процесс информатизации проявляется во всех сферах человеческой деятельности. Использование современных информационных технологий является необходимым условием успешного развития как отдельных отраслей, так и государства в целом. Создание, внедрение, эксплуатация, а также совершенствование информационных технологий немыслимо без участия квалифицированных и увлечённых специалистов, в связи с этим внедрение курса «Основы программирования на языке Python на примере программирования беспилотного летательного аппарата» в учебный процесс актуально.

Программа учебного курса «Основы программирования на языке Python на примере программирования беспилотного летательного аппарата» направлена на подготовку творческой, технически грамотной, гармонично развитой личности, обладающей логическим мышлением, способной анализировать и решать задачи в команде в области информационных и аэротехнологий, решать ситуационные кейсовые задания, основанные на групповых проектах.

Занятия по данному курсу рассчитаны на общенаучную подготовку обучающихся, развитие их мышления, логики, математических способностей, исследовательских навыков.

Учебный курс «Основы программирования на языке Python на примере программирования беспилотного летательного аппарата» направлен на изучение основ программирования на языке Python и программирование автономных квадрокоптеров.

В рамках курса «Основы программирования на языке Python на примере программирования беспилотного летательного аппарата» обучающиеся смогут познакомиться с физическими, техническими и математическими понятиями. Приобретённые знания будут применимы в творческих проектах.

Учебный курс «Основы программирования на языке Python на примере программирования беспилотного летательного аппарата» представляет собой самостоятельный модуль и содержит необходимые темы из курса информатики и физики.

Направленность программы: техническая.

Уровень освоения: базовый.

Адресата программы: обучающиеся в возрасте 15-17 лет

Цель программы: освоение Hard- и Soft-компетенций обучающимися в области программирования и аэротехнологий через использование кейс-технологий.

Задачи:

Предметные:

изучить базовые понятия: алгоритм, блок-схема, переменная, цикл, условия, вычисляемая функция;

сформировать навыки выполнения технологической цепочки разработки программ средствами языка программирования Python;

изучить основные конструкции языка программирования Python, позволяющие работать с простыми и составными типами данных (строками, списками, кортежами, словарями, множествами);

научить применять навыки программирования на конкретной учебной ситуации (программирование беспилотных летательных аппаратов на учебную задачу);

развить навык пилотирования беспилотных летательных аппаратов (БПЛА) на практике;

привить навыки проектной деятельности.

Личностные:

способствовать расширению словарного запаса;

способствовать развитию памяти, внимания, технического мышления, изобретательности;

способствовать развитию алгоритмического мышления;

способствовать формированию интереса к техническим знаниям;

способствовать формированию умения практического применения полученных знаний;

сформировать умение формулировать, аргументировать и отстаивать своё мнение;

сформировать умение выступать публично с докладами, презентациями и т. п.

Метапредметные:

обеспечить образовательные интересы обучающихся с помощью информационных технологий;

развивать познавательный интерес, интеллектуальные и творческие способности средствами ИКТ;

воспитать интерес к творческой и исследовательской деятельности в медиа индустрии.

Объем программы: общее количество учебных часов, запланированных на весь период обучения, необходимых для освоения программы, составляет 68 учебных часов.

Наполняемость группы: 13 человек.

Режим занятий

Занятия в творческом объединении проводятся с учетом возрастных особенностей детей и в соответствии с санитарными нормами: 2 раза в неделю по 40 минут.

Формы организации занятий:

Индивидуальная;

Групповая;

Коллективная.

Теоретические занятия могут проходить с применением дистанционных образовательных технологий, например, посредством программы (ВК-мессенджер (Сферум), Яндекс Телемост), записи лекций. Такая двухсторонняя форма коммуникации позволяет обучающимся, не имеющим возможности посещать все занятия в силу различных обстоятельств, получить доступ к изучению программы.

В процессе реализации программы будут использованы следующие формы обучения: очная.

Все занятия (кроме вводного) имеют практико-ориентированный характер. Каждый учащийся может работать как индивидуально над собственными учебными творческими проектами, так и над общим в команде.

Прогнозируемые результаты:

Предметные результаты:

знать:

- основные алгоритмические конструкции;
- принципы построения блок-схем;
- принципы структурного программирования на языке Python;
- что такое БПЛА и их предназначение.

уметь:

- составлять алгоритмы для решения прикладных задач;
- реализовывать алгоритмы на компьютере в виде программ, написанных на языке Python;
- применять библиотеку Tkinter;
- отлаживать и тестировать программы, написанные на языке Python;
- настраивать БПЛА;
- представлять свой проект.

владеть:

- основной терминологией в области алгоритмизации и программирования;
- основными навыками программирования на языке Python;
- знаниями по устройству и применению беспилотников.

Личностные результаты:

критическое отношение к информации и избирательность её восприятия;
осмысление мотивов своих действий при выполнении заданий;
развитие любознательности, сообразительности при выполнении разнообразных заданий проблемного и эвристического характера;
развитие внимательности, настойчивости, целеустремлённости, умения преодолевать трудности;
развитие самостоятельности суждений, независимости и нестандартности мышления;
освоение социальных норм, правил поведения, ролей и форм социальной жизни в группах и сообществах;
формирование коммуникативной компетентности в общении и сотрудничестве с другими обучающимися.

Метапредметные результаты:

Регулятивные универсальные учебные действия:

умение принимать и сохранять учебную задачу;
умение планировать последовательность шагов алгоритма для достижения цели;
умение ставить цель (создание творческой работы), планировать достижение этой цели;
умение осуществлять итоговый и пошаговый контроль по результату;
способность адекватно воспринимать оценку наставника и других обучающихся;
умение различать способ и результат действия;
умение вносить коррективы в действия в случае расхождения результата решения задачи на основе её оценки и учёта характера сделанных ошибок;
умение в сотрудничестве ставить новые учебные задачи;
способность проявлять познавательную инициативу в учебном сотрудничестве;

умение осваивать способы решения проблем творческого характера в жизненных ситуациях;

умение оценивать получающийся творческий продукт и соотносить его с изначальным замыслом, выполнять по необходимости коррекции либо продукта, либо замысла.

Познавательные универсальные учебные действия:

умение осуществлять поиск информации в индивидуальных информационных архивах обучающегося, информационной среде образовательного учреждения, федеральных хранилищах информационных образовательных ресурсов;

умение использовать средства информационных и коммуникационных технологий для решения коммуникативных, познавательных и творческих задач;

умение ориентироваться в разнообразии способов решения задач;

умение осуществлять анализ объектов с выделением существенных и несущественных признаков;

умение проводить сравнение, классификацию по заданным критериям;

умение строить логические рассуждения в форме связи простых суждений об объекте;

умение устанавливать аналогии, причинно-следственные связи;

умение моделировать, преобразовывать объект из чувственной формы в модель, где выделены существенные характеристики объекта (пространственно-графическая или знаково-символическая);

умение синтезировать, составлять целое из частей, в том числе самостоятельно достраивать с восполнением недостающих компонентов.

Коммуникативные универсальные учебные действия:

умение аргументировать свою точку зрения на выбор оснований и критериев при выделении признаков, сравнении и классификации объектов;

умение выслушивать собеседника и вести диалог;

способность признавать возможность существования различных точек зрения и права каждого иметь свою;

умение планировать учебное сотрудничество с наставником и другими обучающимися: определять цели, функции участников, способы взаимодействия;

умение осуществлять постановку вопросов: инициативное сотрудничество в поиске и сборе информации;

умение разрешать конфликты: выявление, идентификация проблемы, поиск и оценка альтернативных способов разрешения конфликта, принятие решения и его реализация;

умение с достаточной полнотой и точностью выражать свои мысли в соответствии с задачами и условиями коммуникации;

владение монологической и диалогической формами речи.

Способы определения результативности:

Занятия проходят в форме бесед, сообщений, практических занятий. Теоретическое занятие сопровождается демонстрацией методического материала презентации, видео материалы, фильмы. Практические занятия включают в себя работу с компьютером и БПЛА. На определенном этапе обучающиеся могут объединиться в мини-группы, т. е. используется проектный метод обучения.

Формы подведения итогов реализации программы:

Контроль и оценка обучающихся в кружке осуществляется при помощи текущего и итогового контроля в форме тестирования и защиты проектной работы (в конце года).

Виды контроля – тестирование, защита проекта.

2. Учебный план

Количество часов			Промежуточная аттестация и аттестация по завершении реализации программы.
Теория	Практика	Всего	
22	46	68	1 полугодие/конец уч. года

Учебный план (68 часов)

№ п/п	Название раздела, темы	Количество часов			Формы аттестации/ контроля
		Всего	Теория	Практика	
1.	Введение в программирование	14	6	8	Инструктаж по технике безопасности
2.	Tkinter. Программирование GUI на Python.	20	6	14	Практическая работа
3.	Основы программирования автономных квадрокоптеров	14	4	10	Практическая работа
4.	Программирование комплексных операций автономных квадрокоптеров	20	6	14	Проект
Итого:		68	22	46	

3. Содержание курса

№	Темы занятий	Содержание занятий
1.	Введение в программирование	
1.1	Вводное занятие. Введение в предмет, техника безопасности	Теория: введение в образовательную программу. Ознакомление обучающихся с программой, приёмами и формами работы. Вводный инструктаж по ТБ.

1.2	Основы языка Python. Примеры на языке Python с разбором конструкций: ввод и вывод данных, условия, циклы, ветвления, массивы	Теория: история языка Python, сфера применения языка, различие в версиях, особенности синтаксиса. Объявление и использование переменных в Python. Использование строк, массивов, кортежей и словарей в Python. Использование условий, циклов и ветвлений в Python. Практика: запуск интерпретатора. Различия интерпретатора и компилятора. Написание простейших демонстрационных программ. Мини-программы внутри программы. Выражения в вызовах функций. Имена переменных. Упражнения по написанию программ с использованием переменных, условий и циклов. Генерация случайных чисел. Группировка циклов в блоки. Операции сравнения.
1.3	Кейс 1. «Угадай число». Примеры на языке Python с по угадыванию чисел, метод дихотомии.	Теория: алгоритмы поиска числа в массиве. Варианты сортировок. Поиск дихотомией. Работа с переменными, работа с функциями. Практика: упражнения по поиску чисел в массиве. Упражнения на сортировку чисел. Алгоритмы поиска числа. Исследование скорости работы алгоритмов.
1.4	Кейс 1. «Угадай число». Подготовка к публичному выступлению для защиты результатов. Демонстрация отчёта в группе и защита результатов работы	Теория: создание удобной и понятной презентации. Практика: подготовка презентации для защиты. Подготовка речи для защиты.
2.	Программирование GUI на Python.	
2.1	Кейс 2. «Спаси остров» Работа на языке Python со словарями и списками, множественное присваивание, добавление элементов в список и их удаление	Теория: знакомство с кейсом, представление поставленной проблемы. Доступ к элементам по индексам. Получение слова из словаря. Отображение игрового поля игрока. Получение предположений игрока. Проверка допустимости предположений игрока. Практика: мозговой штурм. Анализ проблемы, генерация и обсуждение методов её решения. Создание прототипа программы. Отработка методик.

2.2	Планирование дизайна и механики игры. Создание главного меню игры, подсчёта очков	Теория: понятие «механика игры», ограничения, правила. Практика: упражнения. Проверка наличия буквы в секретном слове. Проверка — не победил ли игрок. Обработка ошибочных предположений. Проверка — не проиграл ли игрок. Завершение или перезагрузка игры. Создание главного меню игры, реализация подсчёта очков.
2.3	Визуализация программы в виде блок-схемы (1 ч)	Теория: проектирование проекта с помощью блок-схем. Практика: создание блок-схем. Ветвление в блок-схемах. Заканчиваем или начинаем игру с начала. Следующая попытка. Обратная связь с игроком.
2.4	Тестирование написанной программы и доработка Подготовка к публичному выступлению для защиты результатов. Демонстрация результатов работ	Практика: тестирование созданной игры-программы, доработка и расширение возможностей. Практика: подготовка презентации и речи для защиты. Презентация созданной программы.
2.6	Кейс «Калькулятор» Оформление проектной идеи. Формирование программы работ	Теория: знакомство с кейсом, представление поставленной проблемы. Практика: мозговой штурм. Анализ проблемы, генерация и обсуждение методов её решения.
2.7	Программа для работы калькулятора	Практика: написание программы для будущего калькулятора.
2.8	Создание внешнего вида калькулятора	Практика: создание внешнего вида калькулятора.
2.9	Тестирование написанной программы и доработка	Практика: тестирование созданной программы, доработка и расширение возможностей.
2.11	Подготовка к публичному выступлению для защиты результатов Демонстрация результатов работы	Практика: подготовка презентация созданной программы
3.	Основы программирования автономных квадрокоптеров	

3.1	Техника безопасности при полётах. Проведение полётов в ручном режиме	Теория: знакомство с кейсом, представление поставленной проблемы, правила техники безопасности. Практика: полёты на квадрокоптерах в ручном режиме.
3.2	Программирование взлёта и посадки беспилотного летательного аппарата	Теория: основы программирования квадрокоптеров. Практика: тестирование в режимах взлёта и посадки.
3.3	Выполнение команд «разворот», «изменение высоты», «изменение позиции»	Теория: теоретические основы выполнения разворота, изменения высоты и позиции на квадрокоптерах. Практика: тестирование в режимах разворота, изменения высоты и позиции.
3.4	Выполнение группового полёта вручную	Практика: выполнение группового полёта на квадрокоптере в ручном режиме.
4.	Программирование комплексных операций автономных квадрокоптеров	
4.1	Выполнение позиционирования по меткам	Теория: основы позиционирования квадрокоптеров. Практика: тестирование режима позиционирования по меткам.
4.2	Программирование группового полёта	Теория: основы группового полёта квадрокоптеров. Практика: программирование роя квадрокоптеров для группового полёта.
4.3	Программирование роевого взаимодействия	Теория: основы программирования роя квадрокоптеров. Практика: программирование квадрокоптеров.

4. Формы контроля, аттестации

Реализация программы «Основы программирования на языке Python на примере программирования беспилотного летательного аппарата» предусматривает входной, текущий, промежуточный контроль и итоговую аттестацию обучающихся.

Входной контроль проводится с целью выявления уровня подготовки учащихся. Осуществляется в форме собеседования.

Текущий (промежуточный) – с целью контроля усвоения учащимися тем и разделов программы. По итогам реализации разделов программы каждый учащийся защищает свою итоговую работу.

Итоговый – с целью усвоения обучающимися программного материала в целом. осуществляется в следующих формах: защита творческих работ и проектов.

5. Оценочный материала

Перечень материалов, позволяющих определить достижение учащимися планируемых результатов:

- Индивидуальная карта учащегося, для фиксации показателей освоения программы;
- Тестирование;
- Защита учебного проекта.

6. Методические обеспечение

Формы проведения занятий: лекция, практическое занятие, индивидуально-групповая работа.

Педагог использует методы:

Словесные: передача необходимой для дальнейшего обучения информации, устное изложение, беседа.

Наглядные: сопровождение рассказа презентацией, показ образцов решения задач.

Практические: решение задач; конструирование модели.

Применяемые педагогические технологии:

Информационно – коммуникационная технология - обеспечение гармоничного развития личности, ориентирующей в информационном пространстве, приобщенной к информационно-коммуникационным возможностям современных технологий и обладающей информационной культурой, а также представить имеющийся опыт и выявить его результативность.

Игровые технологии – создание игровых ситуаций, направленных на воссоздание и усвоение общественного опыта, в котором складывается и совершенствуется самоуправление поведением.

7. Условия реализации программы

Для реализации программы необходимо:

- оборудованный учебный кабинет (стол для педагога, столы для учащихся, стулья, стенды);

- технические средства обучения (компьютеры, интерактивная доска, экран, принтер), квадрокоптер Tello feel the fun или другие.

информационное обеспечение: среда программирования Python.

- расходные материалы на весь учебный год: бумага офисная формата А4;

8. Кадровое обеспечение

Программа «Основы программирования на языке Python на примере программирования беспилотного летательного аппарата» реализуется педагогами, имеющими профессиональное образование в области, соответствующей профилю программы, и постоянно повышающими уровень профессионального мастерства.

9. Материально-техническое обеспечение программы

- рабочее место ученика, оборудованное в соответствии с санитарно-гигиеническими нормами;
- принтер;
- интерактивная панель;
- квадрокоптер Tello или другие..

10. Информационное обеспечение

Ноутбук, интернет-подключение, транслирование успехов и достижений обучающихся в группе Точка Роста МОУ Архангельская СШ в соц.сети ВКонтакте.

12. Список литературы и методического материала

1. Гин, А.А. Приёмы педагогической техники: свобода выбора, открытость, деятельность, обратная связь, идеальность: Пособие для учителей / А.А. Гин. — Гомель: ИПП «Сож», 1999. — 88 с.
2. Бреннан, К. Креативное программирование / К. Бреннан, К. Болкх, М. Чунг. — Гарвардская Высшая школа образования, 2017.
3. Лутц, М. Программирование на Python. Т. 1 / М. Лутц. — М.: Символ, 2016. — 992 с.
4. Лутц, М. Программирование на Python. Т. 2 / М. Лутц. — М.: Символ, 2016. — 992 с.
5. Понфиленок, О.В. Клевер. Конструирование и программирование квадрокоптеров / О.В. Понфиленок, А.И. Шлыков, А.А. Коригодский. — Москва, 2016.
6. Бриггс, Джейсон. Python для детей. Самоучитель по программированию / Джейсон Бриггс. — МИФ. Детство, 2018. — 320 с.
7. <https://github.com/dji-sdk/Tello-Python>.
8. <https://dl-cdn.ryzero.com/downloads/tello/0222/Tello+Scratch+Readme.pdf>.