

**Муниципальное общеобразовательное учреждение
«Средняя общеобразовательная школа п.Первомайский»
Дергачевского района Саратовской области**

**Центр образования цифрового и гуманитарного профилей
«Точка роста»**

ПРИНЯТА

На заседании педагогического совета
Протокол № 5_ от 22.05.2020 г.

УТВЕРЖДАЮ

Директор МОУ «СОШ п.Первомайский»

Кучер Г.У.Кучербаева
Приказ № 104 от 22.05.2020 г.



Дополнительная

общеобразовательная общеразвивающая

программа технической направленности

**«Основы программирования на языке Python на примере
программирования беспилотного летательного аппарата»**

Целевая аудитория: дети 12-15 лет

Срок реализации: 72 часа

Содержание

1. Комплекс основных характеристик дополнительной общеобразовательной общеразвивающей программы	
1.1. Пояснительная записка	3
1.2. Цель и задачи программы	4
1.3. Планируемые результаты освоения программы и способы их проверки	5
1.4. Содержание программы	8
1.5. Формы аттестации и оценочные материалы	13
2. Комплекс организационно-педагогических условий	
2.1. Методическое обеспечение	14
2.2. Условия реализации программы	16
2.3. Календарный учебный график	17
2.5. Список литературы	21

1. Комплекс основных характеристик дополнительной общеобразовательной общеразвивающей программы

1.1. Пояснительная записка

Программа разработана в соответствии с Федеральным законом от 29.12.2012г. № 273-ФЗ "Об образовании в Российской Федерации", приказом Приказ Министерства просвещения России от 09.11.2018 № 196 «Об утверждении Порядка организации и осуществления образовательной деятельности по дополнительным общеобразовательным программам».

Программа предложена для реализации с детьми школьного возраста - с 12 до 15 лет. Количество детей в группе (8-10). Направленность: техническая

Актуальность: 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 немислимо 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.

Отличительные особенности программы

К основным отличительным особенностям настоящей программы можно отнести следующие пункты:

- кейсовая система обучения;
- проектная деятельность;
- направленность на soft-skills;
- игропрактика;
- среда для развития разных ролей в команде;
- направленность на развитие системного мышления;

Новизна настоящей программы заключается в том, что она интегрирует в себе достижения современных и инновационных направлений в малой беспилотной авиации.

Направленность. 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.

Занятия внеурочной деятельности будут проводиться на базе Центра образования цифрового и гуманитарного профилей «Точка роста», созданного в целях развития и реализации основных и дополнительных общеобразовательных программ цифрового, естественнонаучного и гуманитарного профилей, формирования социальной культуры, проектной деятельности, направленной не только на расширение познавательных интересов школьников, но и на стимулирование активности, инициативы и исследовательской деятельности обучающихся.

Педагогическая целесообразность настоящей программы заключается в том, что после ее освоения обучающиеся получают знания и умения, которые позволят им понять основы устройства беспилотного летательного аппарата, принципы работы всех его систем и их взаимодействия, а также управление БПЛА. Использование различных инструментов развития soft-skills у детей (игропрактика, командная работа) в сочетании с развитием у них hard-компетенций (workshop, tutorial) позволит сформировать у ребенка целостную систему знаний, умений и навыков.

1.2. Цель и задачи программы

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

Задачи:

Обучающие:

- 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.

Развивающие:

- 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.

Воспитательные:

- воспитывать аккуратность и дисциплинированность при выполнении работы;
- способствовать формированию положительной мотивации к трудовой деятельности;
- способствовать формированию опыта совместного и индивидуального творчества при выполнении командных заданий;
- воспитывать трудолюбие, уважение к труду;
- формировать чувство коллективизма и взаимопомощи;
- воспитывать чувство патриотизма, гражданственности, гордости за достижения отечественной науки и техники.

1.3. Планируемые результаты освоения программы и способы их проверки

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.

Предметные результаты

В результате освоения программы обучающиеся должны

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.

Формы подведения итогов реализации дополнительной программы

Подведение итогов реализуется в рамках следующих мероприятий: тестирование по программированию на языке Python, защита результатов выполнения кейса № 4, групповые соревнования.

Формы демонстрации результатов обучения

Представление результатов образовательной деятельности пройдет в форме публичной презентации решений кейсов командами и последующих ответов выступающих на вопросы наставника и других команд.

Формы диагностики результатов обучения

Беседа, тестирование, опрос.

Объем программы.

Занятие является основной формой работы с детьми, продолжительность 2 занятий для учащихся составляет 2 академических часа по 45 минут. Количество запланированных занятий - 72 часа. В занятия включены физкультминутки, которые позволяют детям расслабиться.

1.4. Содержание программы

Учебный план

№ п/п	Название раздела, темы	Количество часов			Дата проведения	Формы аттестации/контроля
		Всего	Теория	Практика		
1.	Введение в образовательную программу, техника безопасности	1	1	-	14.10	Тестирование
2.	Основы языка Python. Примеры на языке Python с разбором конструкций: циклы, условия, ветвления, массивы, типы данных	4	2	2	21.10 28.10 4.11 11.11	Тестирование
3.	Кейс 1. «Угадай число»	10	5	5		Демонстрация решений кейса
3.1	Введение в искусственный интеллект. Примеры на языке Python с искусственным интеллектом по угадыванию чисел, метод дихотомии. Управление искусственным интеллектом	8	4	4	18.11 25.11 2.12 9.12 16.12 23.12 30.12 6.01	
3.2	Подготовка к публичному выступлению для защиты результатов. Демонстрация отчёта в группе и защита результатов работы	2	1	1	13.01 20.01	
4.	Кейс 2. «Спаси остров»	10	3	7		Демонстрация решений кейса
4.1	Работа на языке Python со словарями и списками, множественное присваивание, добавление элементов в список и их удаление	4	2	2	27.01 3.02 10.02 17.02	
4.2	Планирование дизайна и механики игры. Создание главного меню игры, подсчёта очков	2	1	1	24.02 2.03	
4.3	Визуализация программы в виде блок-схемы	2	-	2	9.03 16.03	

4.4	Тестирование написанной программы и доработка. Подготовка к публичному выступлению для защиты результатов. Демонстрация результатов работы	2	1	1	23.03 30.03	
5.	Кейс 3. «Калькулятор»	10	2	8		Демонстрация решений кейса
5.1	Постановка проблемы, генерация путей решения	2	1	1	6.04 13.04	
5.2	Создание простейшего калькулятора с помощью библиотеки Tkinter	4	-	4	20.04 27.04 4.05 11.05	
5.3	Тестирование написанной программы и доработка	2	-	2	18.05 25.05	
5.4	Подготовка к публичному выступлению для защиты результатов. Демонстрация результатов работы	2	1	1	1.06 8.06	
6.	Кейс 4. Программирование автономных квадрокоптеров	37	13	24		Демонстрация решений кейса
6.1	Техника безопасности при полётах. Проведение полётов в ручном режиме	2	1	1		
6.2	Программирование взлёта и посадки беспилотного летательного аппарата	4	1	3		
6.3	Выполнение команд «разворот», «изменение высоты», «изменение позиции»	6	2	4		
6.4	Выполнение группового полёта вручную	2	0	2		
6.5	Выполнение позиционирования по меткам	8	2	6		
6.6	Программирование группового полёта	7	3	4		
6.7	Программирование роевого взаимодействия	8	3	5		
	Итого:	72	26	46		

Содержание учебного плана

№	Темы занятий	Содержание занятий
1.	Вводное занятие. Введение в предмет, техника безопасности (1 ч)	Теория: введение в образовательную программу. Ознакомление обучающихся с программой, приёмами и формами работы. Вводный инструктаж по ТБ.
2.	Основы языка Python. Примеры на языке Python с разбором конструкций: циклы, условия, ветвления, массивы, типы данных (4 ч)	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.
3	Кейс «Угадай число»	
3.1	Введение в искусственный интеллект. Примеры на языке Python с искусственным интеллектом по угадыванию чисел, метод дихотомии. Управление искусственным интеллектом (6 ч)	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.
3.2	Подготовка к публичному выступлению для защиты результатов. Демонстрация отчёта в группе и защита результатов работы (2 ч)	Теория: создание удобной и понятной презентации. Практика: подготовка презентации для защиты. Подготовка речи для защиты.
4	Кейс «Спаси остров»	
4.1	Работа на языке Python со словарями и списками, множественное присваивание, добавление элементов в список и их удаление (4 ч)	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.
4.2	Планирование дизайна и	Теория: понятие «механика игры»,

	механики игры. Создание главного меню игры, подсчёта очков (2 ч)	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.
4.3	Визуализация программы в виде блок-схемы (2 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.
4.4	Тестирование написанной программы и доработка (1 ч)	Практика: тестирование созданной игры-программы, доработка и расширение возможностей.
4.5	Подготовка к публичному выступлению для защиты результатов. Демонстрация результатов работы (1 ч)	Практика: подготовка презентации и речи для защиты. Презентация созданной программы.
5	Кейс «Калькулятор»	
5.1	Оформление проектной идеи. Формирование программы работ (1 ч)	Теория: знакомство с кейсом, представление поставленной проблемы. Практика: мозговой штурм. Анализ проблемы, генерация и обсуждение методов её решения.
5.2	Программа для работы калькулятора (2 ч)	Практика: написание программы для будущего калькулятора.
5.3	Создание внешнего вида калькулятора (2 ч)	Практика: создание внешнего вида калькулятора.
5.4	Тестирование написанной программы и доработка (2 ч)	Практика: тестирование созданной программы, доработка и расширение возможностей.
5.5	Подготовка к публичному выступлению для защиты результатов (2 ч)	Практика: подготовка презентации и речи для защиты.
5.6	Демонстрация результатов работы (1 ч)	Практика: презентация созданной программы.
6	Кейс «Программирование автономных квадрокоптеров»	
6.1	Техника безопасности при полётах. Проведение полётов в ручном режиме (2 ч)	Теория: знакомство с кейсом, представление поставленной проблемы, правила техники безопасности. Изучение конструкции квадрокоптеров. Практика: полёты на квадрокоптерах в ручном режиме.
6.2	Программирование взлёта и посадки беспилотного летательного аппарата (4 ч)	Теория: основы программирования квадрокоптеров на языке Python. Практика: тестирование написанного кода в режимах взлёта и посадки.
6.3	Выполнение команд	Теория: теоретические основы выполнения

	«разворот», «изменение высоты», «изменение позиции» (6 ч)	разворота, изменения высоты и позиции на квадрокоптерах. Практика: тестирование программного кода в режимах разворота, изменения высоты и позиции.
6.4	Выполнение группового полёта вручную (2 ч)	Практика: выполнение группового полёта на квадрокоптере в ручном режиме.
6.5	Выполнение позиционирования по меткам (8 ч)	Теория: основы позиционирования indoor и outdoor квадрокоптеров. Практика: тестирование режима позиционирования по ArUco - маркерам.
6.6	Программирование группового полёта (7 ч)	Теория: основы группового полёта квадрокоптеров. Изучение типов группового поведения роботов. Практика: программирование роя квадрокоптеров для группового полёта.
6.7	Программирование роевого взаимодействия (6 ч)	Теория: основы программирования роя квадрокоптеров. Практика: Выполнение группового полета в автоматическом режиме.

Содержание тем программы

Кейс 1. «Угадай число»

При решении данного кейса обучающиеся осваивают основы программирования на языке Python посредством создания игры, в которой пользователь угадывает число, заданное компьютером.

Программа затрагивает много ключевых моментов программирования: конвертирование типов данных, запись и чтение файлов, использование алгоритма деления отрезка пополам, обработка полученных данных и представление их в виде графиков.

Кейс 2. «Спаси остров»

Кейс позволяет обучающимся поработать на языке Python со словарями и списками; изучить, как делать множественное присваивание, добавление элементов в список и их удаление, создать уникальный дизайн будущей игры.

Кейс 3. «Калькулятор»

При решении данного кейса учащиеся создают первое простое приложение калькулятор: выполняют программную часть на языке программирования Python и создают интерфейс для пользователя при помощи библиотеки Tkinter.

Кейс 4. Программирование автономных квадрокоптеров

Роевое взаимодействие роботов является актуальной задачей в современной робототехнике. Квадрокоптеры можно считать летающей робототехникой. Шоу квадрокоптеров, выполнение задания боевыми беспилотными летательными аппаратами - такие задачи решаются с помощью применения алгоритмов роевого взаимодействия.

Данный кейс посвящен созданию шоу коптеров из 3х бпла выполняющих полет в автономном режиме. Обучающиеся получают первые навыки программирования технической системы на языке Python. Познакомятся с алгоритмами позиционирования устройств на улице и в помещении, а также узнают о принципах работы оптического распознавания объектов.

1.5. ФОРМЫ АТТЕСТАЦИИ И ОЦЕНОЧНЫЕ МАТЕРИАЛЫ

Прямыми критериями оценки результатов обучения служит успешное усвоение дополнительной образовательной программы, отзывы детей и родителей об отношениях к занятиям, анализ и др.

Для отслеживания результативности образовательной деятельности по программе проводятся: входная диагностика, текущий контроль, промежуточная аттестация, итоговый контроль.

Входная диагностика — оценка стартового уровня образовательных возможностей обучающихся при поступлении в объединение, проводится в начале учебного года. Форма проведения: беседа.

Текущий контроль — оценка уровня и качества освоения программы и личностных качеств учащихся проводится после изучения каждой темы. Текущий контроль проводится в форме визуального контроля, оценки выполнения практических заданий.

Промежуточная аттестация проводится в конце первого полугодия с целью выявления уровня усвоения программы. Формы контроля: педагогическое наблюдение, практическая работа, тестирование (**Приложение 1**).

Итоговый контроль — оценка уровня и качества освоения учащимися программы по завершению обучения, проводится в конце учебного года. Формы контроля: отзывы, защита творческих работ, соревнования, коллективная рефлексия.

МОНИТОРИНГ

Время проведения	Цель проведения	Формы контроля
Входная диагностика	Определение уровня развития детей, их знаний в проектировании и создании VR-приложений	Беседа
Текущий контроль, промежуточная аттестация	<p>Определение степени усвоения обучающимися учебного материала.</p> <p>Определение готовности обучающихся к восприятию нового материала.</p> <p>Повышение ответственности и заинтересованности воспитанников в обучении.</p> <p>Выявление детей, отстающих и опережающих обучение.</p> <p>Подбор наиболее эффективных методов и средств обучения.</p>	Практические задания, презентация творческих работ, тестирование
В конце учебного года	<p>Определение изменения уровня развития детей, их творческих способностей.</p> <p>Определение результатов обучения.</p> <p>Ориентирование обучающихся на дальнейшее (в том числе самостоятельное) обучение.</p> <p>Получение сведений для совершенствования образовательной программы и методов обучения.</p>	Отзывы, защита творческих работ, соревнования, коллективная рефлексия

2. Комплекс организационно-педагогических условий

2.1. Методическое обеспечение

Образовательный процесс по данной программе предполагает очное обучение. Для обучения детей используются разнообразные методы и приемы.

- **Кейс-метод.** Задается ситуация (реальная или максимально приближенная к реальности). Ученики должны исследовать ситуацию, предложить варианты ее разрешения, выбрать лучшие из возможных решений.
- **Метод проектов** предполагает самостоятельный анализ заданной ситуации и умение находить решение проблемы. Проектный метод объединяет исследовательские, поисковые, творческие методы и приемы обучения по ФГОС.

Педагогические технологии.

Данная программа основывается на решении кейс-технологии и технологии проектной деятельности, которые подразумевают коллективную работу в малых группах.
Алгоритм учебного занятия

1. Установление взаимосвязей
2. Конструирование
3. Рефлексия
4. Развитие

Дидактические материалы

Данная программа предполагает наличие следующего раздаточного материала:

- рабочие бланки ученика;
- рабочие листы ученика;
- технологические карты занятия;
- материалы для учителя;

Кадровые условия реализации программы

Комплектование образовательной организации педагогическими, руководящими и иными работниками, соответствующими квалификационным характеристикам по соответствующей должности.

Требования к кадровым ресурсам:

- укомплектованность образовательного учреждения педагогическими, руководящими и иными работниками;
- уровень квалификации педагогических, руководящих и иных работников образовательного учреждения;

- непрерывность профессионального развития педагогических и руководящих работников образовательного учреждения, реализующего основную образовательную программу.

Компетенции педагогического работника, реализующего основную образовательную программу:

- обеспечивать условия для успешной деятельности, позитивной мотивации, а также самомотивирования обучающихся;
- осуществлять самостоятельный поиск и анализ информации с помощью современных информационно-поисковых технологий;
- организовывать и сопровождать учебно-исследовательскую и проектную деятельность обучающихся, выполнение ими индивидуального проекта;
- интерпретировать результаты достижений обучающихся;
- навык программирования на языке Python;
- использовать библиотеку Tkinter;
- навык создания компьютерных игр и приложений;
- проектирование интерфейса пользователей;
- поиск и интеграция библиотек программного кода с открытых источников типа GitHub в собственный проект;
- навык работы в специализированном ПО для создания презентаций.

2.2. Условия реализации программы

Аппаратное и техническое обеспечение:

- Рабочее место обучающегося:
ноутбук: производительность процессора (по тесту PassMark - CPU BenchMark<http://www.cpubenchmark.net/>): не менее 2000 единиц; объем оперативной памяти: не менее 4 Гб; объем накопителя SSD/eMMC: не менее 128 Гб (или соответствующий по характеристикам персональный компьютер с монитором, клавиатурой и колонками).
- рабочее место преподавателя:
ноутбук: процессор Intel Core i5-4590/AMD FX 8350 аналогичная или более новая модель, графический процессор NVIDIA GeForce GTX 970, AMD Radeon R9 290 аналогичная или более новая модель, объем оперативной памяти: не менее 4 Гб, видеовыход HDMI 1.4, DisplayPort 1.2 или более новая модель (или соответствующий по характеристикам персональный компьютер с монитором, клавиатурой и колонками);
- компьютеры должны быть подключены к единой сети Wi-Fi с доступом в интернет;
- презентационное оборудование (проектор с экраном) с возможностью подключения к компьютеру — 1 комплект;
- флипчарт с комплектом листов/маркерная доска, соответствующий набор письменных принадлежностей — 1 шт.;
- квадрокоптер DJI Ryze tello — не менее 3 шт.;
- поле меток;
- Wi-Fi роутер.

Программное обеспечение:

- компилятор Python 3.5;
- веб-браузер;
- пакет офисного ПО;
- текстовый редактор.

2.3. Оценочные материалы

Тесты

Тест 1 :

1. Какие из приведенных стилей программирования поддерживает язык Python?
 - a) Процедурный
 - b) Объектно-ориентированный
 - c) Функциональный
 - d) Смешанный
2. Каким способом можно объявлять переменные в Python:
 - a) `a=5`
 - b) `a=int (5)`
 - c) `int a=5`
3. Какая функция отвечает за вывод на экран?
 - a) `cout<<a`
 - b) `out (a)`
 - c) `print (a)`
4. Какая функция отвечает за открытие файла?
 - a) `file()`
 - b) `open()`
 - c) `open_file()`
5. В каком из вариантов присутствует ошибка?
 - a) `a=5`
`print ('a')`
 - b) `while True`
`print(a)`
 - c) `a=open("file.txt")`
6. Что делает команда `import`
 - a) импортирует файл модуля
 - b) создает функцию
 - c) удаляет файл
7. Выберите вариант правильного удаления переменной `a`
 - a) `del(a)`
 - b) `delete(a)`
 - c) `delete=a`
8. Какое значение `1//2` вернет выражение в среде IDLE?
 - a) 0
 - b) 0.5
 - c) 0.50
9. Как называется встроенный в языке Python тип данных неупорядоченной коллекции из нуля или более пар ключ-значение?
 - a) `dict`
 - b) `set`

- c) list
- d) frozenset

10. Если предположить, что класс Mydict наследует класс dict, то каким класс dict является по отношению к классу Mydict?

- a) дочерним
- b) подклассом
- c) базовым

Практическая работа №2

ПЛАН По дисциплине: Язык программирования Python Тема занятия: Работа с условным оператором
Цель занятия: Изучить работу условных операторов Количество часов: 4

Содержание работы (Задание, Задачи): Задание: По вводимой дате (число и месяц), подсчитать порядковый номер этого дня в году (високосный год не учитывать). Определить сезон года. Обязательно проверить правильность ввода числа данного месяца (например, 32 января, 29 февраля, 31 апреля). Методические указания по выполнению: if выражение1: оператор1 [elif выражение2: оператор2] [elif выражение3: оператор3] [elif выражениеN: операторN...] [else: операторN+1]

```
if x<0: print 'Число отрицательное' elif x==0: print 'Число равно нулю' else: print 'Число положительное'
```

Ветвь elif может, как совсем отсутствовать, так и присутствовать несколько раз; наличие ветви else необязательно. Ключевое слово elif является короткой формой для else if. Последовательность if...elif...elif...эквивалентна инструкциям switch и case в других языках

Вопросы для защиты лабораторной работы: Конструкции условных операторов if и else Ветвление условных операторов if и else

2.4 Список литературы

Для педагога

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.ryzerobotics.com/downloads/tello/0222/Tello+Scratch+Readme.pdf>.

Для детей

Лекции от «Коптер-экспресс»

<https://youtu.be/GtwG5ajQJvA?t=1344>

<https://www.youtube.com/watch?v=FF6z-bCo3T0>

<http://alexgyver.ru/quadcopters/>