

Муниципальное общеобразовательное учреждение
«Средняя общеобразовательная школа №3 г. Ершова Саратовской области»

| | |
|--|---|
| ПРИНЯТА на заседании педагогического совета МОУ «Средняя общеобразовательная Школа №3 г. Ершова Саратовской области» Протокол № 13 от 19.04.2023 | УТВЕРЖЕНО директор МОУ «Средняя общеобразовательная Школа №3 г. Ершова Саратовской области» Приказ № 140 от 24.04.2023  А.В. Широкова |
|--|---|

Дополнительная общеобразовательная общеразвивающая
программа технической направленности
«Программирование на Python»
(стартовый уровень)

Направленность: техническая
Форма реализации: очная
Возраст обучающихся: 12 - 17 лет
Срок реализации: 2 года

Автор – составитель:
Бурова Ольга Валерьевна
педагог дополнительного
образования

Ершов, 2023

Раздел №1. «Комплекс основных характеристик программы»

1.1. Пояснительная записка

Дополнительная общеобразовательная общеразвивающая программа «Программирование на Python» (базовый уровень) муниципального общеобразовательного учреждения «Средняя общеобразовательная школа № 3 г. Ершова Саратовской области» разработана в рамках **технической направленности** в соответствии с:

- «Закон об образовании в Российской Федерации» (№ 273-ФЗ от 29 декабря 2012 г.);
- «Порядок организации и осуществления образовательной деятельности по дополнительным общеобразовательным программам» (пр. Министерства просвещения РФ от 27 июля 2022 г. №629)
- Приказ Министерства образования и науки Российской Федерации от 23 августа 2017 года №816 «Об утверждении Порядка применения организациями, осуществляющими образовательную деятельность, электронного обучения, дистанционных образовательных технологий при реализации образовательных программ»;
- Постановление Главного государственного санитарного врача РФ от 28 сентября 2020 года №28 «Об утверждении СанПиН 2.4.4.3648-20 «Санитарно-эпидемиологические требования к организациям воспитания и обучения, отдыха и оздоровления детей и молодежи»
- Положением о дополнительной общеобразовательной общеразвивающей программе МОУ «СОШ № 3 г. Ершова Саратовской области»

Актуальность программы обусловлена потребностью общества в технически грамотных специалистах и полностью отвечает социальному заказу по подготовке квалифицированных кадров в области программирования, а также высоким интересом подростков к IT-сфере. Важнейшей характеристикой подрастающего поколения является активность в информационном пространстве, интернет-коммуникации. Всё большее значение приобретает умение человека грамотно обращаться с компьютером, причём зачастую не на пользовательском уровне, а на уровне начинающего программиста.

Новизна. В обязательном школьном курсе информатики программирование нередко представлено лишь на элементарном уровне, на это выделяется недостаточное количество часов. Лишь немногие школы могут себе позволить преподавать программирование на достойном уровне. Следствием этого является формальное восприятие обучающимися основ современного программирования и неумение применять полученные знания на практике. В рамках изучения программы обучающиеся постоянно будут сталкиваться с необходимостью самостоятельной работы над заданиями: обучающиеся учатся решать задачи без помощи преподавателя. Для этого в содержании курса фигурируют задания, в которых для решения задачи необходимо найти информацию в сети Интернет; может потребоваться устранение ошибки, которую не так просто обнаружить; условие

сформулировано недостаточно прозрачно и ученику необходимо самостоятельно формализовать его (или задать правильные вопросы преподавателю). Все эти знания, умения и практические навыки решения актуальных задач, полученные на занятиях, готовят обучающихся к самостоятельной проектно-исследовательской деятельности с применением современных технологий.

Педагогическая целесообразность программы «Программирование на Python (базовый уровень)» заключается в том, что она отражает требования и актуальные тенденции не только сегодняшнего, но и завтрашнего дня и имеет междисциплинарный характер, что полностью отражает современные тенденции построения как дополнительных общеобразовательных программ, так и образования в целом.

Ознакомление с фундаментальными понятиями алгоритмизации и программирования на доступном уровне; имеет практическую направленность с ориентацией на реальные потребности, соответствующие возрасту обучающегося; охватывает как алгоритмическое направление, так и вопросы практического использования полученных знаний при решении задач из различных областей знаний; ориентирована на существующий парк вычислительной техники и дополнительные ограничения; допускает возможность варьирования в зависимости от уровня подготовки и интеллектуального уровня обучающихся (как группового, так и индивидуального), а также предусматривает возможность индивидуальной работы с обучающимися. Практическая значимость курса заключается в том, что он способствует более успешному овладению знаниями и умениями по направлению «Программирование» через развитие самостоятельности обучающихся и оптимизацию средств и методов обучения.

Отличительная особенность программы «Программирование на Python (базовый уровень)» в том, что она является практико-ориентированной. Освоение подростками IT-навыков происходит в процессе практической и самостоятельной работы. Это позволяет обучающимся получать не только теоретические знания в области программирования, но и уверенно овладевать IT-технологиями, что поможет им самоопределиться и выстроить траекторию личностного роста в современном информационном обществе.

Изучение основных принципов программирования невозможно без регулярной практики написания программ на каком-либо языке. Для обучения был выбран язык Python. Данный выбор обусловлен тем, что синтаксис языка достаточно прост и интуитивно понятен, а это понижает порог вхождения и позволяет сосредоточиться на логических и алгоритмических аспектах программирования, а не на выучивании тонкостей синтаксиса. При этом Python является очень востребованным языком; он отлично подходит для знакомства с различными современными парадигмами программирования и активно применяется в самых разных областях от разработки веб-приложений до машинного обучения.

Научившись программировать на языке Python, обучающиеся получат мощный и удобный инструмент для решения как учебных, так и

прикладных задач. Вместе с тем чистота и ясность его конструкций позволит обучающимся потом с лёгкостью выучить любой другой язык программирования.

Знания и умения, приобретённые в результате освоения курса, могут быть использованы обучающимися при сдаче ЕГЭ, при участии в олимпиадах по программированию, при решении задач по физике, химии, биологии, лингвистике и другим наукам, а также они являются фундаментом для дальнейшего совершенствования мастерства программирования.

Адресат программы, возрастные особенности: дополнительная общеобразовательная общеразвивающая программа «Программирование на Python» (базовый уровень) предназначена для подростков в возрасте 12–17 лет, мотивированных к обучению и обладающих системным мышлением.

Подростки этого возраста отличаются открытием своего внутреннего мира, внутреннего «Я». Главным измерением времени в самосознании является будущее, к которому он (она) себя готовит. Ведущая деятельность в этом возрасте — учебно-профессиональная, в процессе которой формируются такие новообразования, как мировоззрение, профессиональные интересы, самосознание, мечта и идеалы, поэтому основной формой проведения занятий выбраны практические занятия.

Также следует отметить, что подростки данной возрастной группы характеризуются такими психическими процессами, как стремление углублённо понять себя, разобраться в своих чувствах, настроениях, мнениях, отношениях. Это порождает у подростка стремление к самоутверждению, самовыражению (проявления себя в тех качествах, которые он считает наиболее ценными) и самовоспитанию. Эти процессы позволяют положить начало созданию начального профессионального самоопределения обучающихся.

Наполняемость группы: 12-15 человек

Объем и срок реализации программы определяется содержанием программы и составляет 2 года (288 часов, 144 часа в год).

Режим занятий: длительность одного занятия составляет 2 академических часа, периодичность занятий – 2 раза в неделю.

Так как в течение учебного года возникает непреодолимая сила, или форс-мажор – обстоятельства (эпидемия, карантин, погодные условия и прочее), не позволяющие осуществлять обучение в обычной (очной) форме, реализация программы возможна с помощью электронных (дистанционных) технологий.

1.2. Цели и задачи программы

Цель программы: творческая самореализация личности ребёнка посредством получения навыков разработки эффективных алгоритмов, для реализации их в виде программы, написанной на языке программирования Python.

Задачи:

Обучающие:

- изучение конструкций языка программирования Python;
- знакомство с принципами и методами функционального и объектно-ориентированного программирования; основными структурами данных и типовыми методами обработки этих структур;
- приобретение навыков работы в интегрированной среде разработки на языке Python.

Развивающие:

- формирование навыков разработки эффективных алгоритмов и программ на основе изучения языка программирования Python; формирование и развитие навыков алгоритмического и логического мышления, грамотной разработки программ;
- приобретение навыков поиска информации в сети Интернет, анализ выбранной информации на соответствие запросу, использование информации при решении задач;
- развитие у обучающихся интереса к программированию, самостоятельности и творческого подхода к решению задач с использованием средств вычислительной техники;
- формирование и развитие навыков работы с различными источниками информации, необходимой для решения учебных задач; умения планировать свои действия с учётом фактора времени, в обстановке с элементами конкуренции, предвидеть результат и достигать его, при необходимости вносить коррективы в первоначальный замысел.

Воспитательные:

- формирование активной жизненной позиции, гражданско-патриотической ответственности;
- воспитание этики групповой работы, отношений делового сотрудничества, взаимоуважения;
- развитие основ коммуникативных отношений внутри проектных групп и в коллективе в целом;
- воспитание упорства в достижении результата;
- пропаганда здорового образа жизни;
- формирование целеустремлённости, организованности, неравнодушия, ответственного отношения к труду, толерантности и уважительного отношения к окружающим.

1.3. Планируемые результаты

Предметные результаты:

□ □ умение определять результат выполнения алгоритма при заданных исходных данных, узнавать изученные алгоритмы обработки чисел и числовых последовательностей, создавать на их основе несложные программы анализа данных, читать и понимать несложные программы, написанные на выбранном для изучения универсальном алгоритмическом языке высокого уровня;

□ □ понимание основных предметных понятий («информация», «алгоритм», «исполнитель», «модель») и их свойств;

□□ развитие логических способностей и алгоритмического мышления, умения составить и записать алгоритм для конкретного исполнителя, знакомство с основными алгоритмическими структурами – линейной, условной и циклической;

□ умение выполнять пошагово (с использованием компьютера или вручную) алгоритмы управления исполнителями и анализа числовых и текстовых данных;

□ навыки и опыт разработки программ в выбранной среде программирования, включая тестирование и отладку программ; умение использовать основные управляющие конструкции объектно-ориентированного программирования и библиотеки прикладных программ, выполнять созданные программы;

□ умение разрабатывать и использовать компьютерно-математические модели, оценивать числовые параметры моделируемых объектов и процессов, интерпретировать результаты, получаемые в ходе моделирования реальных процессов, анализировать готовые модели на предмет соответствия реальному объекту или процессу;

□ умение соблюдать нормы информационной этики и права.

Личностные результаты:

□ формирование ответственного отношения к учению, способности довести до конца начатое дело аналогично завершённым творческим учебным проектам;

□ формирование способности к саморазвитию и самообразованию средствами информационных технологий на основе приобретённой благодаря иллюстрированной среде программирования мотивации к обучению и познанию;

□□ развитие опыта участия в социально значимых проектах, повышение уровня самооценки благодаря реализованным проектам;

□ формирование коммуникативной компетентности в общении и сотрудничестве со сверстниками в процессе образовательной, учебно-исследовательской и проектной деятельности, участия в конкурсах и конференциях различного уровня;

□ формирование целостного мировоззрения, соответствующего современному уровню развития информационных технологий;

□ формирование осознанного позитивного отношения к другому человеку, его мнению, результату его деятельности;

□□ формирование ценности здорового и безопасного образа жизни; усвоение правил индивидуального и коллективного безопасного поведения при работе с компьютерной техникой.

Метапредметные результаты:

□□ умение самостоятельно ставить и формулировать для себя новые задачи, развивать мотивы своей познавательной деятельности;

□ умение самостоятельно планировать пути решения поставленной проблемы для получения эффективного результата; понимание, что в программировании длинная программа не всегда лучшая;

□ умение критически оценивать правильность решения учебно-исследовательской задачи;

□ умение корректировать свои действия, вносить изменения в

программу и отлаживать её в соответствии с изменяющимися условиями;

владение основами самоконтроля, способность к принятию решений;

умение создавать, применять и преобразовывать знаки и символы,

модели и схемы для решения учебно-исследовательских и проектных работ;

формирование и развитие компетентности в области использования информационно-коммуникационных технологий (ИКТ-компетенция);

умение организовывать учебное сотрудничество и совместную деятельность с учителем и сверстниками в процессе проектной и учебно-исследовательской деятельности.

Планируемые результаты первого года обучения

Предметные результаты:

знание необходимой терминологии («информация», «алгоритм»),

«исполнитель», «модель»), смысла этих понятий и умение применять полученные знания на практике;

развитие у обучающихся алгоритмического (знакомство и навыки работы с простейшими алгоритмическими структурами – линейной, условной и циклической), а также логического мышления, что необходимо для грамотного составления алгоритмов, рассчитанных для конкретного исполнителя;

знание основных понятий и этапов проектной деятельности;

- навыки пошагового выполнения алгоритмов управления исполнителями и анализа числовых и текстовых данных, умение осуществлять данные операции как вручную, так и с использованием компьютера;

знания основ программирования и областей применения полученных навыков.

Личностные результаты:

развитие у обучающихся внимания, сосредоточенности, терпения;

использование принципов здоровьесбережения, а также отработка на практике принципов индивидуального и коллективного безопасного поведения при работе с компьютерной техникой;

формирование осознанного уважительного отношения к другому человеку, его мнению, своему и чужому труду, бережное отношение к используемому оборудованию;

развитие коммуникативных навыков, умения работать в команде сверстников в процессе образовательной, учебно-исследовательской и проектной деятельности.

Метапредметные результаты:

умение самостоятельно планировать последовательность своих действий для достижения поставленных целей, а также грамотно распределять свое время и ресурсы для получения максимально эффективного результата;

умение работать в паре и в коллективе, выстраивать совместную деятельность как с педагогом, так и со сверстниками;

умение осуществлять самостоятельный поиск информации, анализировать и обобщать её;

способность к принятию решений, а также умение формулировать, аргументировать и отстаивать своё мнение.

Планируемые результаты второго года обучения

Предметные результаты:

- опыт разработки и использования компьютерно-математических моделей, а также умение интерпретировать и анализировать полученные результаты на предмет их соответствия или несоответствия реальному объекту или процессу;
- умение соблюдать сетевой этикет, другие базовые нормы информационной этики и права при работе с компьютерными программами и в сети Интернет;
- умение выполнять созданные программы, осуществлять их разработку, тестирование и отладку, используя изученный язык программирования;
- получение опыта использования базовых управляющих конструкций объектно-ориентированного программирования;
- умение самостоятельно создавать программы анализа данных на выбранном для этого универсальном языке программирования;
- умение определять результат выполнения алгоритма при заданных исходных данных.

Личностные результаты:

- навыки формулирования и обоснования собственного цельного мнения о возможностях, преимуществах и недостатках предлагаемого программного продукта;
- формирование ответственного отношения к обучению, подбор и комбинирование имеющихся умений программиста для решения учебных задач;
- формирование целостного мировоззрения, технического мышления, соответствующих как возрасту обучающегося, так и современному уровню развития информационных технологий;
- опыт распределения задач между членами группы и умение совместно принимать решения внутри данной группы;
- развитие опыта участия в социально значимых проектах (опыт создания и оценки проекта, повышение уровня самооценки благодаря реализованным проектам).

Метапредметные результаты:

- умение осуществлять самостоятельный поиск информации, анализировать и обобщать её;
- 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 – формирование и развитие компетентности в области использования информационно-коммуникационных технологий);
- умение формулировать, аргументировать и отстаивать своё мнение, а также критически оценивать правильность решения учебно-

исследовательской задачи;

умение работать в паре и коллективе;

умение корректировать свои действия, планировать последовательность шагов для достижения целей, а также вносить необходимые изменения в процессе работы

1.4. Содержание общеразвивающей программы

Учебный план 1 года обучения

| № п/п | Название раздела, темы | Кол-во часов | | | Формы аттестации/ контроля |
|---|--|--------------|-----------|-----------|--|
| | | Всего | Теория | Практика | |
| Базовый уровень | | | | | |
| Модуль 1. Введение в программирование. | | 26 | 7 | 19 | |
| 1.1 | Вводное занятие. Инструктаж по ТБ. Введение в программу 1-го года обучения | 8 | 2 | 6 | Знакомство. Опрос. Инструктаж по ТБ |
| 1.2 | Интегрированные среды, исполнение кода и отладка | 10 | 2 | 8 | Опрос, беседа, выполнение мини-проекта |
| 1.3 | Переменные, основные операторы | 6 | 2 | 4 | Опрос, беседа, выполнение мини-проекта |
| 1.4 | Базовые типы данных, ветвления | 1 | 1 | | Беседа. Опрос |
| 1.5 | Контрольное тестирование | 1 | | 1 | Тест (Приложение 4) |
| Модуль 2. Базовые конструкции в Python | | 42 | 6 | 36 | |
| 2.1 | Циклы, срезы, списочные выражения. | 12 | 2 | 10 | Опрос, беседа, выполнение мини-проекта |
| 2.2 | Методы списков и строк. Функции | 8 | 2 | 6 | Опрос, беседа, выполнение мини-проекта |
| 2.3 | Решение задач по пройденным темам. Практическая работа | 10 | 2 | 8 | Практическое занятие |
| 2.4 | Контрольная работа по темам 1-го и 2-го модулей | 12 | | 12 | Практическое задание |
| Модуль 3. Решение прикладных задач в Python. | | 76 | 28 | 48 | |

| | | | | | |
|--------------|--|------------|-----------|------------|--|
| 3.1 | Функциональный стиль программирования, лямбда функции, модуль functools | 8 | 4 | 4 | Опрос, беседа, выполнение мини-проекта |
| 3.2 | Хэш-таблицы. Модуль Collections | 6 | 2 | 4 | Практическое задание |
| 3.3 | Обработка исключений | 4 | 2 | 2 | Практическое задание |
| 3.4 | Принципы устройства и механика создания модулей и пакетов | 6 | 4 | 2 | Беседа, выполнение мини-проекта |
| 3.5 | ООП: инкапсуляция, наследование, полиморфизм | 6 | 2 | 4 | Практическое задание |
| 3.6 | Основы объектно-ориентированного анализа и дизайна, шаблоны проектирования | 4 | 2 | 2 | Опрос, беседа, выполнение мини-проекта |
| 3.7 | Итераторы и генераторы | 4 | 2 | 2 | Практическое задание |
| 3.8 | Автоматизированное тестирование в python | 6 | 2 | 4 | Практическое задание |
| 3.9 | Tkinter: создание графического интерфейса | 8 | 4 | 4 | Опрос, беседа, выполнение мини-проекта |
| 3.10 | Голосовые помощники и чат боты. Создание навыка для Алисы Яндекса | 8 | 4 | 4 | Опрос, беседа, выполнение мини-проекта |
| 3.11 | Проверочные работы и итоговая контрольная работа по темам модулей 1, 2 и 3 | 16 | | 16 | Практическое задание |
| Итого | | 144 | 41 | 103 | |

Содержание учебного плана 1 года обучения.

Модуль 1. Введение в программирование

Тема 1.1 Вводное занятие. Инструктаж по ТБ. Введение в программу 1-го года обучения

Теория: Инструктажи по технике безопасности с отметкой в журнале. Введение в программу «Программирование на Python» 1-го года обучения. Понятия кода, интерпретатора, программы. Простейшие программы с выводом на экран. Знакомство с виртуальной средой взаимодействия – регистрация, организация личного кабинета, поиск и выкладывание материалов; знакомство с системой автоматизированной проверки задачи системой Яндекс.

Практика: решение задач.

Тема 1.2 Интегрированные среды, исполнение кода и отладка

Теория: интегрированные среды, исполнение кода. Основные понятия программирования: исполнитель, система команд, алгоритм,

программа, среда разработки.

Практика: решение задач.

Тема 1.3 Переменные, основные операторы

Теория: условный оператор. Переменные и арифметика. Погружений условия. Ввод-вывод в программе, условный оператор, оператор цикла с предусловием. Простейшие программы с использованием условного оператора if, оператора циклов while и операторов ввода-вывода. Технология разработки программы.

Практика: решение задач.

Тема 1.4 Базовые типы данных, ветвления

Теория: знакомство со списками, строками, множествами и кортежами в Python. Практика: разработка алгоритмов и программ, определение работоспособности разработанной программы.

Тема 1.5 Контрольное тестирование

Модуль 2. Базовые конструкции в Python

Тема 2.1 Циклы, срезы, списочные выражения

Теория: понятие о языке высокого уровня Python. Структура программы, переменные и константы, работа с числовыми переменными, арифметические операторы в Python. Основные управляющие конструкции алгоритмов с ветвлением в Python. Знакомство со срезами и диапазонами. Равенство и совпадение объектов. Устройство циклов for. Основные управляющие конструкции циклического алгоритма в Python. Простейшие циклы и циклы с переменными.

Практика: решение задач по теме «Срезы и диапазоны»

Тема 2.2 Методы списков и строк. Функции

Теория: списочные выражения. Методы split и join. Другие методы списков и строк. Знакомство с функциями. Области видимости переменных. Работа со списками, строками, множествами и кортежами в Python. Понятие итератора, подпрограммы, процедуры, функции. Функции в Python.

Практика: решение задач.

Тема 2.3 Решение задач по пройденным темам. Практическая работа

Теория: повторение пройденных тем.

Практика: практическая работа.

Тема 2.4 Контрольная работа по темам 1-го и 2-го модулей

Практика: решение задач контрольной работы на методы списков и строк, по темам «Методы split и join», «Функции», «Области видимости переменных». Анализ результатов.

Модуль 3. Решение прикладных задач в Python

Тема 3.1 Функциональный стиль программирования, лямбда-функции, модуль functools

Теория: функции, функциональная парадигма программирования. Понятие лямбда функции и области ее применения. Функции модуля functools.

Практика: решение задач.

Тема 3.2 Хэш-таблицы. Модуль Collections

Теория: Словари и множества. Подключение и использование модулей стандартной библиотеки Python. Функции модуля Collections.

Практика: решение задач.

Тема 3.3 Обработка исключений

Теория: Понятие исключения, синтаксис их обработки. Применение исключений при разработке и отладке программ.

Практика: решение задач.

Тема 3.4 Принципы устройства и механика создания модулей и пакетов

Теория: Понятие модуля и пакета, размещение и импорт модуля.

Практика: создание собственного пакета модулей.

Тема 3.5 ООП: инкапсуляция, наследование, полиморфизм

Теория: Причины появления, принципы и основные сущности объектно-ориентированного подхода к разработке ПО. Инкапсуляция, полиморфизм, наследование, композиция.

Практика: решение задач.

Тема 3.6 Основы объектно-ориентированного анализа и дизайна, шаблоны проектирования

Теория: Восходящий и нисходящий способы разработки. Методика объектно-ориентированного дизайна приложения. Шаблоны проектирования.

Практика: проектирование и реализация компьютерной игры.

Тема 3.7 Итераторы и генераторы

Теория: Понятие и реализация итераторов и генераторов. Предназначение, особенности устройства и работы, типовые сферы применения.

Практика: решение задач.

Тема 3.8 Автоматизированное тестирование в python

Теория: Виды тестирования и наиболее распространенные системные подходы автоматического тестирования. Модульные тесты.

Практика: написание серии модульных тестов для выбранной задачи.

Тема 3.9 Tkinter: создание графического интерфейса

Теория: Знакомство с библиотекой tkinter, изучение основных команд, виджетов, упаковщиков и приемов работы.

Практика: создание приложения с графическим интерфейсом.

Тема 3.10 Голосовые помощники и чат боты. Создание навыка для Алисы Яндекса

Теория: Изучение общих принципов работы голосовых помощников.

Знакомство с платформой Яндекс.Диалоги и протоколом работы навыка.

Практика: изучение документации на платформу, создание навыка для Алисы.

Тема 3.11 Проверочные работы и итоговая контрольная работа по темам модулей 1, 2 и 3

Теория: подготовка к контрольной работе. Разбор контрольной работы.

Практика: выполнение проверочных работ. Итоговая контрольная работа по темам модулей 1, 2 и 3. Разработка проекта по индивидуальному заданию, составление отчёта о выполнении индивидуальной или совместной работы.

Учебный план 2 года обучения

| № п/п | Название раздела, темы | Кол-во часов | | | Формы аттестации/ контроля |
|---|---|--------------|-----------|-----------|---|
| | | Всего | Теория | Практика | |
| Модуль 1. Программирование на Python | | 40 | 17 | 23 | |
| 1.0 | Вводное занятие. Введение в программу 2-го года обучения Инструктаж по ТБ | 2 | 1 | 1 | Опрос. Отметка о прослушанных инструктажах в журнале. |
| 1.1 | Повторение. Решение задач на основные конструкции и структуры данных. Решение задач на классы и библиотеки | 4 | 3 | 1 | Опрос. Самостоятельная работа |
| 1.2 | Решение задач по теме «Повторение» | 2 | 0 | 2 | Самостоятельная работа |
| 1.3 | QT 1. Что такое QT и PyQT | 2 | 2 | 0 | Опрос. Беседа. Мини-проект |
| 1.4 | QT 2. QtDesigner | 4 | 2 | 2 | Опрос. Беседа |
| 1.5 | QT 3. Файлы в Python. Типы файлов и работа с ними. Внутреннее устройство файлов | 4 | 2 | 2 | Опрос |
| 1.6 | QT 4. Диалоги, работа с изображениями | 4 | 2 | 2 | Опрос. Беседа. Мини-проект |
| 1.7 | QT 5. Работа с простыми таблицами (csv). Работа с табличными данными в PyQT | 4 | 2 | 2 | Практическое задание |
| 1.8 | QT 6. Введение в БД, работа с SQL-таблицами и отображение данных в PyQT. Часть 1 | 4 | 2 | 2 | Практическое задание |

| | | | | | |
|------------------------------------|--|-----------|-----------|-----------|-------------------------------|
| 1.9 | QT 7. Введение в БД, работа с SQL-таблицами и отображение данных в PyQT. Часть 2 | 4 | 0 | 4 | Опрос. Беседа. Мини-проект |
| 1.10 | QT 8. Обработка событий. Сборка независимого приложения | 4 | 1 | 3 | Практическое задание |
| 1.11 | Защита проекта QT | 2 | 0 | 2 | Рубежный контроль |
| Модуль 2. PyGame | | 44 | 17 | 27 | |
| 2.1 | Введение в репозитории. Подключение PyCharm. Работа с удаленными репозиториями | 4 | 1 | 3 | Опрос. Беседа |
| 2.2 | PyGame 1. Введение | 2 | 1 | 1 | Опрос. Беседа |
| 2.3 | PyGame 2. Игровой цикл. События | 4 | 2 | 2 | Практическое задание |
| 2.4 | PyGame 3. Основные команды при одиночной работе с Git | 4 | 2 | 2 | Самостоятельная работа |
| 2.5 | PyGame 4. Клеточное поле | 4 | 1 | 3 | Опрос. Проектная деятельность |
| 2.6 | PyGame 5. Классические игры на клеточном поле | 4 | 2 | 2 | Самостоятельная работа |
| 2.7 | PyGame 6. Изображения. Спрайты | 2 | 1 | 1 | Опрос. Беседа |
| 2.8 | PyGame 7. Столкновения и другие взаимодействия | 4 | 2 | 2 | Опрос. Беседа |
| 2.9 | PyGame 8. Игра в целом | 6 | 2 | 4 | Проектная деятельность |
| 2.10 | PyGame 9. Украшения игр | 4 | 2 | 2 | Практическое задание |
| 2.11 | Цели и подходы к тестированию. Создание "самодельных" тестов (без библиотек) | 4 | 1 | 3 | Проектная деятельность |
| 2.12 | Защита проекта PyGame | 2 | 0 | 2 | Рубежный контроль |
| Модуль 3. Проекты WEB и API | | 60 | 22 | 38 | |
| 3.1 | WEB. Работа с популярными форматами файлов (json, xml) | 4 | 2 | 2 | Опрос. Беседа |
| 3.2 | WEB. Знакомство с API | 4 | 2 | 2 | Практическое задание |
| 3.3 | WEB. Понятие исключения, обработка исключений. Собственные исключения. | 4 | 2 | 2 | Самостоятельная работа |
| 3.4 | WEB. Работа с командной строкой (скрипты, аргументы). Периодические задачи (модуль schedule) | 4 | 2 | 2 | Опрос. Беседа |

| | | | | | |
|------|--|------------|-----------|-----------|----------------------------|
| 3.5 | WEB. Библиотека argparse. Задачи на создание скриптов с ее помощью | 2 | 0 | 2 | Опрос. Беседа. |
| 3.6 | WEB. Работа с протоколом HTTP | 4 | 2 | 2 | Самостоятельная работа |
| 3.7 | WEB. Решение задач на API Яндекс-карт | 4 | 2 | 2 | Практическое задание |
| 3.8 | WEB. Введение во flask. Обработка html-форм | 4 | 2 | 2 | Самостоятельная работа |
| 3.9 | WEB. Шаблоны. flask--wtf | 4 | 2 | 2 | Опрос. Беседа. Мини-проект |
| 3.10 | WEB. Знакомство с flask-sqlalchemy | 4 | 2 | 2 | Опрос. Беседа. |
| 3.11 | WEB. Flask-sqlalchemy | 2 | 0 | 2 | Рубежный контроль |
| 3.12 | WEB. REST-API. Понятие. Делаем простое REST-API. | 4 | 2 | 2 | Практическое задание |
| 3.13 | WEB. REST-API. Flask-restfull | 2 | 0 | 2 | Проектная деятельность |
| 3.14 | WEB. Разворачиваем проект в облаке. Дорешка. | 4 | 0 | 4 | Проектная деятельность |
| 3.1 | WEB. Алиса. Часть 1 | 4 | 2 | 2 | Опрос. Беседа |
| 3.16 | WEB. Алиса. Часть 2 | 4 | 0 | 4 | Практическое задание |
| 3.17 | WEB. Защита проекта WebServer + API | 2 | 0 | 2 | Рубежный контроль |
| 3.12 | Итого программа года | 144 | 56 | 88 | |

Содержание учебного плана 2 года обучения.

Модуль 1. Программирование на Python

Тема 1.0 Вводное занятие. Инструктажи по технике безопасности с отметкой в журнале. **Введение в программу 2-го года обучения.** Общая характеристика программы. Цели и задачи. Тематические модули.

Практика: решение задач.

Тема 1.1 Повторение. Решение задач на основные конструкции структуры данных. Решение задач на классы и библиотеки

Теория: Повторение основных конструкций и структур данных. Повторение модулей стандартной библиотеки Python.

Практика: решение задач.

Тема 1.2 Решение задач по теме «Повторение»

Практика: решение задач в рамках самостоятельной работы.

Тема 1.3 QT 1. Что такое QT и PyQt

Теория: Обзор рынка фреймворков для разработки настольных приложений на Python. История, предназначение, преимущества и недостатки фреймворка Qt. Подключение и первые шаги с PyQt.

Практика: решение задач.

Тема 1.4 QT 2. QtDesigner

Теория: Изучение основных элементов графического интерфейса и настроек QtDesigner, способы интеграции графического и текстового стили создания настольных приложений.

Практика: решение задач, связанных с созданием простейшего однооконного приложения.

Тема 1.5 QT 3. Файлы в Python. Типы файлов и работа с ними.

Внутреннее устройство файлов

Теория: Принципы работы с файловой системой средствами Python. Чтение и запись информации в файл приложением, созданным при помощи QtDesigner.

Практика: решение задач, связанных с работой с файлами из настольного приложения.

Тема 1.6 QT 4. Диалоги, работа с изображениями

Теория: Изучение набора встроенных в QtDesigner диалоговых окон. Способы чтения, модификации и внедрения графических файлов в настольное приложение.

Практика: решение задач по обработке изображений.

Тема 1.7 QT 5. Работа с простыми таблицами (csv). Работа с табличными данными в PyQt

Теория: Формат данных csv, его преимущества и недостатки. Способы чтения и записи файлов в данном формате. Работа с табличными данными в PyQt в рамках собственного настольного приложения.

Практика: решение задач по работе с таблицами.

Тема 1.8 QT 6. Введение в БД, работа с SQL-таблицами и отображение данных в PyQt. Часть 1

Теория: Предназначение и разновидности баз данных. Реляционные базы данных и язык запросов SQL. Обращение к базе данных из создаваемого при помощи PyQt настольного приложения.

Практика: решение задач по работе с БД и SQL.

Тема 1.9 QT 7. Введение в БД, работа с SQL-таблицами и отображение данных в PyQt. Часть 2

Теория: Технология написания более сложных SQL запросов. Объектно-реляционное отображение.

Практика: решение задач на работу с БД из PyQt.

Тема 1.10 QT 8. Обработка событий. Сборка независимого приложения.

Теория: Петля событий настольного приложения. Обработка событий стандартных устройств ввода (мышь, клавиатура). Обработка событий, генерируемых модулями программы. Сборка интерактивного приложения.

Практика: работа над проектом.

Тема 1.11 Защита проекта QT

Практика: защита проекта (настольное приложение).

Модуль 2. PyGame

Тема 2.1 Введение в репозитории. Подключение PyChart. Работа с удаленными репозиториями.

Теория: Разновидности и предназначение систем контроля версий. Репозитории для хранения и обмена данными. Использование среды разработки PyCharm с удаленными репозиториями.

Практика: решение задач по работе с репозиторием.

Тема 2.2 PyGame 1. Введение

Теория: Что такое PyGame, его место в экосистеме и какие игры можно создавать с его помощью.

Практика: решение задач.

Тема 2.3 PyGame 2. Игровой цикл. События

Теория: Создание игрового цикла, обработка событий стандартных устройств ввода.

Практика: решение задач.

Тема 2.4 PyGame 3. Основные команды при одиночной работе с Git

Теория: Система контроля версий Git. Использование git при разработке приложения в одиночку — графический и консольный интерфейс, основные команды.

Практика: решение задач.

Тема 2.5 PyGame 4. Клеточное поле

Теория: Реализация клеточного поля в PyGame. Способы хранения глобальных настроек и обмена данными между игровыми агентами.

Практика: решение задач.

Тема 2.6 PyGame 5. Классические игры на клеточном поле

Теория: Обзор классических игр на клеточном поле и вариантов реализации при помощи PyGame.

Практика: решение задач по реализации игры на клеточном поле.

Тема 2.7 PyGame 6. Изображения. Спрайты

Теория: Графические возможности PyGame, чтение, изменение и встраивание изображений. Техника работы со спрайтами.

Практика: решение задач по работе с изображениями.

Тема 2.8 PyGame 7. Столкновения и другие взаимодействия

Теория: Обсуждение основных видов взаимодействия между игроками и другими сущностями игры, а также способов программной реализации такого рода взаимодействий.

Практика: решение задач.

Тема 2.9 PyGame 8. Игра в целом

Теория: Проектирование архитектуры игрового приложения.

Практика: создание своей игры.

Тема 2.10 PyGame 9. Украшения игр

Теория: Нюансы создания интерфейса игры, визуального оформления игровой механики, подключение звуковых эффектов.

Практика: украшение своей игры.

Тема 2.11 Цели и подходы к тестированию. Создание "самодельных" тестов (без библиотек)

Теория: Как и зачем осуществляют тестирование в промышленной

разработке. Подходы к тестированию программных продуктов. Технология создания тестов стандартными средствами Python.

Практика: тестирование своего игрового приложения.

Тема 2.12 Защита проекта PyGame

Практика: защита проекта (игра).

Модуль 3. Проекты WEB и API

Тема 3.1 WEB. Работа с популярными форматами файлов (json, xml)

Теория. Структура форматов json и csv. Python-модули для работы с ними. Основные сферы применения, практика применения в WEB.

Практика: решение задач.

Тема 3.2 WEB. Знакомство с API

Теория: Программный интерфейс приложения в разработке для основных существующих платформ. Особенности API в WEB-разработке.

Практика: решение задач.

Тема 3.3 WEB. Понятие исключения, обработка исключений.

Собственные исключения.

Теория: Программные ошибки, понятие исключительной ситуации.

Механизм исключений в Python, различные практики его использования.

Практика: решение задач - самостоятельная работа.

Тема 3.4 WEB. Работа с командной строкой (скрипты, аргументы).

Периодические задачи (модуль schedule)

Теория: Интерфейс командной строки основных операционных систем. Его применение в backend-разработке и администрировании.

Практика: решение задач.

Тема 3.5 WEB. Библиотека argparse. Задачи на создание скриптов с ее помощью

Теория: Парсинг аргументов командной строки средствами argparse.

Практика: решение задач.

Тема 3.6 WEB. Работа с протоколом HTTP

Теория: Протокол HTTP и клиент-серверная архитектура. Обмен данными с использованием данного протокола.

Практика: решение задач - самостоятельная работа.

Тема 3.7 WEB. Решение задач на API Яндекс-карт

Теория: Картографические сервисы Яндекса для сайтов, мобильных приложений и логистики. Работа с ними средствами Python применительно к WEB-проектам.

Практика: решение задач.

Тема 3.8 WEB. Введение во flask. Обработка html-форм

Теория: Фреймворк для создания WEB-приложений Flask, структура и базовая механика. Создание и обработка HTML-форм с его помощью.

Практика: решение задач - самостоятельная работа.

Тема 3.9 WEB. Шаблоны. flask--wtf

Теория: Расширения фреймворка Flask. Предназначение и использование расширения Flask-WTF.

Практика: решение задач.

Тема 3.10 WEB. Знакомство с flask-sqlalchemy

Теория: Предназначение и использование расширения (фреймворка Flask) Flask-SQLAlchemy.

Практика: решение задач.

Тема 3.11 WEB. Flask-sqlalchemy

Теория: Нюансы Flask-SQLAlchemy.

Практика: решение задач — контрольная работа.

Тема 3.12 WEB. REST-API. Понятие. Делаем простое REST-API

Теория: REST подход к организации взаимодействия компонентов распределенного приложения. REST-API.

Практика: решение задач.

Тема 3.13 WEB. REST-API. Flask-restful

Теория: Расширение (фреймворка Flask) Flask-RESTful для быстрого построения REST-API.

Практика: работа над проектом — создание своего web-сервиса. **Тема 3.14 WEB. Разворачиваем проект в облаке. Дорешка** *Теория:* Размещение web-проекта в сети интернет.

Практика: работа над проектом.

Тема 3.15 WEB. Алиса. Часть 1

Теория: История развития и основные технологии, лежащие в основе голосовых помощников. Яндекс.Диалоги. Алиса.

Практика: решение задач.

Тема 3.16 WEB. Алиса. Часть 2 *Теория:* Создание навыка для Алисы. *Практика:* решение задач.

Тема 3.17 WEB. Защита проекта WebServer + API

Практика: защита своего web-проекта.

1.5. Формы аттестации и контроля

Система отслеживания результатов обучающихся выстроена следующим образом:

- определение начального уровня знаний, умений и навыков;
- промежуточный контроль;
- итоговый контроль.

Входным контролем при приёме на первый год обучения по данной общеразвивающей программе – успешное предварительное тестирование (школа анализа данных Яндекса (Яндекс.Лицей)).

Текущий контроль осуществляется путём наблюдения, определения качества выполнения, отслеживания динамики развития обучающегося. Способы проверки уровня освоения тем: опрос, тестирование, решение задач, наблюдение, оценка выполненных самостоятельных работ.

Итоговая аттестация осуществляется в форме презентации, защиты проекта.

Раздел №2. «Комплекс организационно-педагогических условий реализации общеразвивающей программы»

2.1. Методическое обеспечение

Методические пособия, разработанные преподавателем с учётом конкретных задач, варианты демонстрационных программ, материалы по терминологии ПО, инструкции по настройке оборудования, учебная и техническая литература. Используются педагогические технологии индивидуализации обучения и коллективной деятельности.

Электронное приложение к учебникам К. Ю. Полякова «Информатика и ИКТ». Набор цифровых образовательных ресурсов – дидактические материалы, интерактивные тесты, анимационные плакаты.

Образовательный процесс осуществляется в очной форме.

В образовательном процессе используются следующие **методы**:

1. Объяснительно-иллюстративный;
2. Метод проблемного изложения (постановка проблемы и решение её самостоятельно или группой);
3. Проектно-исследовательский;
4. Наглядный:
 - демонстрация плакатов, схем, таблиц, диаграмм;
 - использование технических средств;
 - просмотр кино- и телепрограмм;
5. Практический:
 - практические задания;
 - анализ и решение проблемных ситуаций и т. д.

Выбор методов обучения осуществляется исходя из анализа уровня готовности обучающихся к освоению содержания модуля, степени сложности материала, типа учебного занятия. В связи с тем, что в работу по программе включаются подростки, уже имеющие первичные навыки программирования первый год обучения не имеет стартового уровня освоения. Поэтому вместе с объяснительно-иллюстративным используются, а также становятся ведущими частично-поисковый и проектно-исследовательский методы, на продвинутом уровне (2-ой год обучения) – проблемного изложения и проектно-исследовательский. На выбор методов обучения значительно влияет персональный состав группы, индивидуальные особенности, возможности и запросы детей.

Формы обучения:

фронтальная – предполагает работу педагога сразу со всеми обучающимися в едином темпе и с общими задачами. Для реализации обучения используется компьютер педагога с мультимедиа проектором, посредством которых учебный материал демонстрируется на общий экран. Активно используются Интернет-ресурсы;

коллективная – это форма сотрудничества, при котором коллектив обучает каждого своего члена и каждый член коллектива активно участвует в обучении своих товарищей по совместной учебной работе;

□□ **групповая** – предполагает, что занятия проводятся с подгруппой. Для этого группа распределяется на подгруппы не более 6 человек, работа в которых регулируется педагогом;

□□ **индивидуальная** – подразумевает взаимодействие преподавателя с одним обучающимся. Как правило данная форма используется в сочетании с фронтальной. Часть занятия (объяснение новой темы) проводится фронтально, затем обучающийся выполняют индивидуальные задания или общие задания в индивидуальном темпе;

□□ **дистанционная** – взаимодействие педагога и обучающихся между собой на расстоянии, отражающее все присущие учебному процессу компоненты. Для реализации дистанционной формы обучения весь дидактический материал размещается в свободном доступе в сети Интернет, происходит свободное общение педагога и обучающихся в социальных сетях, по электронной почте, посредством видеоконференции или в общем чате. Кроме того, дистанционное обучение позволяет проводить консультации обучающегося при самостоятельной работе дома.

Формы организации учебного занятия:

В образовательном процессе помимо традиционного учебного занятия используются многообразные формы, которые несут учебную нагрузку и могут использоваться как активные способы освоения детьми образовательной программы, в соответствии с возрастом обучающихся, составом группы, содержанием учебного модуля: беседа, лекция, мастер-класс, практическое занятие, защита проектов, конкурс, викторина, диспут, круглый стол, «мозговой штурм», воркшоп, глоссирование, деловая игра, квиз, экскурсия.

Некоторые формы проведения занятий могут объединять несколько учебных групп или весь состав объединения, например, экскурсия, викторина, конкурсы и т. д.

Методы воспитания: мотивация, убеждение, поощрение, упражнение, стимулирование, создание ситуации успеха и др.

Педагогические технологии: индивидуализации обучения; группового обучения; коллективного взаимообучения; дифференцированного обучения; разноуровневого обучения; проблемного обучения; развивающего обучения; дистанционного обучения; игровой деятельности; коммуникативная технология обучения; коллективной творческой деятельности; решения изобретательских задач; здоровьесберегающая технология.

Дидактические материалы:

Методические пособия, разработанные преподавателем с учётом конкретных задач, варианты демонстрационных программ, материалы по терминологии ПО, инструкции по настройке оборудования, учебная и техническая литература. Используются педагогические технологии индивидуализации обучения и коллективной деятельности.

2.2. Условия реализации программы

Материально-техническое обеспечение

Требования к помещению:

- помещение для занятий, отвечающие требованиям СанПин для учреждений дополнительного образования;
- качественное освещение;
- столы, стулья по количеству обучающихся и 1 рабочим местом для педагога.

Оборудование:

- компьютеры и ноутбуки на каждого обучающегося и преподавателя;
- проекционное оборудование (экраны) – 2 шт.
- маркерная доска – 1 шт.

Расходные материалы:

- whiteboard маркеры;
- бумага писчая;
- шариковые ручки;
- permanent маркеры.

Информационное обеспечение:

- операционная система;
- поддерживаемые браузеры (для работы LMS): Yandex Browser, Chrome, Chrome Mobile, Firefox, Opera, Safari, Mobile Safari, Edge, Python;
- среда Wing IDE 101 (версии 6 или выше);
- среда PyCharm Community Edition;
- пакет PyQt4 (на Qt5);
- пакет библиотек со SciPy: numpy, scipy, matplotlib, ipython +ipythonnotebook, sympy, pandas;
- рекомендуется установить ПО Anaconda.

Кадровое обеспечение: педагог дополнительного образования, знакомый с технологией обучения основам программирования на языке Python.

2.3. Календарный учебный график 1-го года обучения
дополнительной общеобразовательной общеразвивающей программы «Программирование на Python (базовый уровень)»

| № п/п | Месяц | Число | Время проведения | Форма занятия | Кол-во часов | Тема занятия | Место проведения | Форма контроля |
|-------|-------|-------|------------------|---------------|--------------|--|--|------------------------------------|
| 1-8 | | | | Практикум | 8 | Вводное занятие. Инструктаж по ТБ. Введение в программу 1-го года обучения | Учебный кабинет Компьютерный класс https://telemost.yandex.ru/ | Практическое задание, тестирование |
| 9-18 | | | | Практикум | 10 | Интегрированные среды, исполнение кода и отладка | | Практическое задание, тестирование |
| 19-24 | | | | Практикум | 6 | Переменные, основные операторы | | Практическое задание, тестирование |
| 25 | | | | Практикум | 1 | Базовые типы данных, ветвления | | Практическое задание, тестирование |
| 26 | | | | Практикум | 1 | Контрольное тестирование | | Практическое задание, тестирование |
| 27-38 | | | | Практикум | 12 | Циклы, срезы, списочные выражения. | | Практическое задание, тестирование |
| 39-46 | | | | Практикум | 8 | Методы списков и строк. Функции | | Практическое задание, тестирование |
| 47-56 | | | | Практикум | 10 | Решение задач по пройденным темам. Практическая работа | | Практическое задание, тестирование |
| 57-68 | | | | Практикум | 12 | Контрольный практикум по темам 1-го и 2-го модулей | | Практическое задание, тестирование |

| | | | | | | | | |
|---------|--|--|--|-----------|----|---|--|------------------------------------|
| 69-76 | | | | Практикум | 8 | Функциональный стиль программирования, лямбдафункции, модуль functools | Учебный кабинет Компьютерный класс https://telemost.yandex.ru/ | Практическое задание, тестирование |
| 77-82 | | | | Практикум | 6 | Хэш-таблицы. Модуль Collections | | Практическое задание, тестирование |
| 83-86 | | | | Практикум | 4 | Обработка исключений | | Практическое задание, тестирование |
| 87-92 | | | | Практикум | 6 | Принципы устройства и механика создания модулей и пакетов | | Практическое задание, тестирование |
| 93-98 | | | | Практикум | 6 | ООП: инкапсуляция, наследование, полиморфизм | | Практическое задание, тестирование |
| 99-102 | | | | Практикум | 4 | Основы объектно- ориентированного анализа и дизайна, шаблоны проектирования | | Практическое задание, тестирование |
| 103-106 | | | | Практикум | 4 | Итераторы и генераторы | | Практическое задание, тестирование |
| 107-112 | | | | Практикум | 6 | Автоматизированное тестирование в python | | Практическое задание, тестирование |
| 113-120 | | | | Практикум | 8 | Tkinter: создание графического интерфейса | | Практическое задание, тестирование |
| 121-128 | | | | Практикум | 8 | Голосовые помощники и чатботы. Создание навыка для Алисы Яндекса | | Практическое задание, тестирование |
| 129-144 | | | | Практикум | 16 | Проверочные работы и итоговая контрольная работа по темам | Практическое задание, тестирование | |

Календарный учебный график 2-го года обучения

дополнительной общеобразовательной общеразвивающей программы «Программирование на Python (базовый уровень)»

| № п/п | Месяц | Число | Время проведения | Форма занятия | Кол-во часов | Тема занятия | Место проведения | Форма контроля |
|-------|-------|-------|------------------|---------------|--------------|---|--|------------------------------------|
| 1-2 | | | | Практикум | 2 | Вводное занятие. Введение в программу 2-го года обучения Инструктаж по ТБ | Учебный кабинет Компьютерный класс https://telemost.yandex.ru/ | Практическое задание, тестирование |
| 3-6 | | | | Практикум | 4 | Повторение. Решение задач на основные конструкции и структуры данных. Решение задач | | Практическое задание, тестирование |
| 7-8 | | | | Практикум | 2 | Решение задач по теме «Повторение» | | Практическое задание, тестирование |
| 9-10 | | | | Практикум | 2 | QT 1. Что такое QT и PyQT | | Практическое задание, тестирование |
| 11-14 | | | | Практикум | 4 | QT 2. QtDesigner | | Практическое задание, тестирование |
| 15-18 | | | | Практикум | 4 | QT 3. Файлы в Python. Типы файлов и работа с ними. Внутреннее устройство файлов | | Практическое задание, тестирование |
| 19-22 | | | | Практикум | 4 | QT 4. Диалоги, работа с изображениями | | Практическое задание, тестирование |
| 23-26 | | | | Практикум | 4 | QT 5. Работа с простыми таблицами (csv). Работа с табличными данными в PyQT | | Практическое задание, тестирование |
| 27-30 | | | | Практикум | 4 | QT 6. Введение в БД, работа с SQL-таблицами и отображение данных в PyQT. Часть | | |
| 31-34 | | | | Практикум | 4 | QT 7. Введение в БД, работа с SQL-таблицами и отображение данных в PyQT. Часть 2 | | |

| | | | | | | | | |
|-------|--|--|--|-----------|---|---|--|------------------------------------|
| 35-38 | | | | Практикум | 4 | QT 8. Обработка событий. Сборка независимого приложения | Учебный кабинет Компьютерный класс https://telemost.yandex.ru/ | Практическое задание, тестирование |
| 39-40 | | | | Практикум | 2 | Защита проекта QT | | Практическое задание, тестирование |
| 41-44 | | | | Практикум | 4 | Введение в репозитории. Подключение PyCharm. Работас удаленными репозиториями | | Практическое задание, тестирование |
| 45-46 | | | | Практикум | 2 | PyGame 1. Введение | | Практическое задание, тестирование |
| 47-50 | | | | Практикум | 4 | PyGame 2. Игровой цикл. События | | Практическое задание, тестирование |
| 51-54 | | | | Практикум | 4 | PyGame 3. Основные командыпри одиночной работе с Git | | Практическое задание, тестирование |
| 55-58 | | | | Практикум | 4 | PyGame 4. Клеточное поле | | Практическое задание, тестирование |
| 59-62 | | | | Практикум | 4 | PyGame 5. Классические игрына клеточном поле | | Практическое задание, тестирование |
| 63-64 | | | | Практикум | 2 | PyGame 6. Изображения. Спрайты | | Практическое задание, тестирование |
| 65-68 | | | | Практикум | 4 | PyGame 7. Столкновения идругие взаимодействия | | Практическое задание, тестирование |
| 69-74 | | | | Практикум | 6 | PyGame 8. Игра в целом | | Практическое задание, тестирование |
| 75-78 | | | | Практикум | 4 | PyGame 9. Украшения игр | | Практическое задание, тестирование |
| 79-82 | | | | Практикум | 4 | Цели и подходы к тестированию. Создание "самодельных" тестов | | Практическое задание, тестирование |
| 83-84 | | | | Практикум | 2 | Защита проекта PyGame | | Практическое задание, тестирование |

2.4. Оценочные материалы

Оценочные материалы образовательных результатов

| Показатели (оцениваемые параметры) | Степень выраженности оцениваемого качества | Число баллов | Методы диагностики |
|---|---|-----------------|---|
| Теоретические знания по разделам/темам учебно- тематического плана программы | овладел менее чем ½ объема знаний, предусмотренных программой | 1 | Наблюдение, тестирование, защита работы и др. |
| | объем усвоенных знаний составляет более ½ | 2 | |
| | освоил практически весь объем знаний, предусмотренных программой за конкретный период | 3 | |
| Практические умения и навыки, предусмотренные программой | овладел менее чем ½ предусмотренных умений и навыков | 1 | Наблюдение, защита работы |
| | объем усвоенных умений и навыков составляет более ½ | 2 | |
| | овладел умениями и навыками, предусмотренными программой за конкретный период | 3 | |

Оценочные материалы личностных результатов

| Показатели (оцениваемые параметры) | Степень выраженности оцениваемого качества | Число баллов | Методы диагностики |
|--|---|-----------------|-----------------------|
| Сформированность активности, организаторских способностей | мало активен, наблюдает за деятельностью других, забывает выполнить задание. Результативность невысокая | 1 | Наблюдение |
| | активен, проявляет стойкий познавательный интерес, трудолюбив, добивается хороших результатов | 2 | |
| | активен, проявляет стойкий познавательный интерес, добивается выдающихся результатов, инициативен, организует деятельность других | 3 | |
| Сформированность коммуникативных навыков, коллективизма | поддерживает контакты избирательно, чаще работает индивидуально, публично не выступает | 1 | Наблюдение |
| | вступает и поддерживает контакты, не вступает в конфликты, дружелюбен со всеми, по инициативе руководителя или группы выступает перед аудиторией | 2 | |
| | легко вступает и поддерживает контакты, разрешает конфликты, дружелюбен со всеми, инициативен, по собственному желанию успешно выступает перед аудиторией | 3 | |
| Сформированность ответственности, самостоятельности, дисциплинированности | неохотно выполняет поручения. Начинает работу, но часто не доводит ее до конца. | 1 | Наблюдение |

| | | | |
|--|--|---|-------------------|
| | справляется с поручениями и соблюдает правила поведения только при наличии контроля и требовательности преподавателя; выполняет поручения охотно, ответственно. Хорошо ведет себя независимо от наличия или отсутствия контроля, но не требует этого от других | 2 | |
| | выполняет поручения охотно, ответственно, часто по собственному желанию, может привлечь других. Всегда дисциплинирован, везде соблюдает правила поведения, требует того же от других | 3 | |
| Сформированность креативности, склонности к самостоятельному творчеству | может работать в проектно-исследовательской группе при постоянной поддержке и контроле. Способен принимать творческие решения, но в основном использует традиционные способы | 1 | Наблюдение |
| | может разработать свой творческий проект с помощью педагога. Способен на творческие решения, но в основном использует традиционные способы | 2 | |
| | высокий творческий потенциал. Самостоятельно выполняет работы. Находит нестандартные решения, новые способы выполнения заданий | 3 | |

Оценочные материалы метапредметных результатов

| Показатели (оцениваемые параметры) | Степень выраженности оцениваемого качества | Число баллов | Методы диагностики |
|--|---|---------------------|---------------------------|
| Понимать и принимать учебную задачу, сформулированную педагогом | овладел менее чем ½ объема задач, предусмотренных программой | 1 | Наблюдение |
| | объем усвоенных задач составляет более ½ | 2 | |
| | демонстрирует полное понимание, предусмотренных программой задач за конкретный период | 3 | |
| Планировать свои действия на отдельных этапах работы над выполнением творческого задания | овладел менее чем ½ объема знаний, предусмотренных программой | 1 | Наблюдение |
| | демонстрирует неполное освоение планируемых действий, но более ½ | 2 | |
| | освоил план действий в заданных условиях | 3 | |

| | | | |
|--|---|---|------------|
| Осуществлять контроль, коррекцию и оценку результатов своей деятельности; понимать и применять полученную информацию при выполнении заданий | знает, но избегает их употреблять в деятельности | 1 | Наблюдение |
| | демонстрирует неполное освоение заданных параметров, но более ½ | 2 | |
| | освоил план действий в заданных условиях | 3 | |

Мониторинг результатов обучения ребенка по дополнительной общеразвивающей программе

| Показатели (оцениваемые параметры) | Методы диагностики |
|--|---|
| 1. Уровни знаний / пониманий <ul style="list-style-type: none"> ▪ Наличие общих представлений (менее ½ объема знаний) ▪ Наличие ключевых понятий (объем усвоенных знаний более 1/2) ▪ Наличие прочных системных знаний, (освоен практически весь объем) | Наблюдение, тестирование, контрольный опрос, собеседование |
| 2. Уровни умения применять знания на практике <ul style="list-style-type: none"> ▪ Репродуктивный несамостоятельный (деятельность осуществляется под непосредственным контролем преподавателя на основе устных и письменных инструкций). ▪ Репродуктивный самостоятельный (деятельность осуществляется на основе типовых алгоритмов). ▪ Творческий (в процессе деятельности творчески используются знания, умений, предлагаются и реализуются оригинальные решения) | Контрольное задание |
| 3. Наличие опыта самостоятельной деятельности <ul style="list-style-type: none"> ▪ Очень незначительный опыт; ▪ Незначительный балл (от случая к случаю); ▪ Эпизодическая деятельность; ▪ Периодическая деятельность; ▪ Богатый опыт (систематическая деятельность) | Анализ, исследовательские работы, конкурсные работы, наблюдение |
| 4. Сформированность личностных качеств <ul style="list-style-type: none"> ▪ Очень низкая (проявились отдельные элементы); ▪ Низкая (проявилась частично); ▪ Недостаточно высокая (проявилась в основном); ▪ Высокая (проявились полностью) | Анализ, наблюдение, собеседование |

На основе вышеприведенного анализа заполняется диагностическая карта (оценочный лист) таблица 2.

Диагностическая карта успеваемости воспитанников объединения

| Ф.И.О. | Знать / понимать (макс-3 балла) | | | | | Уметь использовать (макс-4 балла) | | | | | Владеть опытом (макс-5 баллов) | | | | | Личностные качества (макс-4 балла) | | | | | Итого баллов | Оценка |
|-----------|---------------------------------|---|---|---|---|-----------------------------------|---|---|---|---|--------------------------------|---|---|---|---|------------------------------------|---|---|---|---|--------------|--------|
| | 1 | 2 | 3 | 4 | 5 | 1 | 2 | 3 | 4 | 5 | 1 | 2 | 3 | 4 | 5 | 1 | 2 | 3 | 4 | 5 | | |
| Иванов А. | | | | | | | | | | | | | | | | | | | | | | |

Результаты деятельности каждого обучающегося по каждому из показателей суммируются для определения итогового балла. Показатель усвоения (продуктивности обучения) вычисляется по формуле:

$$K_{\text{усв}} = \frac{\Phi}{\Pi} * 100\%$$

Где K усв- коэффициент усвоения

Φ – фактический объем знаний (набранная сумма баллов)

Π – полный объем знаний (максимальная сумма баллов).

В дальнейшем можно перейти к пятибалльной системе оценки.

Коэффициент сформированности:
80-100 - «отлично»
50-79 - «хорошо»
30-49 - «удовлетворительно»
Менее 29 - «неудовлетворительно»

Примерные темы мини-проектов 1-ый год обучения.

1. Разработка игры «Шахматы» для двух игроков с консольным интерфейсом
2. Разработка игры «Змейка» с графическим интерфейсом (Tkinter)
3. Разработка игры «Футбол» с графическим интерфейсом
4. Создание приложения для автоматической генерации писем и документов по шаблону MS Word
5. Создание консольного графического редактора (применение фильтров, изменение размеров, повороты изображения и пр.)

Примерные темы мини-проектов 2-ый год обучения.

1. Разработка приложения «Текстовый редактор»
2. Разработка приложения «Графический редактор»
3. Разработка игры «Гексагон» с графическим интерфейсом
4. Создание чат бота
5. Разработка игры для трех игроков с возможностью добавления ботов
6. Создание WEB-проекта, предоставляющего для команд пользователей совместное рабочее пространство

Критерии оценки мини-проектов 1-ый год обучения

№ группы: _____

Дата: _

| № п/п | ФИО обучающегося | Соответствие теме (по шкале от 0 до 5 баллов) | Сложность алгоритмов, структур данных (по шкале от 0 до 5 баллов) | Качество реализации (по шкале от 0 до 5 баллов) | Степень владения материалом (по шкале от 0 до 5 баллов) | Презентация (шт. за одно занятие) |
|-------|------------------|---|---|---|---|-----------------------------------|
| 1. | | | | | | |
| 2. | | | | | | |
| 3. | | | | | | |

Критерии оценки мини-проектов 2-ый год обучения

№ группы: _____

Дата: _

| № п/п | ФИО обучающегося | Соответствие теме (по шкале от 0 до 5 баллов) | Сложность функционала (по шкале от 0 до 5 баллов) | Качество кода (по шкале от 0 до 5 баллов) | Удобство использования (шт. за одно занятие) | Презентация (шт. за одно занятие) |
|-------|------------------|---|---|---|--|-----------------------------------|
| 1. | | | | | | |
| 2. | | | | | | |
| 3. | | | | | | |

Лист оценки работы обучающихся в процессе разработки

№ группы: _____

Дата: _

| № п/п | ФИ обучающегося | Сложность языка разработки (по шкале от 0 до 5 баллов) | Соответствие написания программного кода поставленной задаче (по шкале от 0 до 5 баллов) | Презентация модели по плану. Степень владения специальными терминами (по шкале от 0 до 5 баллов) | Степень увлеченности процессом и стремления к оригинальности при выполнении заданий (по шкале от 0 до 5 баллов) | Кол-во вопросов и затруднений (шт. за одно занятие) |
|-------|-----------------|--|--|--|---|---|
| 1. | | | | | | |
| 2. | | | | | | |
| 3. | | | | | | |

Пример контрольного тестирования

| № | Вопросы | Верный ответ |
|----|--|--------------------|
| 1. | <p>Что выведет следующий фрагмент кода?</p> <pre>x = 4.5 y = 2 print(x // y)</pre> <p>Варианты ответов:</p> <ol style="list-style-type: none"> 1) 2.0 2) 2.25 3) 9.0 4) 20.25 5) 21 | 2.0 |
| 2. | <p>Что выведет следующий код, при его исполнении?</p> <p>Используется Python 3.x.</p> <pre>print(type(1 / 2))</pre> <p>Варианты ответов</p> <pre>class 'int' class 'number' class 'float' class 'double' class 'tuple'</pre> | class 'float' |
| 3. | <p>Что будет напечатано?</p> <pre>kvps = {"user", "bill", "password", "hillary"} print(kvps['password'])</pre> <p>Варианты ответов:</p> <ol style="list-style-type: none"> 1) user 2) bill 3) password 4) hillary 5) Ничего. TypeError. | Ничего. TypeError. |
| 4. | <p>Что будет напечатано?</p> <pre>name = "snow storm" print("%s" % name[6:8])</pre> <p>Варианты ответов:</p> <ol style="list-style-type: none"> 1) st 2) sto 3) to 4) Syntax Error | to |
| 5. | <p>Что напечатает следующий код:</p> <pre>word = 'foobar' print(word[3:] + word[:3])</pre> <p>Варианты ответов:</p> <ol style="list-style-type: none"> 1) foobar 2) obarof 3) barfoo 4) SyntaxError | barfoo |

| | | |
|----|--|--------------|
| 6 | <p>Что выведет следующая программа?</p> <pre>x = True y = False False = False if not x or y:print(1) elif not x or not y and z: print(2) elif not x or y or not y and x:print(3) else: print(4)</pre> <p>Варианты ответов:</p> <ol style="list-style-type: none"> 1) 1 2) 2 3) 3 5) 4) 4 | 3 |
| 7. | <p>Что выведет следующая программа?</p> <pre>a = [1,2,3,None,(),[],] print(len(a))</pre> <p>Варианты ответов</p> <ol style="list-style-type: none"> 1) 4 2) 5 3) 6 4) 7 | 6 |
| 8. | <p>Имеем следующую последовательность действий, чему равна переменная L2 ?:</p> <pre>>>> L1 = [2, 3, 4] >>> L2 = L1 >>> L1[0] = 24 >>> L1 [24, 3, 4] >>> L2</pre> <p>Варианты ответов:</p> <ol style="list-style-type: none"> 1) [2,3,4] 2) [24,3,4] 3) [2,3,24] 4) [3,4,2] | [24,3,4] |
| 9. | <p>Что покажет этот код?</p> <pre>for i in range(5): if i % 2 == 0: continue print(i)</pre> <p>Варианты ответов:</p> <ol style="list-style-type: none"> 1) Ошибку, так как i не присвоена 2) Ошибку из-за неверного вывода 3) Числа: 1, 3 и 5 4) Числа: 0, 2 и 4 5) Числа: 1 и 3 | Числа: 1 и 3 |

| | | |
|-----|--|-----------|
| 10. | <p>Что покажет этот код?</p> <pre>for j in 'Hi! I\'m mister Robert': if j == "\": print ("Найдено") break else: print ("Готово")</pre> <p>Варианты ответов:</p> <ol style="list-style-type: none">1) Ошибку в коде2) "Найдено" и "Готово"3) "Готово"4) "Найдено" | "Найдено" |
|-----|--|-----------|

Бланк наблюдения за обучающимися

Группа _____

Педагог _____

| № п/п | ФИО | ПОКАЗАТЕЛИ | | | | | РЕЗУЛЬТАТ |
|----------|-----|------------------------------|------------------------------------|---|--------------------------------|---|-----------|
| | | Внимателен в течение занятия | Использует базовую систему понятий | Проявляет инициативу, интерес в течение занятия | Идет на деловое сотрудничество | Аккуратно относится к материально-техническим ценностям | |
| 1 | | | | | | | |
| 2 | | | | | | | |
| 3 | | | | | | | |
| 4 | | | | | | | |
| 5 | | | | | | | |
| 6 | | | | | | | |
| 7 | | | | | | | |
| 8 | | | | | | | |
| 9 | | | | | | | |
| 10 | | | | | | | |

За каждое согласие с утверждением 1 – балл

2.5. Список литературы

Для педагога

1. Бэрри П. Изучаем программирование на Python. — М., 2017. — 624 с.
2. Буйначев С. К. Основы программирования на языке Python: учебное пособие. — Екатеринбург: Изд-во Урал. ун-та, 2014. — 91 с.
3. Бхаргава А. Грокаем алгоритмы: иллюстрированное пособие для программистов и любопытствующих. — СПб.: Питер, 2017. — 288 с.
4. Гэддис Т. Начинаем программировать на Python / пер. с англ. 4-е изд. — СПб.: БХВ-Петербург, 2019. — 768 с.
5. Доусен М. Програмируем на Python / М. Доусен - СПб.: Питер, 2016. -416с.
6. Мюллер Дж. Python для чайников. — СПб. : Диалектика, 2019. — 416с.
7. Луридаш П. Алгоритмы для начинающих: теория и практика для разработчика. — М. : Эксмо, 2018. — 608 с.
8. Лутц М. Изучаем Python, пер. с англ. 3-е изд. — СПб.: Символ Плюс, 2009. — 848 с.
9. Любанович Б. Простой Python. Современный стиль программирования/ Б. Любанович. - СПб.: Питер, 2016. - 480с.
10. Прохоренок Н.А., Дронов В.А. Python 3 и PyQt 5. Разработка приложений / Н.А. Прохоренок, В.А. Дронов - СПб.: «БХВ-Петербург», 2016. - 832с.
11. Рафгарден Т. Совершенный алгоритм. Жадные алгоритмы и динамическое программирование. — СПб.: Питер, 2020. — 256 с.
12. Рейтц К., Шлюссер Т. Автостопом по Python. — СПб. : Питер, 2017. — 336 с.
13. Саммерфильд М. Python на практике / М. Саммерфильд, пер. А.А.Слинкин – М.: ДМК-Пресс, 2014. - 338с.
14. Фёдоров Д. Ю. Программирование на языке высокого уровня Python: учебное пособие для прикладного бакалавриата. — М. : Издательство Юрайт, 2019. — 161 с.

Для обучающихся

1. Задачи по программированию. Под ред. С. М. Окулова, М.: БИНОМ. Лаборатория знаний, 2006.
2. Информатика и ИКТ. Задачник-практикум в 2 частях. Под ред. И. Г. Семакина и Е. К. Хеннера. М.: БИНОМ. Лаборатория знаний, 2014.
3. Лутц М. Изучаем Python. СПб.: Символ-Плюс, 2011.
4. Окулов С. М. Основы программирования. М.: Бином. Лаборатория знаний, 2012.
5. Поляков К. Ю., Еремин Е. А. Информатика. Углублённый уровень. Учебник для 10 класса в 2 частях. М.: БИНОМ. Лаборатория знаний, 2014.
6. Эльконин, Д. Б. Детская психология: учеб. пособие для студ. высш. учеб. заведений / Д. Б. Эльконин; ред. сост. Б. Д. Эльконин. – 4-е изд., стер. – М.: Издательский центр «Академия», 2007. – 384 с

Электронные ресурсы:

2. Материалы и презентации к урокам в LMS Яндекс.Лицея.
3. Сайт «Python 3 для начинающих» – pythonworld.ru.
4. Сайт «Питонтьютор» – pythontutor.ru.
5. Лекции А. В. Умнова, прочитанные в Школе анализа данных Яндекса – <https://www.youtube.com/playlist?list=PLJOzdkh8T5kpIBTG9mM2wVBjh5OpdwBl>.