

УТВЕРЖДАЮ
Ректор
ФГБОУ ВО «РГУПС»



В.Н. Верескун
М.П.

УТВЕРЖДАЮ
Президент
БФ «Система»



О.П. Косаченко
М.П.

УТВЕРЖДАЮ
Генеральный директор
ООО «Алгоритмика»



А.С. Лобанов
М.П.

ДОПОЛНИТЕЛЬНАЯ ОБРАЗОВАТЕЛЬНАЯ ПРОГРАММА ТЕХНИЧЕСКОЙ НАПРАВЛЕННОСТИ

Возраст обучающихся: 14–17 лет
Уровень: модульная образовательная программа
«Курс программирования Python. Start (второй год обучения)»

Авторы-составители:
Нагпал Арвинд
Голубцова Полина

Москва, Ростов
2021 г.

СОДЕРЖАНИЕ

ПОЯСНИТЕЛЬНАЯ ЗАПИСКА	3
1. НОРМАТИВНО-ПРАВОВАЯ БАЗА	5
2. ЦЕЛЬ И ЗАДАЧИ ПРОГРАММЫ	6
3. ПЛАНИРУЕМЫЕ РЕЗУЛЬТАТЫ	7
4. УЧЕБНЫЙ ТЕМАТИЧЕСКИЙ ПЛАН	14
5. СОДЕРЖАНИЕ УЧЕБНО-ТЕМАТИЧЕСКОГО ПЛАНА.....	18
6. ОРГАНИЗАЦИОННО-ПЕДАГОГИЧЕСКИЕ УСЛОВИЯ РЕАЛИЗАЦИИ ПРОГРАММЫ	22

Приложение 1. Список литературы

ПОЯСНИТЕЛЬНАЯ ЗАПИСКА

Данная общеобразовательная общеразвивающая программа дополнительного образования детей имеет техническую направленность и реализуется в рамках благотворительной программы «Лифт в будущее. Школа».

Программа предполагает дополнительное образование детей в области программирования на языке Python.

Организация образовательной деятельности в рамках программы основана на модульном принципе представления содержания образовательной программы, построения учебных планов и использовании современных образовательных технологий, в сетевом взаимодействии, с использованием ресурсов нескольких организаций.

В рамках программы обучающиеся освоят 4 модуля:

- 1 модуль — «Структуры данных»;
- 2 модуль — «Разработка оконных приложений»;
- 3 модуль — «Работа с файлами»;
- 4 модуль — «Автоматическая обработка изображений».

Содержание курса программирования Python Start для обучающихся 14–17 лет рассчитано на обучение в течение 5 месяцев (20 занятий — 40 академических часов). Занятия проходят 1 раз в неделю и длятся 2 академических часа. Предусмотрены необязательные домашние задания. Отбор на курс предусмотрен только по возрасту.

В отличие от других курсов программирования Python Start учит ребенка думать, как программист, а не просто писать код. Более того, проекты, выполняемые на уроках, охватывают различные области интересов, в том числе школьные предметы. Также ученики развивают умение презентовать себя и свои программы как в группе, так и перед учителями и родителями. На курсе возможны следующие **формы деятельности** учащихся: учебная дискуссия, занятие-проект, практическое занятие, конференция. При решении

одной задачи ученики могут работать сначала в малых группах, затем индивидуально, а при проверке задачи — в паре с учителем.

Особенность курса заключается в том, что детям не дают образец решения задачи.

Ученики сами пытаются составить последовательность действий, ведущую к цели, и запрограммировать ее. Полученные знания ребенок применяет при создании собственных проектов, которые защищаются перед другими учениками, педагогами и родителями.

Теоретической основой данной программы являются:

Компетентностный подход: обучение в системе, развивающей предметные способности. Развитие каждой способности происходит в три шага: формирование учебной мотивации; формирование предметных знаний, умений и навыков; формирование личностных изменений, связанных с усвоением новых знаний (Хуторской А. В. и др.)

Личностный подход: обучение в условиях развития каждого учащегося как личности (Вербцкий А. А. и др.).

1. НОРМАТИВНО-ПРАВОВАЯ БАЗА

Программа разработана в соответствии с нормативно-правовыми документами:

1. Федеральный закон от 29 декабря 2012 г. № 273-ФЗ «Об образовании в Российской Федерации».

2. Приказ Минпросвещения России от 09.11.2018 № 196 «Об утверждении Порядка организации и осуществления образовательной деятельности по дополнительным общеобразовательным программам».

3. Постановление Главного государственного санитарного врача Российской Федерации от 04.07.2014 № 41 Об утверждении СанПиН 2.4.4.3172-14 «Санитарно-эпидемиологические требования к устройству, содержанию и организации режима работы образовательных организаций дополнительного образования детей».

2. ЦЕЛЬ И ЗАДАЧИ ПРОГРАММЫ

Цель всего курса — научить ребенка использовать язык программирования Python как инструмент для реализации идей в области своих интересов.

Для этого сначала ученики получают базовые знания, затем учатся проектной деятельности, после чего реализуют проекты в области своих увлечений. При этом базовая часть также подразумевает возможность выразить свои интересы. Проекты выполняются преимущественно индивидуально, чтобы дети сначала научились работать над ними, затем уже — работать в группе.

Достижение поставленной цели складывается из выполнения следующих задач:

1. Развитие профессиональных компетенций.
2. Развитие коммуникативных способностей и умений работать в команде.
3. Развитие навыков планирования и презентации проектов.
4. Развитие креативных способностей и умений учиться.
5. Формирование мотивации успеха, готовности к действиям в новых условиях и нестандартных ситуациях.

Данные задачи не только способствуют формированию предметных компетенций, но еще и формируют мировоззрение школьника, раскрывают значение Soft skills, учат грамотному использованию компьютера и готовят к жизни в цифровом обществе.

3. ПЛАНИРУЕМЫЕ РЕЗУЛЬТАТЫ

Достижение цели и задач образовательной программы предполагает получение следующих результатов:

Ожидаемый результат	Способ достижения	Критерий достижения образовательного результата
Личностные результаты обучения		
Освоение социальной роли обучающегося и формирование личностного смысла учения.	Демонстрация связи между способностью выполнить интересную задачу и наличием/отсутствием соответствующих знаний.	За отведенное время ученик пытается не только выполнить базовые уровни, но и приступить к бонусным.
Развитие навыков сотрудничества со взрослыми и сверстниками и умений находить выходы из спорных ситуаций.	Использование парного программирования как способа показать преимущества работы в команде.	Ученик не боится просить помощи и сам пытается помогать одноклассникам и учителю.
Овладение начальными навыками адаптации в динамично изменяющемся и развивающемся мире.	Обсуждение научно-популярных материалов, демонстрирующих необходимость не только овладения предметными	При работе над проектом ученик сначала пытается найти информацию в Интернете или теоретической справке на платформе и только

	<p>навыками, но и умениями учиться.</p>	<p>в случае неудачи задает вопрос учителю.</p>
<p>Установка «Мир удивительный».</p>	<p>Через разнообразие проектов, затрагивающих самые разные области, ребенок видит, что даже в самой неинтересной для него области есть много занимательного.</p>	<p>Интересуется как разными аспектами своей области интересов, так и другими направлениями.</p>
<p>Установка «Компьютер — удивительный инструмент, с помощью которого можно добиться многого».</p>	<p>Акцентируем внимание детей на том, что с помощью компьютера они делают вещи, которые без него делали бы месяцами; добавляем научно-популярные вставки про использование компьютеров.</p>	<p>В тестовом задании ставит компьютер на самую большую отметку.</p>
<p>Установка «Любой рутинный процесс можно автоматизировать».</p>	<p>Прежде чем показать легкий способ выполнения задачи, предлагаем начать делать ее вручную, затем показываем, что это рутинный процесс,</p>	<p>В сценарном задании предлагает автоматизировать рутинный процесс, при выполнении проекта использует алгоритмы для автоматизации рутинных операций.</p>

	и его можно автоматизировать.	
Метапредметные навыки		
Развитие и формирование учебных действий.	Создание благоприятных условий для участия в диалоге, в коллективном обсуждении. Строится продуктивное взаимопонимание со сверстниками и взрослыми в процессе коллективной деятельности.	Ученик легко общается, не боится просить помощь или оказать ее другим. Ученик способен кооперироваться, чтобы достичь цели.
Умение осознанно выбирать наиболее эффективные способы решения учебных и познавательных задач.	Задания, предполагающие необходимость апробации нескольких методов, один из которых заработает (единственно верный вариант не дается заранее).	Ученик пробует свои идеи, не ожидая разрешения учителя. Задачи, решение которых подразумевает теорию, которой еще не было, решают с n-ой попытки.
Умение корректировать свои действия в соответствии с изменяющейся ситуацией.	Создание ситуации успеха при решении незнакомых задач (даем задачу и через подсказки помогаем прийти до решения).	Сталкиваясь с новой задачей, ученик ищет решение: задает вопросы, проясняющие структуру задачи; находит знакомые

		подзадачи и аналогии; разделяет задачу на составные части.
Умение осознанно использовать речевые средства для обратной связи.	Презентация учениками своих проектов, в ходе которой учитель и другие ученики дают обратную связь друг другу.	Ученик в ходе комментирования работы одноклассника избегает оценочных суждений, выделяет как сильные, так и слабые стороны.
Умение ставить и формулировать для себя новые задачи в учебе и познавательной деятельности.	В начале выполнения проекта ученики зарисовывают его план, учитель показывает, как это лучше делать.	Ученик раскладывает проект на составные элементы, затем выстраивает их в последовательность шагов.
Умеет презентовать свою работу.	Ученики презентуют свои проекты, учитель и другие ученики дают обратную связь. Учитель также дает советы, каким образом это лучше делать.	Во время презентации своих проектов пользуется вниманием аудитории.
Развитие критического и алгоритмического мышления учеников.	Выполнение логических операций: сравнение, анализ, синтез, обобщение, установление аналогий на уроке. Анализ любой	Ученик строит логическую цепь рассуждений. Управляет своей деятельностью.

	получаемой информации, составление плана действий для решения задачи.	
Предметные навыки		
Формирование базовых навыков работы с основными инструментах разработчика на Python.	Изучение теоретического материала совместно с учителем, выполнение заданий на платформе с постепенным увеличением сложности.	Ученик оперирует базовыми профессиональными инструментами разработчика при работе над проектами.
Знакомство с расширением PyQt.	Изучение теоретического материала совместно с учителем, выполнение заданий в профессиональной среде с расширением образовательной платформы «Алгоритмики» с постепенным увеличением сложности.	Ученик самостоятельно пишет программы, используя расширение PyQt.

<p>Развитие умений применять Python для решения задач практического характера и задач из смежных дисциплин.</p>	<p>Выполнение заданий в профессиональной среде с расширением образовательной платформы «Алгоритмики» с постепенным увеличением сложности.</p>	<p>Ученик самостоятельно пишет программы на языке Python, решающие задачи практического характера.</p>
<p>Следование стандарту при оформлении кода.</p>	<p>Изучение теоретического материала совместно с учителем, выполнение заданий в профессиональной среде с расширением образовательной платформы «Алгоритмики».</p>	<p>Ученик оформляет код в соответствии с общепринятыми стандартами и использует комментарии.</p>
<p>Умение тестировать программу и находить в ней ошибки.</p>	<p>Выполнение проектных заданий с постепенным увеличением сложности.</p>	<p>Ученик самостоятельно определяет и исправляет ошибки в своем и чужом коде.</p>
<p>Использование итерационного подхода.</p>	<p>Выполнение заданий в профессиональной среде с расширением образовательной</p>	<p>Ученик пишет программы, разделяя код на отдельные части.</p>

	платформы «Алгоритмики».	
--	-----------------------------	--

Механизмы отслеживания результатов:

1. Задания на платформе с автопроверкой (все задания на тему базовых понятий проверяются автоматически системой).
2. Ревью кода (выборочная оценка кода учеников на соблюдение стандартов его написания).
3. Конференции и «круглые столы», на которых возможна взаимопроверка.
4. Чек-листы для преподавателей, других учеников (большинство метанавыков оценивается через заполнение преподавателями и учениками анкет).
5. Анкета родителей (для оценки появления и развития установок).

Учебно-дидактическое обеспечение:

1. Поурочные методические указания.
2. Тематические презентации.
3. Персональные компьютеры по числу учащихся в группе.
4. Задания на платформе <https://lk.bf.sistema.ru/> в личных ученических аккаунтах.
5. Компьютер, доска, проектор для учителя.

4. УЧЕБНЫЙ ТЕМАТИЧЕСКИЙ ПЛАН

В рамках личностного подхода, реализуемого в Python Start, в учебном тематическом плане (УТП) указана примерная продолжительность каждого модуля, поскольку преподаватель может ускорять или замедлять темп урока для каждой группы.

Например, если преподаватель видит, что группа не усвоила тему за отведенное время, то он имеет право повторить урок.

№	Наименование раздела	Количество академических часов			Форма аттестации
		всего	теория	практика	
<i>Модуль 1. Структуры данных</i>					
1	Повторение. Обработка исключений	2	1	1	Задания на платформе с автопроверкой
2	Повторение. Списки	2	1	1	Задания на платформе с автопроверкой
3	Словари	2	1	1	Задания на платформе с автопроверкой
4	Сложенные структуры данных	2	1	1	Задания на платформе с автопроверкой, сюжетного задания «Хакатон»
<i>Модуль 2. Разработка оконных приложений</i>					
5	Классы. Введение в PyQt	2	1	1	Задания на платформе с автопроверкой,

					создание программ в Visual Studio Code и их защита перед учителем
6	Проектирование интерфейса	2	1	1	Задания на платформе с автопроверкой, создание программ в Visual Studio Code и их защита перед учителем
7	Приложение Memory Card	8	4	4	Задания на платформе с автопроверкой, создание программ в Visual Studio Code и их защита перед учителем
<i>Модель 3. Работа с файлами</i>					
8	Основы работы с файлами	2	1	1	Задания на платформе с автопроверкой, создание программ в Visual Studio Code и их защита перед учителем
9	Приложение «Умные заметки»	6	3	3	Задания на платформе с автопроверкой,

					создание программ в Visual Studio Code и их защита перед учителем
<i>Модуль 4. Автоматическая обработка изображений</i>					
10	Основы обработки изображений	2	1	1	Задания на платформе с автопроверкой, создание программ в Visual Studio Code и их защита перед учителем
11	Приложение Easy Editor	6	3	3	Задания на платформе с автопроверкой, создание программ в Visual Studio Code и их защита перед учителем
12	Выполнение итогового проекта	4	2	2	Создание программ в Visual Studio Code в соответствии с техническим заданием, самостоятельное тестирование программы, ревью программы одногруппником,

					ревью программы учителем, защита проекта
--	--	--	--	--	--

5. СОДЕРЖАНИЕ УЧЕБНО-ТЕМАТИЧЕСКОГО ПЛАНА

Тема 1.1. Повторение. Обработка исключений

Теория

Функции ввода и вывода, переменные и типы данных. Алгоритмические конструкции — следование, ветвление и цикл. Модули Random и Time. Обработка исключений, операторы try, except.

Практика

Выполнение тренировочных задач на повторение. Создание программы с использованием оператора обработки исключений для ситуаций: ввода данных с клавиатуры и деления на 0.

Тема 1.2. Повторение. Списки

Теория

Списки. Методы работы со списком: добавление, удаление, вставка, сортировка, подсчет вхождений. Типы данных, которые может содержать список. Перебор элементов списка с помощью цикла for. Определение понятия «строка». Аналогии между методами работы со списками и со строками.

Практика

Выполнение тренировочных задач на повторение. Написание программы с использованием методов работы со списками.

Тема 1.3. Словари

Теория

Определение понятия «словарь». Добавление в словарь пары «ключ-значение». Методы работы с ключами и значениями словаря. Поиск ключа в словаре. Типы данных ключей и значений.

Практика

Выполнение тестовых заданий по теме урока. Отработка ввода команд с клавиатуры, исправление ошибок в написанных программах и написание собственных программ на платформе.

Тема 1.4. Вложенные структуры данных

Теория

Правила создания лаконичного кода. Вложенные структуры. Применение методов к вложенным структурам. Обращение к объектам внутри вложенных структур с помощью индексов. Решение задач повышенной сложности.

Практика

Выполнение тестовых заданий по теме урока. Создание программы с использованием вложенной структуры для решения задач повышенной сложности.

Тема 2.1. Классы. Введение в PyQt

Теория

Определение «класс». Связь классов и объектов. Конструкторы, поля и методы в классах. Приложения и виджеты. Расположение виджетов. Обработка событий средствами PyQt.

Практика

Выполнение тестовых заданий по теме урока. Создание первого приложения на PyQt.

Тема 2.2. Проектирование интерфейсов

Теория

Наследование. Суперклассы, классы-наследники, их методы. Виджеты QRadioButton, QMessageBox и др.

Практика

Создание приложение на PyQt в среде Visual Studio Code.

Тема 2.3. Приложение Memory Card

Теория

Основные элементы приложения: окно, текст, кнопки, флажки, переключатели, списки, меню. Проектирование интерфейса приложения с помощью лейаутов и виджетов.

Практика

Создание приложение Memory Card на PyQt в среде Visual Studio Code.

Тема 3.1. Основы работы с файлами

Теория

Атрибуты доступа к файлам. Оператор with. Чтение информации из файла. Строки и классы для считывания информации из файла.

Практика

Создание программы для выполнения задания на чтение и запись данных в файл.

Тема 3.2. Приложение «Умные заметки»

Теория

Json-файлы. Запись и чтение данных из json-файлов. Глобальные и локальные переменные. Программирование корректного отображения данных из файла.

Практика

Создание приложения для работы с заметками с возможностью создавать, удалять и редактировать заметки, задавать заметкам теги и выполнять поиск по ним.

Тема 4.1. Основы обработки изображений

Теория

Библиотека Python PIL как инструмент обработки изображений. Модули Image и ImageFilter библиотеки PIL. Параметры изображения Image, их изменение с помощью PIL.

Практика

Создание приложения в среде Visual Studio Code.

Тема 4.2. Приложение Easy Editor

Теория

Возможности и методы обработки изображений библиотекой PIL. Локальные и глобальные переменные. Программирование функций работы с файлами и интерфейсом. Массовая обработка изображений.

Практика

Создание приложения-фоторедактора Easy Editor (загрузка фото с компьютера, наложение ч/б фильтра, наведение резкости и поворот изображения).

6. ОРГАНИЗАЦИОННО-ПЕДАГОГИЧЕСКИЕ УСЛОВИЯ РЕАЛИЗАЦИИ ПРОГРАММЫ

Для успешной реализации курса программирования Python Start необходимо соблюдение следующих условий:

Материально-технические условия реализации программы	<ul style="list-style-type: none">- Учебный кабинет с местами для 12 учеников.- Компьютеры для учеников (1 ученик — 1 компьютер) и компьютер для учителя.- Проектор, подключенный к компьютеру для учителя, выводящий изображение на экран.- Доступ в Интернет на каждом компьютере.- Распечатанный список логинов и паролей учеников для доступа на платформу.- Среда Visual Studio Code с интеграцией платформы, установленная на каждый компьютер.- Доска или флипчарт.- Листы бумаги А4.- Ручки или карандаши (1 ученик — 1 ручка).
Учебно-методическое и информационное обеспечение программы	<ul style="list-style-type: none">- Поурочные методические рекомендации к занятиям.- Презентации к занятиям.- Задания на платформе с автопроверкой.- Задания на платформе со свободным решением.- Бонусные задания на платформе.- Задачи со свободным решением для реализации в среде программирования Visual Studio Code.

Список литературы

1. Хуторской А.В. Компетентностный подход в обучении. Научно-методическое пособие. — М.: Издательство «Эйдос»; Издательство Института образования человека, 2013. — 73 с.

2. Вербицкий, Ларионова: Личностный и компетентностный подходы в образовании: проблемы интеграции. Монография. — М.: Издательство «Логос», 2017. — 336 с.

3. К. Вольдерман, К. Стили, К. Квигли, М. Гудфеллоу, Д. Маккафферти, Дж. Вудкок. Программирование на Python: иллюстрированное руководство для детей. — М.: Манн, Иванов и Фербер, 2018. — 224 с.: илл.