

**Государственное профессиональное образовательное учреждение
Тульской области
«Чернский профессионально-педагогический колледж»**

Программа рассмотрена и одобрена
на заседании научно-методического совета
Протокол от 10.03.2020 г. № 7



**Дополнительная общеразвивающая программа
«Поколение Python: базовый курс»
рекомендована обучающимся 10 -17 лет**

Направленность: **техническая**
Уровень образования: **базовый**
Срок реализации: **1 год**

Автор-разработчик:
педагог дополнительного образования
Краева Татьяна Сергеевна

Чернь, 2020

Пояснительная записка

Дополнительная общеобразовательная (общеразвивающая) программа «Поколение Python: базовый курс» имеет техническую направленность.

Дополнительность программы заключается в том, что она отвечает интересам детей, способствует углублению знаний, умений и навыков, приобретаемых в общеобразовательной школе на уроках информатики.

Нормативные основания для создания дополнительной общеразвивающей программы

Данная программа разработана в соответствии с требованиями Федерального закона от 29 декабря 2012 года № 273-ФЗ «Об образовании в Российской Федерации», Федерального закона от 24.06.1999 года № 120-ФЗ «Об основах профилактики безнадзорности и правонарушений несовершеннолетних» (в ред. от 04.06.2014 г.), приказа Министерства просвещения Российской Федерации от 09.11.2018 года № 196 «Об утверждении Порядка организации и осуществления образовательной деятельности по дополнительным общеобразовательным программам», письма Министерства образования и науки Российской Федерации от 18 ноября 2015 года № 09-3242 «Методические рекомендации по проектированию дополнительных общеразвивающих программ (включая разноуровневые программы)», закона Тульской области от 30 сентября 2013 года № 1989-ЗТО «Об образовании», соответствует нормам санитарно-эпидемиологических требований к устройству, содержанию и организации режима работы образовательных организаций дополнительного образования детей СанПин 2.4.4. 3172-14.

Актуальность программы

Большинство школьных программ по информатике в основном охватывают пользовательский уровень обучения работы на компьютере. Школьники учат работать с текстовыми документами, элементарной графикой, создавать презентации и так далее. Темам «алгоритмы» и «программирование» отводится небольшое количество часов, изучаются они поздно, в старших классах. Отсутствуют такие необходимые для развития логического мышления предметы, как алгоритмика, логика. Это замедляет формирование алгоритмического мышления и не способствует развитию интереса учащихся к программированию. Парадокс ещё заключается в том, что наибольший балл в заданиях Единого государственного экзамена и во всех значимых олимпиадах, приносят как раз задания на программирование.

Обучение по данной программе создает благоприятные условия для

более раннего «погружения» учащихся в мир логики и позволяет сформировать у детей стойкий интерес к получению и усовершенствованию знаний в области программирования и IT - технологий, выявить способных к программированию детей и оказать помощь в профессиональном самоопределении.

Направленность программы: техническая.

Новизна программы

Новизна Программы заключается в том, что Python дает более широкие возможности в области программирования, чем Pascal, который входит в школьный курс информатики. На языке Python можно легко и быстро создавать простые компьютерные игры, трёхмерные модели и программировать роботов. Этот язык быстрее и легче усваивается, чем Pascal. Многие мировые компании такие, как Intel, Cisco, Hewlett-Packard, используют этот язык при реализации своих проектов. Крупнейшие интернет-ресурсы такие, как Google, YouTube, также разработаны с помощью языка программирования Python.

Педагогическая целесообразность

Педагогическая целесообразность программы заключается в привлечении учащихся к занятиям техническим творчеством, что способствует развитию логического мышления, творческих способностей и навыков решения задач программирования. Программирование мотивирует к занятиям в различных научных областях (физики, информатики, алгебры, геометрии и др.), развивает воображение и способствует ранней профориентации подростков. Для достижения поставленных задач занятия проводятся в формате «от простого к сложному». Учащиеся вспоминают свои знания по основам алгоритмизации и программирования и на их основе, углубляя их, учатся составлять простые и сложные программы.

Отличительная особенность

Основное количество часов отводится практическому написанию программ. Каждый обучающийся реализует индивидуальный проект в результате освоения программы. Продукт, полученный в результате освоения программы, имеет прикладной характер и может быть использован по необходимости.

Цель - способствовать формированию творческой личности, обладающей информационными компетенциями, владеющей базовыми понятиями теории алгоритмов, умеющей разрабатывать эффективные алгоритмы и реализовывать их в виде программы, написанной на языке

программирования Python.

Задачи Программы

Обучающие:

- обучить языку программирования Python;
- обучить навыкам разработки, тестирования и отладки несложных программ;

• обучить синтаксису языка и основным базовым алгоритмическим конструкциям

- научить создавать прикладное программное обеспечение;
- расширять кругозор обучающихся в области программирования.

Развивающие:

• развивать память и внимание, познавательную и творческую активность;

• развивать творческие способности, эстетическое и эргономическое восприятие объектов труда;

• развивать логическое мышление.

Воспитательные:

• прививать интерес к активному творческому самовыражению, культуре труда;

- воспитывать упорство в достижении желаемого результата;
- воспитывать эстетический вкус;
- воспитывать чувство взаимопомощи, доверия, коллективизма.

Принципы реализации программы:

- принцип добровольности;
- принцип учёта возрастных особенностей детей;
- принцип систематичности и последовательности в освоении знаний и умений;
- принцип опоры на интерес;
- принцип доступности (весь предлагаемый материал должен быть доступен пониманию ребёнка);
- принцип обратной связи;
- принцип ориентации на успех;
- принцип взаимоуважения;
- принцип индивидуально-личностной ориентации;
- принцип связи обучения с жизнью;
- принцип креативности (творчества) и коллективности;
- принцип научности.

Формы реализации программы – очная форма с элементами дистанционного обучения.

Уровень образования: базовый.

Сроки реализации: 1 год.

Занятия проводятся 1 раз в неделю по 2 учебных часа. (1 час = 40 минут, обязательный перерыв – 10 минут). Всего 72 часа в год.

Адресат программы: дополнительная общеразвивающая программа «Поколение Python: базовый курс» предназначена для обучающихся 10-17 лет образовательных организаций всех типов. Набор детей в объединении осуществляется в начале учебного года. В связи с тем, что занятия требуют индивидуального подхода, группы комплектуется из расчёта 10 человек для обучения.

Текущий контроль успеваемости и формы оценки результативности обучающихся:

Основной формой подведения итогов дополнительной общеразвивающей программы «Язык программирования Python» является решения задач (разработка и отладка программ на языке программирования Python), реализация индивидуального проекта прикладного характера.

Прогнозируемый результат освоения дополнительной общеобразовательной общеразвивающей программы «Поколение Python: базовый курс»

В процессе освоения программы у обучающихся формируются:

Личностные компетенции:

- формирование ответственного отношения к обучению, готовности и способности, обучающихся к саморазвитию и самообразованию на основе мотивации к обучению и познанию;
- формирование коммуникативной компетентности в общении и сотрудничестве со сверстниками, взрослыми в процессе образовательной, общественно-полезной, учебно-исследовательской, творческой и других видах деятельности.

Метапредметные компетенции:

- умение самостоятельно планировать пути достижения цели, в том числе альтернативные, осознанно выбирать наиболее эффективные способы решения задач;
- умение соотносить свои действия с планируемыми результатами, осуществлять контроль всей деятельности в процессе достижения результата, корректировать свои действия в соответствии с изменяющейся ситуацией;
- умение оценивать правильность выполнения учебной задачи, собственные возможности ее решения;
- владение основами самоконтроля, самооценки, принятия решений и осуществления осознанного выбора в учебной и познавательной

деятельности;

- умение организовывать сотрудничество и совместную деятельность с учителем и сверстниками; работать индивидуально в группе: находить общее решение и разрешать конфликты на основе согласования позиций и учета интересов; формулировать, аргументировать и отстаивать свое мнение;

- формирование и развитие компетентности в области использования информационно-коммуникационных технологий.

**Учебно-тематический план
дополнительной общеобразовательной общеразвивающей
программы «Поколение Python: базовый курс»
(1 год обучения, 72 часа)**

№ п/п	Название разделов, тем	Количество часов			Форма оценки
		теори я	практик а	всег о	
1.	Ввод-вывод данных	2	5	7	
	Знакомство с Python	1	1	2	Итоговая работа на ввод- вывод данных
	Команды print и input		1	1	
	Параметры sep и end		1	1	
	Целочисленная арифметика.	1	2	3	
2.	Условный оператор	2	5	7	
	Выбор из двух		1	1	Итоговая работа на условный оператор
	Логические операции	1	2	3	
	Вложенные и каскадные условия	1	2	3	
3.	Типы данных	3	3	6	
	Числовые типы данных: int, float	1	1	2	
	Модуль math	1	1	2	
	Строковый тип данных	1	1	2	
4.	Циклы for и while	2	9	11	
	Цикл for		1	1	Итоговая работа на циклы
	Цикл for: функция range	1	1	2	
	Частые сценарии		1	1	
	Цикл while		1	1	
	Цикл while: обработка цифр числа	1	1	2	
	break, continue и else		1	1	
	Поиск ошибок и ревью кода		1	1	
	Вложенные циклы		2	2	
5.	Строковый тип данных	1	6	7	
	Индексация		1	1	Итоговая работа на строковый тип данных
	Срезы		1	1	
	Методы строк.	1	2	3	
	Строки в памяти компьютера, кодировка Unicode		2	2	
6.	Списки	3	7	10	
	Введение в списки	1		1	Итоговая работа на списки
	Основы работы со списками		1	1	
	Методы списков. Часть 1	1	1	2	
	Вывод элементов списка		1	1	

	Методы строк: split, join		1	1	
	Методы списков. Часть 2	1	1	2	
	Списочные выражения		1	1	
	Сортировка списков		1	1	
7.	Функции	4	6	10	
	Функции без параметров	1	1	2	Итоговая работа на функции
	Функции с параметрами	1	1	2	
	Локальные и глобальные переменные	1	2	3	
	Функции с возвратом значения.	1	2	3	
8.	Работа над мини-проектом	7	7	14	
	Модуль random	1	1	2	Итоговая работа индивидуальный проект
	Числовая угадайка	1	1	2	
	Магический шар 8	1	1	2	
	Генератор безопасных паролей	1	1	2	
	Шифр Цезаря	1	1	2	
	Калькулятор систем счисления	1	1	2	
	Угадайка слов	1	1	2	
	ИТОГО:	24	48	72	

Содержание дополнительной общеразвивающей программы «Поколение Python: базовый курс» (72 часа)

1. Ввод-вывод данных (теория 2 ч./практика 5 ч.)

Теория:

Язык программирования Python. Достоинства и недостатки. Области применения. Интерактивный режим. Официальный сайт языка Python. Дистрибутивы языка Python. Область видимости переменных. Присваивание значения переменных в коде программы. Ввод значения переменных пользователем с помощью функции input(). Функция print() для вывода данных на экран и в файл.

Практические занятия:

- 1.1 Знакомство с Python
- 1.2 Команды print и input
- 1.3 Параметры sep и end
- 1.4 Целочисленная арифметика.

2. Условный оператор (теория 2 ч./практика 5 ч.)

Теория:

Понятие алгоритма их типы. Описание задачи в виде алгоритма. Алгоритмы с ветвлением. Условный оператор. Синтаксис условного оператора. Вложенные условия. Множественные условия. Ленивая оценка условий. Сложные условия. Формулировка условий.

Практические занятия:

- 2.1 Выбор из двух
- 2.2 Логические операции
- 2.3 Вложенные и каскадные условия

3. Типы данных (теория 3 ч./практика 3 ч.)

Теория:

Числовые и строковые типы данных.

Практические занятия:

- 3.1 Числовые типы данных: int, float
- 3.2 Модуль math
- 3.3 Строковый тип данных

4. Циклы for и while (теория 2 ч./практика 9 ч.)

Теория:

Циклические алгоритмы. Циклы с предусловием и постусловием. Счетчик итераций. Цикл for, синтаксис. Функция range(). Команды break и continue. Вложенные циклы for. Цикл while, синтаксис. Счетчик итераций. Команды break и continue. Вложенные циклы. Бесконечный цикл. Формулировка условия. Составление оптимального алгоритма программы.

Практические занятия:

- 4.1 Цикл for
- 4.2 Цикл for: функция range
- 4.3 Частые сценарии
- 4.4 Цикл while
- 4.5 Цикл while: обработка цифр числа
- 4.6 break, continue и else
- 4.7 Поиск ошибок и ревью кода
- 4.8 Вложенные циклы

5. Строковый тип данных (теория 1 ч./практика 6 ч.)

Теория:

Строка как последовательность. Функция len. Срезы строк. Методы строк. Работа с текстовыми файлами.

Практические занятия:

- 5.1 Индексация
- 5.2 Срезы
- 5.3 Методы строк.
- 5.4 Строки в памяти компьютера, кодировка Unicode

6. Списки (теория 3 ч./практика 7 ч.)

Теория:

Списки. Перебор элементов списка. Операции со списками. Срезы списков. Отображение, фильтрация и сокращение. Удаление элементов. Создание синонимов.

Практические занятия:

- 6.1 Введение в списки
- 6.2 Основы работы со списками
- 6.3 Методы списков. Часть 1
- 6.4 Вывод элементов списка
- 6.5 Методы строк: split, join
- 6.6 Методы списков. Часть 2
- 6.7 Списочные выражения
- 6.8 Сортировка списков

7. Функции (теория 4 ч./практика 6 ч.)

Теория:

Подпрограмма, функция, процедура, объявление, определение и вызов функции, возврат значений, глобальные и локальные переменные, передача параметров, работа с памятью, стек вызовов.

Практические занятия:

- 7.1 Функции без параметров

- 7.2 Функции с параметрами
- 7.3 Локальные и глобальные переменные
- 7.4 Функции с возвратом значения.

8. Работа над мини-проектом (теория 7 ч./практика 7 ч.)

Теория:

Обсуждение и разработка мини проектов в портфолио учащихся.

Практические занятия:

- 8.1 Модуль random
- 8.2 Числовая угадайка
- 8.3 Магический шар 8
- 8.4 Генератор безопасных паролей
- 8.5 Шифр Цезаря
- 8.6 Калькулятор систем счисления
- 8.7 Угадайка слов

**Методическое обеспечение дополнительной общеобразовательной
общеразвивающей программы «Поколение Python: базовый курс»**

Тема	Форма проведения занятия	Дидактические средства обучения		
		Учебно-методическая литература, методические пособия	Наглядные пособия	ИКТ, ТСО
Ввод-вывод данных	Рассказ, беседа, обсуждение, практическая работа	Федоров Д. Ю. ОСНОВЫ программирования на примере языка Python. Васильев А.Н. Python на примерах. Практический курс по программированию Пейн, Брайсон. Python для детей и родителей	Платформа Stepik, презентация, раздаточный материал.	компьютерный класс, мультимедийный проектор, необходимое программное обеспечение (среда программирования языка Python, операционная система Windows, пакет Microsoft Office), компьютеры, принтер и копировальный аппарат, интернет.
Условный оператор	Рассказ, беседа, обсуждение, практическая работа			
Типы данных	Рассказ, беседа, обсуждение, практическая работа			
Циклы for и while	Рассказ, беседа, обсуждение, практическая работа			
Строковый тип данных	Рассказ, беседа, обсуждение, практическая работа			
Списки	Рассказ, беседа, обсуждение, практическая работа			
Функции	Рассказ, беседа, обсуждение, практическая работа			
Работа над мини-проектом	Исследовательские занятия			

**Способы и формы проверки результатов дополнительной
общеобразовательной общеразвивающей программы
«Поколение Python: базовый курс»**

Тема 1. «Ввод-вывод данных»: итоговая работа, решение задач.
Тема 2. «Условный оператор»: итоговая работа, решение задач.

- Тема 4. «Циклы for и while»:** итоговая работа, решение задач.
- Тема 5. «Строковый тип данных»:** итоговая работа, решение задач.
- Тема 6. «Списки»:** тематические тесты, решение экологических задач.
- Тема 7. «Функции»:** итоговая работа, решение задач.
- Тема 8. «Работа над мини-проектом»:** разработка мини-проектов для портфолио.

**Планируемые результаты освоения дополнительной
общеобразовательной общеразвивающей программы
«Поколение Python: базовый курс»**

Обучающиеся должны знать:

- основные типы алгоритмов;
- иметь представление о структуре программы, основы программирования на языках высокого уровня;
- базовые алгоритмические конструкции;
- содержание этапов разработки программы: алгоритмизация-кодирование-отладка- тестирование;
- дополнительные возможности языка Python для выражения различных алгоритмических ситуаций;
- алгоритмы и программы на языке Python решения простых, сложных и нестандартных задач в математической области;
- основы разработки простых игр в системе программирования Python.

Обучающиеся должны уметь:

- записывать алгоритмические структуры на языке программирования Python;
- использовать Python для решения задач из области математики, физики;
- использовать основные алгоритмические приемы при решении математических задач;
- решать простые, сложные и нестандартные задачи;
- создавать простые игры;
- анализировать текст чужих программ, находить в них неточности, оптимизировать алгоритм, создавать собственные варианты решения.

**Условия реализации дополнительной общеразвивающей
программы «Поколение Python: базовый курс»**

№ п/п	Наименование	Назначение/краткое описание функционала оборудования
1	Учебное (обязательное) оборудование	
1.1	Учебные пособия	Предназначены для изучения основ программирования
1.2	Таблицы, схемы	Предназначены для наглядного изучения блок схем
2	Компьютерное оборудование	
2.1	Ноутбук/ компьютер	Использование возможностей ИКТ- технологий, работа в кабинете
2.2	Интернет	
2.3	Мультимедийный проектор с экраном	

Список использованной литературы для педагога

1. Бизли Д. М. Язык программирования Python : справочник : пер. с англ. / Д. М. Бизли. – Киев : ДиаСофт, 2000
2. Гифт Н. Python в системном администрировании UNIX и Linux : пер. с англ. / Н. Гифт, Д. Джонс. – СПб. : Символ-Плюс, 2009
3. Лейнингем И. Освой самостоятельно Python за 24 часа : пер. с англ. / И. Лейнингем. – М. : Издательский дом «Вильямс», 2001
4. Лесса А. Python. Руководство разработчика : пер. с англ. / А. Лесса. – СПб. : ДиасофтЮП, 2001
5. Лутц М. Изучаем Python : пер. с англ. / М. Лутц. – СПб. : Символ-Плюс, 2009
6. Лутц М. Программирование на Python : пер. с англ. / М. Лутц. – СПб. : Символ-Плюс, 2002
7. Саммерфельд М. Программирование на Python 3 Подробное руководство : пер. с англ. / М. Саммерфельд. – СПб. : Символ-Плюс, 2009
8. Сузи Р. А. Python / Р. А. Сузи. – СПб. : БХВ-Петербург, 2002
9. Сузи Р. А. Язык Python и его применения : учеб. пособие / Р. А. Сузи. – М. : Интернет- Университет информационных технологий: БИНОМ. Лаборатория знаний, 2006
10. Язык программирования Python / Г. Россум [и др.]. – СПб. : АНО «Институт логики» – Невский диалект, 2001
11. Сэнд У., Сэнд К. Hello World! Занимательное программирование. — СПб.: Питер, 2016
12. Шоу, Зед. Легкий способ выучить Python / Зед Шоу ; [пер. с англ. М. А. Райтмана]. — М: Издательство «Э», 2017.
13. Мэттиз Эрик. Изучаем Python. Программирование игр, визуализация данных, веб-приложения. — СПб.: Питер, 2017.
14. Прохоренок, Н. А. Python 3 и PyQt 5. Разработка приложений / Н. А. Прохоренок, В. А. Дронов. – СПб.: БХВ-Петербург, 2016
15. Саммерфилд М. Программирование на Python 3. Подробное руководство. СПб.: Символ-Плюс — 2009
16. Златопольский Д. М. Основы программирования на языке Python. – М.: ДМК Пресс, 2017
17. Пэйн, Брайсон Python для детей и родителей. М.: «Э», 2017
18. Лучано Рамальо Python. К вершинам мастерства / Пер. с англ. Слинкин А. А. — М.: ДМК Пресс, 2016
19. Гифт Н., Джонс Д. Python в системном администрировании UNIX и Linux. СПб.: Символ-Плюс, 2009
20. М.Э. Абрамян. 1000 задач по программированию. Часть I. Методические указания для студентов механико-математического,

физического и экономического факультетов, ГОУ ПО РФ Ростовский государственный университет, 2004.

21. М.Э. Абрамян. 1000 задач по программированию. Часть II. Методические указания для студентов механико-математического, физического и экономического факультетов, ГОУ ПО РФ Ростовский государственный университет, 2004.

22. М.Э. Абрамян. 1000 задач по программированию. Часть III. Методические указания для студентов механико-математического, физического и экономического факультетов, ГОУ ПО РФ Ростовский государственный университет, 2004.

23. Златопольский Д.М. Сборник задач по программированию. – 3-е изд., перераб. и доп. – СПб.: БХВ-Петербург, 2011. – 304 с.: ил. – (ИиИКТ)

Список использованной литературы для обучающихся

1. Allen Downey. Думать на языке Python. Green Tea Press. 2012. Перевод на русский язык Николай Орехов 2017. https://bitbucket.org/thinkpython_ru/book/src

2. Федоров Д. Ю. Основы программирования на примере языка Python. //Учебное пособие. – Санкт-Петербург: 2016.

3. Васильев А.Н. Python на примерах. Практический курс по программированию. - СПб.: Наука и Техника, 2016. - 432 с.: ил.

4. Доусен М. Програмируем на Python. - СПб.: Питер, 2014. - 416 с.: ил.

5. Пейн, Брайсон. Python для детей и родителей / Брайсон Пейн. – Москва: Издательство «Э». 2017. – 352 с.: ил.

6. Хайнеман, Джордж, Пояяис, Гэри, Сеяков, Стэнли. Алгоритмы. Справочник с примерами на C, C++, Java и Python, 2-е изд.: Пер. с англ. — СПб.: ООО «Альфа-книга», 2017. — 432 с.: ил. — Парал. тит. англ.

7. Седжвик, Роберт, Уэйн, Кевин, Дондеро, Роберт. Программирование на языке Python: учебный курс. : Пер. с англ. - СПб. : ООО "Альфа-книга": 2017. - 736 с. : ил. - Парал. тит. англ.

8. У. Сэнд, К. Сэнд. Hello World! Занимательное программирование. — СПб.: Питер, 2016. — 400 с.: ил. — (Серия «Вы и ваш ребенок»).

Электронные образовательные ресурсы