

краевое государственное бюджетное учреждение дополнительного образования
«Камчатский центр детского и юношеского технического творчества»

«Согласовано»

Представитель экспертного совета
КГБУДО «Камчатский центр детского и
юношеского технического творчества»
структурного подразделения «IT
зам. директора по учебно- работе
Э.С. Бутенко
протокол № 4 от « 06 » мая 2022 года

«Утверждено»

Директор КГБУДО «Камчатский центр
детского и юношеского
технического
творчества»
А.А. Юхин
Протокол № 1 Педагогического совета
от « 29 » августа 2022 года



**Дополнительная общеобразовательная программа по
тематическому направлению «Программирование на языке
Python» с использованием оборудования центра цифрового
образования детей «IT-куб»**

Направленность: техническая

Возраст обучающихся: 12 – 15 лет

Срок реализации программы: 2 года (288 часов)
(общее количество часов по годам обучения)

Формы обучения: очная, очная с применением дистанционных технологий

Организация обучения: длительность обучения 9 месяцев. Групповая, при реализации программы с применением дистанционных образовательных технологий — персональная, материалы курса будут размещены в виртуальной обучающей среде.

Уровень программы: углублённый уровень

Статус программы: модифицированная

Составители:

Шкотин Даниил Вадимович,
педагоги дополнительного образования

ПОЯСНИТЕЛЬНАЯ ЗАПИСКА

В настоящее время, в 20-е г. XXI в., наше общество находится на этапе глобальной информатизации и компьютеризации. Поэтому возрастает потребность в специалистах с высоким уровнем владения информационными компетенциями, отвечающих социальному заказу по подготовке квалифицированных кадров в области программирования, а также обладающих высоким интересом к IT-сфере.

Одной из составляющих информационной компетентности является владение языком программирования. Встаёт вопрос о выборе языка программирования, который отвечает современным требованиям к написанию программ, служит основой для дальнейшего развития и совершенствования программистских компетенций.

Какой язык наиболее отвечает современному этапу развития процесса информатизации общества?

Для определения «популярности» языка программирования существует несколько рейтингов. Опишем кратко основные из них. Рейтинг TIOBE Index представляет собой анализ результатов поисковых запросов, содержащих название языка. В результате на первые позиции выходят те языки, названия которых чаще всего встречаются в поисковых запросах таких систем, как Google, Blogger, Wikipedia, YouTube, Baidu, Yahoo!, Bing, Amazon. Такой расчёт производится ежемесячно: так, по результатам рейтинга за август 2021 г. наиболее популярным языком является C. На второй и третьей позициях находятся Python и Java соответственно.

Язык программирования Python был представлен в 1990 г. Гвидо ван Россумом. В основе лежал язык ABC, который разрабатывался в центре математики и информатики в Нидерландах. Изначально в языке не была реализована концепция объектно-ориентированного программирования (ООП). В феврале 1991 г. был опубликован исходный текст языка. В него уже были заложены принципы ООП. Версия Python 2.0 была выпущена в 2000 г. В 2008 г.

вышла версия Python 3.0, которая не полностью поддерживает вторую версию языка. Версия Python 3.8 вышла 14 октября 2019 г.

Цель представленной дополнительной общеобразовательной программы по тематическому направлению «Программирование на языке Python» является изучение основ программирования на языке Python, основных приёмов написания программ на современном языке программирования, развитие алгоритмического мышления учащихся, творческих способностей, аналитических и логических компетенций.

ЦЕЛЬ

Целью программы является изучение основ программирования на языке Python, основных приёмов написания программ на современном языке программирования, развитие алгоритмического мышления учащихся, творческих способностей, аналитических и логических компетенций.

ЗАДАЧИ:

Познавательные задачи:

- освоение среды программирования Python в качестве инструмента для программирования программ;
- изучение основ программирования;
- освоение среды разработки графических интерфейсов QT Designer;
- создание реляционных баз данных;
- создание чат-ботов;
- создание завершённых проектов с использованием освоенных навыков.

Регулятивные задачи:

- формирование навыков планирования — определения последовательности промежуточных целей с учётом конечного результата;

- освоение способов контроля в форме сопоставления способа действия и его результата с заданным образцом с целью обнаружения отличий от эталона.

Коммуникативные задачи:

- формирование умения работать над проектом в команде;
- овладением умением эффективно распределять обязанности.

ВОЗРАСТ ОБУЧАЮЩИХСЯ

Программа адресована детям в возрасте 12-15 лет.

Условия набора детей в коллектив: принимаются все желающие (не имеющие медицинских противопоказаний).

Наполняемость в объединении устанавливается в количестве до 12 обучающихся.

Уровень освоения: программа является общеразвивающей (углублённый уровень), не требует предварительных знаний и входного тестирования.

УСЛОВИЯ РЕАЛИЗАЦИИ ПРОГРАММЫ

Программа рассчитана на 144 учебных часа в год. Срок реализации – 2 года.

Занятия проводятся в группах 2 раза в неделю по 2 часа, то есть 4 часа в неделю.

Для успешной реализации программы «Программирование на языке Python» необходимо:

- наличие учебной аудитории;
- рабочее место для обучающихся – 12 шт.;
- рабочее место для педагога – 1 шт.;
- моноблочное интерактивное устройство – 1 шт.;
- напольная мобильная стойка для интерактивных досок – 1 шт.;
- флипчарт – 1 шт.;

- МФУ (принтер, сканер, копир) – 1 шт.;
- ноутбук для обучающихся – 12 шт.;
- ноутбук для педагога – 1 шт.;
- WEB – камера – 1 шт.;
- наушники – 12 шт.

Также необходимо наличие следующего программного обеспечения:

- PyCharm Community;
- QT Designer;
- Microsoft Access;
- DB Browser (SQLite).

ВИДЫ ОЦЕНКИ РЕЗУЛЬТАТИВНОСТИ УЧЕБНЫХ ЗАНЯТИЙ

Предметом диагностики и контроля являются внешние образовательные продукты обучающихся (созданные работы), а также их личностные качества (освоенные способы деятельности, знания, умения), которые относятся к целям и задачам программы. Основой для оценивания деятельности обучающихся являются результаты анализа его продукции и деятельности по её созданию. Оценка имеет различные способы выражения: устные суждения педагога, письменные качественные характеристики. Оценке подлежит в первую очередь уровень достижения обучающимся минимально необходимых результатов.

Для оценки результативности процесса обучения предусматриваются следующие виды контроля:

- вводный (для выяснения знаний, умений и навыков воспитанников на начало учебного года);
- промежуточный (в середине учебного года по пройденным разделам или темам);
- итоговый (после завершения всей учебной программы по годам обучения).

Проверка достигаемых обучающимися образовательных результатов

проводится в следующих формах:

- текущая диагностика и оценка педагогом деятельности обучающихся;
- текущий контроль осуществляется по результатам выполнения практических заданий, мини-проектов (при этом тематические состязания роботов также являются методом проверки), тесты по теме;
- взаимооценка обучающимися работ друг друга или работ, выполненных в группах;
- публичная защита выполненных обучающимися творческих работ (индивидуальных и групповых).

Цель контроля — диагностика имеющихся знаний и умений, оценка качества усвоения материала. Также, контроль проводится с целью выяснения, каким воспитанникам требуется больше уделить внимание и оказать вовремя помощь, какие темы были наиболее интересными, а какие более сложными для детей.

Кроме того, оценивать проделанную работу необходимо в конце каждой темы. Оценку даёт педагог. Для закрепления полученных знаний и умений большое значение имеет коллективный анализ работ. При этом отмечаются наиболее удачные решения, оригинальные подходы к выполнению задания, разбираются характерные ошибки.

Основной формой контроля являются конкурсы, выставки, соревнования и т.д. Участие в мероприятиях различного уровня характеризуют степень усвоения программного материала обучающимися.

Выполненные обучающимися работы включаются в их «портфель достижений». Итоговый контроль проводится в конце всего курса обучения. Он может иметь форму защиты проектной работы. Данный тип контроля предполагает комплексную проверку образовательных результатов по всем заявленным целям и направлениям курса. Формой итоговой оценки каждого обучающегося выступает характеристика, в которой указывается уровень освоения им образовательного курса.

ОЖИДАЕМЫЕ РЕЗУЛЬТАТЫ

Личностные результаты:

- формирование умения самостоятельной деятельности;
- формирование умения работать в команде;
- формирование коммуникативных навыков;
- формирование навыков анализа и самоанализа;
- формирование целеустремлённости и усидчивости в процессе творческой, исследовательской работы и учебной деятельности.

Предметные результаты:

- формирование понятий «алгоритм», «программа», «база данных», «автоматизированная информационная система»;
- формирование понятий об основных конструкциях языка программирования Python, таких как оператор ветвления if, операторы цикла while, for, вспомогательные алгоритмы;
- формирование понятий о структурах данных языка программирования Python;
- формирование основных приёмов составления программ на языке программирования Python;
- формирование алгоритмического и логического стилей мышления.

Метапредметные результаты:

- формирование умения ориентироваться в системе знаний;
- формирование умения выбирать наиболее эффективные способы решения задач на компьютере в зависимости от конкретных условий;
- формирование приёмов проектной деятельности, включая умения видеть проблему, формулировать тему и цель проекта, составлять план своей деятельности, осуществлять действия по реализации плана, результат деятельности соотносить с целью, классифицировать, наблюдать, проводить эксперименты, делать выводы и заключения, доказывать, защищать свои идеи,

оценивать результаты своей работы;

- формирование умения распределять время;
- формирование умений успешной самопрезентации.

УЧЕБНЫЙ ПЛАН

Год обучения	Нагрузка (час. в неделю)	Кол-во обуч-ся	Возраст обуч-ся	Всего часов	Из них	
					теория	практика
I	4	12	12 - 15 лет	144	61	83
II	4	12	12 - 15 лет	144	51	93

УЧЕБНО-ТЕМАТИЧЕСКИЙ ПЛАН

1 год обучения

№ п/п	Наименование модуля	Количество часов		
		всего	теория	практика
1.	Модуль 1. Вводное занятие	2	2	-
2.	Модуль 2. Знакомство со средой программирования на языке Python. Переменные	14	7	7
3.	Модуль 3. Условными инструкциями	16	8	8
4.	Модуль 4. Циклы	20	10	10
5.	Модуль 5. Самостоятельная работа по темам «Переменная», «Условия», «Цикл»	6	-	6
6.	Модуль 6. Строки	10	5	5
7.	Модуль 7. Список	14	7	7
8.	Модуль 8. Функция	14	7	7
9.	Модуль 9. Самостоятельная работа по темам «Строки», «Список», «Функция»	6	-	6
10.	Модуль 10. Массив	4	2	2
11.	Модуль 11. Множество	4	2	2
12.	Модуль 12. Словари	18	9	9
13.	Модуль 13. Самостоятельная работа по темам	6	-	6

	«Массив», «Множества, «Словари»			
14.	Модуль 14. Проектная деятельность	8	1	7
15.	Модуль 15. Итоговое занятие	2	1	1
	Итого:	144	61	83

СОДЕРЖАНИЕ УЧЕБНЫХ МОДУЛЕЙ

1 год обучения

Модуль 1. Вводное занятие.

Всего 2 часа: из них: теоретических — 2.

Краткое содержание.

Техника безопасности. Пожарная безопасность. Террористическая безопасность.

Модуль 2. Знакомство со средой программирования на языке Python.

Переменные.

Всего 14 часов: из них: теоретических — 7; практических - 7.

Краткое содержание.

Знакомство со средой программирования на языке Python, изучение основных элементов интерфейса, запуск программы. Изучение арифметических операторов, операторов присваивания, операторов сравнения, логических операторов, операторов принадлежности. Изучение типов данных на Python.

Практическая часть.

Написание простых программ на языке программирования Python, реализующих линейные алгоритмы. Создание переменных и присвоение им данных с помощью функции input() и вывод данных с помощью функции print ().

Модуль 3. Условными инструкциями

Всего 16 часов: из них: теоретических — 8; практических - 8.

Краткое содержание.

Изучение условного оператора и конструкции if-else.

Практическая часть.

Создание программ условного оператора if с конструкцией if-else. Разработка программ, реализующих условные алгоритмы if.

Модуль 4. Циклы

Всего 20 часов: из них: теоретических — 10; практических - 10.

Краткое содержание.

Знакомство с циклами for, while на языке программирования Python.
Изучение работы вложенных циклов.

Практическая часть.

Применение на практике циклов и ветвлений. Использование циклов и ветвлений для решения математических задач.

Модуль 5. Самостоятельная работа по темам «Переменная», «Условия», «Цикл».

Всего 6 часа: из них: практических - 6.

Практическая часть.

Решение задач по темам переменная, условия и циклы. Самостоятельная работа для закрепления знаний по темам операторов, условий и циклов.

Модуль 6. Строки

Всего 10 часов: из них: теоретических — 5; практических - 5.

Краткое содержание.

Изучение строк, функций строк, методов строк. Изучение понятия «конкатенация». Изучение срезов строк.

Практическая часть.

Разработка программ, реализующих методы и функции строк, конкатенацию. Разработка программ, реализующих работу строк через срезы.

Модуль 7. Список

Всего 14 часов: из них: теоретических — 7; практических - 7.

Краткое содержание.

Создание списков, изменение, работа с данными из списка. Изучение методов списков. Изучение способов создания, изменения списков с помощью циклов. Изучение вложенных списков.

Практическая часть.

Создание списков, изменение, работа с данными из списка. Работа с вложенными списками.

Модуль 8. Функция

Всего 14 часов: из них: теоретических — 7; практических - 7.

Краткое содержание.

Изучение структуры простой и сложной структур функций. Изучение алгоритмов с использованием условий и функций.

Практическая часть.

Создание программ, используя простые функции. Создание программ, используя сложные функции. Создание программ, используя циклы и функции. Создание программы с функцией в цикле.

Модуль 9. Самостоятельная работа по темам «Строки», «Список», «Функция»

Всего 6 часов: из них: теоретических — 3; практических - 3.

Краткое содержание.

Закрепление полученных ранее знаний по темам «Строки», «Список», «Функция».

Практическая часть.

Решение задач по темам строки, списки, функции.

Модуль 10. Массив

Всего 4 часа: из них: теоретических — 2; практических - 2.

Краткое содержание.

Знакомство с понятием «массив». Изучение методов массива.

Практическая часть.

Разработка программ, реализующих массивы. Создание программы с массивами, используя методы массивов.

Модуль 11. Множество

Всего 4 часа: из них: теоретических — 2; практических - 2.

Краткое содержание.

Знакомство с понятием «множество». Изучение методов множеств.

Практическая часть.

Разработка программ, реализующих множества. Создание программы с множествами, используя методы множества.

Модуль 12. Словари

Всего 18 часов: из них: теоретических — 9; практических - 9.

Краткое содержание.

Знакомство с понятием «Словарь». Создание словарей, изменение, работа с данными из списка. Изучение методов словарей, способов создания, изменения словарей с помощью циклов.

Практическая часть.

Работа с методами списков, словарями с помощью циклов, вложенными словарями. Создание программы с использованием вложенных списков в словари, вложенные словари в списки.

Модуль 13. Самостоятельная работа по темам «Массив», «Множества», «Словари»

Всего 6 часов: из них: теоретических — 3; практических - 3.

Краткое содержание.

Закрепление полученных ранее знаний по темам «Массив», «Множества», «Словари».

Практическая часть.

Решение задач по темам массивы, множества и словари.

Модуль 14. Проектная деятельность

Всего 8 часов: из них: теоретических — 4; практических - 4.

Краткое содержание.

Распределение проектов между группами.

Практическая часть.

Разработка группового проекта на языке программирования Python.

Модуль 15. Итоговое занятие

Всего 2 часа: из них: теоретических — 1; практических - 1.

Краткое содержание.

Защита обучающимися проектов, подведение итогов.

Практическая часть.

Защита групповых проектов, подведение итогов курса.

УЧЕБНО-ТЕМАТИЧЕСКИЙ ПЛАН

2 год обучения

№ п/п	Наименование модуля	Количество часов		
		всего	теория	практика
1	Модуль 1. Вводное занятие	2	2	0
2	Модуль 2. Повторение изученных тем	18	7	11
3	Модуль 3. Изучение среды разработки графических интерфейсов QT Designer	20	10	10
4	Модуль 4. Создание программ с помощью среды разработки графических интерфейсов QT Designer	24	3	21
5	Модуль 5. Изучение языка программирования SQL для работы с реляционными базами данных	20	10	10
6	Модуль 6. Создание автоматизированной информационной системы с использованием QT и SQL	26	9	17
7	Модуль 7. Создание и проектирование чат-бота Telegram	24	8	16
8	Модуль 8. Проектная деятельность	8	1	7
9	Модуль 9. Итоговое занятие	2	1	1
Итого:		144	51	93

СОДЕРЖАНИЕ УЧЕБНЫХ МОДУЛЕЙ

2 год обучения

Модуль 1. Вводное занятие.

Всего 2 часа: из них: теоретических – 2.

Краткое содержание.

Техника безопасности. Пожарная безопасность. Террористическая безопасность.

Модуль 2. Повторение изученных тем.

Всего 18 часов: из них: теоретических – 7; практических – 11.

Краткое содержание.

Повторение изученных тем 1 года обучения.

Практическая часть.

Написание программ на языке программирования Python для закрепления навыков и знаний по изученным темам 1 года обучения.

Модуль 3. Изучение среды разработки графических интерфейсов QT Designer.

Всего 20 часов: из них: теоретических – 10; практических – 10.

Краткое содержание.

Знакомство со средой разработки графических интерфейсов QT Designer, изучение основных элементов интерфейса, а также его компонентов, их свойств и методов.

Практическая часть.

Написание простых программ с компонентами в среде разработки графических интерфейсов QT Designer и практическое изучение их свойств и методов с помощью языка программирования Python.

Модуль 4. Создание программ с помощью среды разработки графических интерфейсов QT Designer.

Всего 24 часов: из них: теоретических – 3; практических – 21.

Краткое содержание.

Создание известных классических игр с использованием среды разработки графических интерфейсов QT Designer и языка программирования Python.

Практическая часть.

Применение на практике методов и свойств компонентов среды разработки графических интерфейсов QT Designer, и написание известных классических игр с помощью языка программирования Python.

Модуль 5. Изучение языка программирования SQL для работы с реляционными базами данных

Всего 20 часа: из них: теоретических – 10; практических – 10.

Краткое содержание.

Изучение основ проектирования баз данных с помощью реляционной системы управления базами данных Microsoft Access и встраиваемой системы управления базами данных SQLite.

Практическая часть.

Проектирование различных баз данных с помощью реляционной системы управления базами данных Microsoft Access и встраиваемой системы управления базами данных SQLite, изучение методов модуля Python – SQLite3.

Модуль 6. Создание автоматизированной информационной системы с использованием QT и SQL.

Всего 26 часов: из них: теоретических – 9; практических – 17.

Краткое содержание.

Изучение понятия «АИС» и требований для создания АИС. Проектирование базы данных, интерфейса и функционала АИС. Выработка умений по тестированию программ.

Практическая часть.

Создание АИС с помощью среды программирования Python, среды разработки графических интерфейсов QT Designer и встраиваемой системы управления базами данных SQLite. Создание базы данных, интерфейса и функционала АИС. Тестирование программы.

Модуль 7. Создание и проектирование чат-бота Telegram.

Всего 24 часов: из них: теоретических – 8; практических – 16.

Краткое содержание.

Изучение библиотеки Telebot для создания Telegram-бота.

Практическая часть.

Изучение библиотеки Telebot с помощью среды программирования Python.

Создание базы данных и функционала Telegram-бота, а также его тестирования.

Модуль 8. Проектная деятельность

Всего 8 часов: из них: теоретических – 1; практических – 7.

Краткое содержание.

Распределение проектов между группами и разработка группового проекта.

Практическая часть.

Разработка группового проекта на языке программирования Python.

Модуль 9. Итоговое занятие

Всего 2 часа: из них: теоретических – 1; практических – 1.

Краткое содержание.

Защита обучающимися проектов, подведение итогов курса.

Практическая часть.

Защита групповых проектов, подведение итогов курса.

ЛИТЕРАТУРА

1. Гэддис Т. Начинаем программировать на Python // Т. Гэддис. – БХВ-Петербург, 2018. – 768 с.
2. Доусон М. Програмируем на Python // М. Доусон. – Питер, 2019. – 416 с.
3. Дронов В., Прохоренок Н. Python 3 и PyQt 5. Разработка приложений // В. Дронов, Н. Прохоренок . – БХВ-Петербург, 2018. – 832 с.
4. Златопольский Д. Основы программирования на языке Python // Д. Златопольский. – ДМК Пресс, 2018. – 396 с.
5. Персиваль Г. Python. Разработка на основе тестирования // Г. Персиваль. – ДМК Пресс, 2018. – 622 с.
6. Свейгарт Э. Учим Python, делая крутые игры //Э. Свейгарт. – Эксмо, 2018. – 416 с.
7. Солем Я. Э. Программирование компьютерного зрения на Python // Я. Э. Солем. – ДМК Пресс, 2016. – 312 с.
8. QT Documentation [Электронный ресурс]. Режим доступа: <https://doc.qt.io/>, свободный.