

**ГОСУДАРСТВЕННОЕ БЮДЖЕТНОЕ ОБЩЕОБРАЗОВАТЕЛЬНОЕ УЧРЕЖДЕНИЕ
"ЛИЦЕЙ №1 г. СУНЖА»
ЦЕНТР ЦИФРОВОГО ОБРАЗОВАНИЯ ДЕТЕЙ «IT-КУБ»**

Принято
на заседании педагогического
совета центра цифрового образования
детей «IT-куб»

протокол № 1 от 10.09.2023 г.

Согласовано
руководитель центра цифрового
образования детей «IT-куб»


А.М.Чербижев.
« 10 » 09 2023г



директор ГБОУ «Лицей №1 г. Сунжа».

Х.Х.Черкиева.

**ДОПОЛНИТЕЛЬНАЯ ОБЩЕОБРАЗОВАТЕЛЬНАЯ ОБЩЕРАЗВИВАЮЩАЯ
ПРОГРАММИРОВАНИЕ НА ЯЗЫКЕ PYTHON (ЯНДЕКС.ЛИЦЕЙ)**

Дзаитов Зелимхан Макшарипович

г.Сунжа
2023-2024 учебный год.

ОГЛАВЛЕНИЕ

ПОЯСНИТЕЛЬНАЯ ЗАПИСКА	3
УЧЕБНО-ТЕМАТИЧЕСКИЙ ПЛАН	13
СОДЕРЖАНИЕ ПРОГРАММЫ.....	18
МЕТОДИЧЕСКОЕ ОБЕСПЕЧЕНИЕ ПРОГРАММЫ	37
СПИСОК ЭЛЕКТРОННЫХ ИСТОЧНИКОВ ИНФОРМАЦИИ	42

ПОЯСНИТЕЛЬНАЯ ЗАПИСКА

В настоящее время мы переживаем большие изменения в развитии общества. В современную жизнь человека все больше и больше внедряются компьютеры и информационные технологии. Всё большее значение приобретает умение человека грамотно обращаться с компьютером, причем зачастую не на пользовательском уровне, а на уровне начинающего программиста.

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 учащимися основ современного программирования и неумение применять полученные знания на практике.

Изучение основных принципов программирования невозможно без регулярной практики написания программ на каком-либо языке. В данной программе выбран язык программирования 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.

Направленность программы

Программа имеет техническую направленность, в рамках которой рассматриваются следующие аспекты изучения:

1. Технологический. Содержание программы рассматривается как средство формирования образовательного потенциала, позволяющего развивать наиболее передовые на сегодняшний день технологии — информационные, интегрирующие в себе науку, технологию, инженерное дело.

2. **Общеразвивающий.** Обучение по данной программе создает благоприятные условия для интеллектуального и духовного воспитания личности ребенка, социально-культурного и профессионального самоопределения, развития познавательной активности и творческой самореализации учащихся.

3. **Общеобразовательный.** Содержание программы рассматривается как средство развития основных познавательных процессов, умения анализировать, выявлять сущности и отношения, описывать планы действий и делать логические выводы, опираясь на такие дисциплины, как теория управления, программирование, теория информации.

Актуальность программы

Актуальность данной программы состоит в том, что она составлена с учётом современных потребностей рынка в специалистах в области информационных технологий, особенно в области программирования.

Научившись программировать на языке Python, учащиеся получают мощный и удобный инструмент для решения как учебных, так и прикладных задач. Вместе с тем чистота и ясность его конструкций позволит учащимся потом с легкостью выучить любой другой язык программирования.

Знания и умения, приобретенные 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.

Данная программа дает возможность детям творчески мыслить, находить самостоятельные индивидуальные решения, а полученные умения и навыки применять в жизни. Развитие творческих способностей помогает также в профессиональной ориентации подростков.

Новизна программы

Новизна программы состоит в том, что она учитывает новые технологические уклады, которые требуют новый способ мышления и тесного

взаимодействия при постоянном повышении уровня междисциплинарности проектов, а также использует новые формы диагностики и подведения итогов реализации программы, выполняемые в формате защиты проектов.

В основу программы «Программирование на Python» заложены принципы модульности практической направленности, что обеспечит вариативность обучения. Содержание учебных модулей направлено на:

- детальное изучение алгоритмизации;
- реализацию межпредметных связей;
- организацию проектной и исследовательской деятельности.

Цели программы:

– создание условий для изучения методов программирования на языке Python;

– рассмотрение различных парадигм программирования, предлагаемых этим языком (процедурная, функциональная, объектно-ориентированная);

– подготовка к использованию как языка программирования, так и методов программирования на Python в учебной и последующей профессиональной деятельности в различных предметных областях.

Задачи:

Образовательные:

• познакомить с принципами и методами фундаментального программирования;

• познакомить с принципами и методами объектно-ориентированного программирования;

• привить навыки работы в интегрированной среде разработки на языке Python;

• изучить конструкции языка программирования Python;

• привить навыки разработки эффективных алгоритмов и программ на основе изучения языка программирования Python;

• дать представление о значении информационных технологий в развитии общества и в изменении характера труда человека.

Воспитательные:

- воспитать мотивацию учащихся к изобретательству, созданию собственных программных реализаций;
- привить стремление к получению качественного законченного результата в проектной деятельности;
- привить информационную культуру: ответственное отношение к информации с учетом правовых и этических аспектов её распространения, избирательного отношения к полученной информации;
- формировать правильное восприятие системы ценностей, принципов, правил информационного общества;
- формировать потребность в самостоятельном приобретении и применении знаний, потребность к постоянному саморазвитию;
- воспитывать социально-значимые качества личности человека: ответственность, коммуникабельность, добросовестность, взаимопомощь, доброжелательность.

Развивающие:

- способствовать развитию навыков алгоритмического и логического мышления, грамотной разработки программ;
- способствовать приобретению навыков поиска информации в сети Интернет, анализ выбранной информации на соответствие запросу, использование информации при решении задач;
- развивать познавательные способности ребенка, память, внимание, пространственное мышление, аккуратность и изобретательность;
- формировать творческий подход к поставленной задаче;
- развивать навыки инженерного мышления, умения работать как по предложенным инструкциям, так и находить свои собственные пути решения поставленных задач;
- развивать навыки эффективной деятельности в проекте;
- развивать стрессоустойчивость;
- формировать и развивать навыки публичного выступления;

- развивать способности к самоанализу, самопознанию;
- формировать навыки рефлексивной деятельности.

Отличительные особенности программы

Программа предназначена для учащихся, проявляющих повышенный интерес к программированию. Программа имеет практическую направленность с ориентацией на реальные потребности, соответствующие возрасту ученика.

Ключевым элементом обучения является проектная деятельность, которая ориентирована на использование знаний, умений и навыков, полученных в ходе обучения, для постановки и решения практических задач, которые носят прикладной характер. Она позволяет учащимся участвовать в создании конкретного результата и научиться работать в условиях ограниченного времени, под руководством заказчика, презентовать проект, а также обрести навыки профессиональной коммуникации с контрагентами.

Категория обучающихся

Программа предназначена для детей, проявляющих интерес к программированию, стремящимся к саморазвитию, профессиональному самоопределению, имеющим начальные представления о языках программирования.

Возраст обучающихся: 14 — 17 лет.

Наполняемость группы: 10 человек.

Состав группы: разновозрастной.

Условия приема детей

На курсы программы зачисляются учащиеся прошедшие отбор в два этапа: первый – дистанционное онлайн тестирование, второй – очное собеседование с преподавателем.

Срок реализации программы: 2 года.

Структура программы:

Программа первого года обучения «Основы программирования на языке Python» состоит из 6 модулей:

Модуль 1. Введение в программирование на Python.

Модуль 2. Множества, строки, списки и словари в Python.

Модуль 3. Функции в Python.

Модуль 4. Библиотеки Python.

Модуль 5. Объектно-ориентированное программирование.

Модуль 6. Практическое применение языка Python.

Программа второго года обучения «Основы промышленного программирования» состоит из 3 модулей:

Модуль 1. Разработка GUI приложений.

Модуль 2. Pygame, создание игр на Python.

Модуль 3. Web-разработка.

Форма реализации программы — очная с использованием электронного обучения.

Под электронным образованием понимается реализация образовательных программ с использованием информационно-образовательных ресурсов, информационно-коммуникационных технологий, технических средств, а также информационно-телекоммуникационных сетей, обеспечивающих передачу информационно-образовательных ресурсов и взаимодействие участников образовательного пространства.

Формы организации деятельности обучающихся

При изучении тем программа предусматривает использование фронтальной, индивидуальной и групповой формы учебной работы обучающихся.

При организации занятий по программе «Программирование на языке Python» для достижения поставленных целей и задач используются формы проведения занятий с активными методами обучения:

- занятие в форме проблемно-поисковой деятельности;
- занятие с использованием межпредметных связей;
- занятие в форме мозгового штурма;
- занятие в форме частично-поисковой деятельности.

Методы обучения

Основным методом обучения является метод проектов.

Методика обучения ориентирована на индивидуальный подход. Для того чтобы каждый ученик получил наилучший результат обучения, программой предусмотрены индивидуальные задания для самостоятельной работы на компьютере. Такая форма организации обучения стимулирует интерес ученика к предмету, активность и самостоятельность учащихся, способствует объективному контролю глубины и широты знаний, повышению качества усвоения материала обучающимися, позволяет педагогу получить объективную оценку выбранной им тактики и стратегии работы, методики индивидуального обучения и обучения в группе, выбора предметного содержания.

Для самостоятельной работы используются разные по уровню сложности задания, которые носят репродуктивный и творческий характер.

В ходе обучения проводится промежуточные тестирование по темам для определения уровня знаний учеников. Выполнение контрольных заданий способствует активизации учебно-познавательной деятельности и ведёт к закреплению знаний, а также служит индикатором успешности образовательного процесса.

Типы занятий: теоретические, практические, комбинированные, контрольные.

Режим занятий

Два часа по два раза в неделю с разницей между занятиями не менее 2 дней.

Ожидаемые результаты

Основным результатом обучения является достижение высокой информационно-коммуникационной компетентности учащегося в области программирования на языке Python.

В процессе занятий проводится целенаправленная работа на достижение личностных, метапредметных и предметных результатов обучения, обозначенных ФГОС.

Предметные:

- знание правил работы с персональным компьютером и технику безопасности;

- понимание назначения и функции используемых информационных технологий;

- знание особенностей работы с интегрированной средой разработки;

- оперирование базовыми и сложными конструкциями, знание способов организации процедур и функций в языке программирования Python;

- умение эффективно использовать интегрированную среду разработки;

- определять результат выполнения алгоритма при заданных исходных данных, узнавать изученные алгоритмы обработки чисел и числовых последовательностей, создавать на их основе программы анализа данных, читать и понимать программы, написанные на языке программирования высокого уровня Python;

- выполнять пошагово алгоритмы управления исполнителями и анализа числовых и текстовых данных;

- 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 писать код программы на языке Python, используя специальные средства и библиотеки языка Python.

Метапредметные:

- умение искать информацию с применением правил поиска в компьютерных сетях при выполнении заданий и проектов по различным темам;

- умение следовать требованиям техники безопасности, гигиены, эргономики и ресурсосбережения при работе со средствами информационных и коммуникационных технологий;

- навыки проектирования, разработки, документирования и

представления собственных проектов в составе команды;

– умение самостоятельно определять цели своего обучения, ставить и формулировать для себя новые задачи в учебе и познавательной деятельности, развивать мотивы и интересы своей познавательной деятельности;

– умение соотносить свои действия с планируемыми результатами, осуществлять контроль своей деятельности в процессе достижения результата, определять способы действий в рамках предложенных условий и требований, корректировать свои действия в соответствии с изменяющейся ситуацией;

– умение оценивать правильность выполнения учебной задачи, собственные возможности ее решения;

– владение основами самооценки, принятия решений и осуществления осознанного выбора в учебной и познавательной деятельности;

– умение определять понятия, создавать обобщения, устанавливать аналогии, классифицировать, самостоятельно выбирать основания и критерии для классификации, устанавливать причинно-следственные связи, строить логическое рассуждение, умозаключение (индуктивное, дедуктивное и по аналогии) и делать выводы;

– умение организовывать учебное сотрудничество и совместную деятельность с учителем и сверстниками; работать индивидуально и в команде; формулировать, аргументировать и отстаивать свое мнение.

Личностные:

– готовность и способность обучающихся к самообразованию, саморазвитию, личностному и профессиональному самоопределению;

– сформированность мотивации к обучению и целенаправленной познавательной деятельности;

– систему значимых социальных и межличностных отношений, ценностно-смысловых установок, отражающих личностные и гражданские

– умение ставить цели и строить жизненные планы;

– способность к осознанию российской идентичности в поликультурном социуме.

Способы определения результативности

Педагогическое наблюдение, педагогический анализ результатов решения задач с использованием автоматизированной системы контроля знаний, результаты участия в интеллектуальных конкурсах всероссийского уровня.

Виды контроля:

- тестирование;
- устный опрос;
- самостоятельные и контрольные работы;
- участие в проектной деятельности.

Формы подведения итогов реализации программы

В течение первого года обучения проводятся промежуточные аттестации в формате контрольного тестирования и выполнения самостоятельных работ, а также решения задач. За выполнение заданий учащимся начисляются баллы рейтинга в системе LMS Яндекс.Лицей. Также по итогу первого года обучения учащимся проводится аттестация в форме публичной защиты. Документальной формой подтверждения итогов первого года обучения является документ об образовании установленного филиалом Центра «Поиск» образца.

После первого года обучения проходит отбор учащихся для перевода на второй год по результатам рейтинга системы LMS Яндекс.Лицей.

В течение второго года обучения проводятся промежуточные аттестации в формате проектных заданий, за выполнение которых учащимся также начисляются баллы рейтинга в системе LMS Яндекс.Лицей. По окончании второго года обучения проводится итоговая аттестация в форме публичной защиты проектов. Документальной формой подтверждения итогов второго года обучения является документ об образовании установленного филиалом Центра «Поиск» образца и сертификат интеллектуального партнера «Яндекс.Лицей».

УЧЕБНО-ТЕМАТИЧЕСКИЙ ПЛАН

ПЕРВЫЙ ГОД ОБУЧЕНИЯ

РАЗДЕЛ «ОСНОВЫ ПРОГРАММИРОВАНИЯ НА ЯЗЫКЕ PYTHON»

№	Наименование кейса, темы	Количество часов		
		Теория	Практика	Всего
Модуль 1. Введение в программирование на Python.		9	16	25
1	Тема 1.1. Знакомство со средой.	1	2	3
2	Тема 1.2. Условный оператор.	1	2	3
3	Тема 1.3. Простые встроенные функции.	1	2	3
4	Тема 1.4. Знакомство с циклом while.	1	2	3
5	Тема 1.5. Отладчик.	1	2	3
6	Тема 1.6. Знакомство с циклом for.	1	2	3
7	Тема 1.7. Погружение в условия. True, False, Break, Continue.	1	2	3
8	Тема 1.8. Вложенные циклы.	2	2	4
Модуль 2. Множества, строки, списки и словари в Python.		11	15	26
9	Тема 2.1. Множества	1	2	3
10	Тема 2.2. Строки. Индексация	1	2	3
11	Тема 2.3. Строки. Срезы.	1	2	3
12	Тема 2.4. Знакомство со списками.	1	2	3
13	Тема 2.5. Кортежи. Преобразование коллекций.	2	1	3
14	Тема 2.6. Методы Split и join. Списочные выражения.	2	1	3
15	Тема 2.7. Методы списков и строк.	1	2	3
16	Тема 2.8. Вложенные списки.	1	2	3
17	Тема 2.9. Знакомство со словарями.	1	1	2
Модуль 3. Функции в Python.		7	14	21
18	Тема 3.1. Функции.	1	2	3
19	Тема 3.2. Функции. Возвращение значений из функций.	1	2	3
20	Тема 3.3. Области видимости переменных.	1	2	3

21	Тема 3.4. Функции: передача параметров.	1	2	3
22	Тема 3.5. Функции с переменным числом аргументов.	1	2	3
23	Тема 3.6. Функции как объект. Лямбда-функции.	1	2	3
24	Тема 3.7. Обработка коллекций. Поточковый ввод sys.stdin.	1	2	3
Модуль 4. Библиотеки Python.		5	13	18
25	Тема 4.1.Random.	1	2	3
26	Тема 4.2. Графика.	1	2	3
27	Тема 4.3. Графика + звук.	1	3	4
28	Тема 4.4. Морфология.	1	3	4
29	Тема 4.5. Документы.	1	3	4
Модуль 5. Объектно-ориентированное программирование.		8	13	21
30	Тема 5.1. Введение в ООП.	2	4	6
31	Тема 5.2. Полиморфизм.	2	3	5
32	Тема 5.3. Определение операторов.	2	3	5
33	Тема 5.4. Наследование.	2	3	5
Модуль 6. Практическое применение языка Python.		11	14	25
34	Тема 6.1. Основы ведения проектной деятельности и «Тайм-менеджмент».	1	1	2
35	Тема 6.2. Постановка проблемной ситуации и поиск путей решения. Деление на команды.	2	3	5
36	Тема 6.3. Работа команд над реализацией своего проекта.	2	3	5
37	Тема 6.4. Работа над возможными ошибками и недочётами в проекте.	1	2	3
38	Тема 6.5. Подытожим и суммируем полученные навыки и знания по результатам работы над проектом.	1	2	3
39	Тема 6.6. Готовим презентацию для итоговой защиты.	2	2	4
40	Тема 6.7. Итоговая защита.	2	1	3
Итого		51	85	136

ВТОРОЙ ГОД ОБУЧЕНИЯ

РАЗДЕЛ «ОСНОВЫ ПРОМЫШЛЕННОГО ПРОГРАММИРОВАНИЯ»

№	Наименование кейса, темы	Количество часов		
		Теория	Практика	Всего
Модуль 1.Разработка GUI приложений.		15	31	46
1	Тема 1.1. Повторение. Решение задач на основные конструкции и структуры данных. Решение задач на классы и библиотеки.	2	4	6
2	Тема 1.2. Основные команды при одиночной работе с GIT	1	1	2
3	Тема 1.3. Что такое QT и PyQT. QtDesigner, pyuis, два способа подключения uic-файла.	2	2	4
4	Тема 1.4. Обработка исключений. Создание собственных исключений.	1	1	2
5	Тема 1.5. Файлы в Python. Работа с различными типами файлов. Использование библиотек.	2	2	4
6	Тема 1.6. Диалоги, работа с изображениями	1	1	2
7	Тема 1.7. Самостоятельная работа на файлы.		2	2
8	Тема 1.8. Работа с простыми таблицами (csv). Фильтрация, сортировка, вывод. Работа с табличными данными в PyQT.	1	1	2
9	Тема 1.9. Введение в БД, работа с SQL-таблицами и отображение данных в PyQT	2	4	6
10	Тема 1.10. Установка дополнительных компонентов. PyQTGraph	1	1	2
11	Тема 1.11. Обработка клавиатуры и курсора. Сборка независимого приложения.	2	2	4
12	Тема 1.14. Самостоятельная работа на SQL.		2	2
13	Тема 1.12. Работа над проектами индивидуально.		6	6

14	Тема 1.13. Защита проектов по модулю.		2	2
Модуль 2. Pygame, создание игр на Python.		10	22	32
15	Тема 2.1. Введение в репозитории. Подключение в PyCharm. Работа с удаленным репозиторием.	1	1	2
16	Тема 2.2. Введение в Pygame. Игровой цикл. События.	2	2	4
17	Тема 2.3. Клетчатое поле.	2	4	6
18	Тема 2.4. Совместная работа над проектом, основные понятия. Работа с репозиториями в среде разработки.	1	1	2
19	Тема 2.5. Изображения. Спрайты. Столкновения и другие взаимодействия.	2	2	4
20	Тема 2.6. Разработка и украшение игры.	2	2	4
21	Тема 2.7. Работа над проектами в команде.		8	8
22	Тема 2.8. Защита проектов по модулю.		2	2
Модуль 3. Web-разработка.		20	46	66
23	Тема 3.1. Работа с популярными форматами файлов (json, zip).	1	1	2
24	Тема 3.2. Знакомство с API.	1	1	2
25	Тема 3.3. Работа с командной строкой (скрипты, аргументы). Периодические задачи (модуль schedule).	1	1	2
26	Тема 3.4. Библиотека argparse. Задачи на создание скриптов с ее помощью.	1	1	2
27	Тема 3.5. Работа с протоколом HTTP.	1	1	2
28	Тема 3.6. Решение задач на API Яндекс.Карт.		4	4
29	Тема 3.7. Самостоятельная работа на http, json и командную строку.		2	2
30	Тема 3.8. Введение во flask. Обработка HTML-форм..	1	1	2
31	Тема 3.9. Шаблоны. Flask-wtf.	1	3	4
32	Тема 3.10. Знакомство с SQLAlchemy. Flask-sqlalchemy.	2	2	4

33	Тема 3.11. REST-API. Понятие. Flask-restfull.	2	6	8
34	Тема 3.12. Разворачиваем проект в облаке.	1	1	2
35	Тема 3.12. Цели и подходы к тестированию. Библиотеки unittest и pytest.	2	2	4
36	Тема 3.13. Чат-боты. Вконтакте, Telegram, Discrod.	3	3	6
37	Тема 3.14. Разработка навыка для Алисы.	2	4	6
38	Тема 3.15. Итоговая контрольная работа.		2	2
39	Тема 3.16. Работа над проектами в команде.	1	9	10
40	Тема 3.17. Защита проектов по модулю.		2	2
	Итого	45	99	144

СОДЕРЖАНИЕ ПРОГРАММЫ

«ПРОГРАММИРОВАНИЕ НА PYTHON»

РАЗДЕЛ «ОСНОВЫ ПРОГРАММИРОВАНИЯ НА ЯЗЫКЕ PYTHON»

Модуль 1. Введение в программирование на Python.

Знакомство с виртуальной средой взаимодействия: регистрация, организация личного кабинета, поиск и выкладывание материалов. Знакомство с системой автоматизированной проверки задач.

Основные понятия программирования: исполнитель, система команд, алгоритм, программа, среда разработки, интерпретатор, код программы и редактор кода.

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.

На этом этапе обучающиеся разрабатывают первые алгоритмы и программы, а также анализируют, на какие функциональные блоки может быть разбита программа, и определяют работоспособность разработанной программы.

В течение модуля ученики начинают решать задачи: от самых простых до продвинутых.

Учащиеся должны знать:

– базовые конструкции языка программирования Python (операции присваивания, ветвления, цикла, ввод\вывод, запись констант и выражений);

– основные методы ветвления программы - условные операторы;

– логические типы данных, прерывания, переходы (сброс циклов на определённой итерации, пропуск итерации при выполнении условий).

Учащиеся должны уметь:

– устанавливать необходимый инструментарий для разработки приложений;

- составлять программный код с использованием разных типов данных и операций над ними;
- составлять, отлаживать и исполнять на персональном компьютере программы решения учебных задач по программированию;
- применять условные конструкции;
- решать задачи с использованием разных видов циклов;
- искать информацию в свободных источниках и структурировать ее;
- объективно оценивать результаты своей работы.

Формы занятий, используемые при изучении данного модуля:

- лекционная,
- групповая работа,
- групповые консультации,
- самостоятельная работа,
- контрольная работа.

Тема 1.1. Знакомство со средой.

Теория. Знакомство с теми техническими средствами, которые понадобятся детям для обучения.

Практика. Ознакомительная работа с системой.

Тема 1.2. Условный оператор.

Теория. Изучаем оператор ветвления (if), реализующий выполнение определённых команд при условии, что некоторое логическое выражение истинно.

Практика. Разбор примеров программ с использованием ветвления . Вычисление логических выражений с помощью операции ветвления средствами языка python. Решение задач по теме.

Тема 1.3. Простые встроенные функции.

Теория. Знакомство с типами данных и функциями.

Практика. Разбор примеров создания и использования функций.

Тема 1.4. Знакомство с циклом while.

Теория. Оператор цикла с постусловием. Оператор с неизвестным

числом повторов.

Практика. Решение задач с использованием цикла while.

Тема 1.5. Отладчик.

Теория. Проверка программы на ошибки. Способы поиска ошибок и их исправлений.

Практика. Примеры задач, перестающие работать при неверных входных данных. Самостоятельный поиск возможностей ошибки.

Тема 1.6. Знакомство с циклом for.

Теория. Изучаем цикл с известным количеством повторений.

Практика. Решение задач с использованием цикла for.

Тема 1.7. Погружение в условия. True, False, Break, Continue.

Теория. Принцип работы операторов True, False, Break, Continue в циклах.

Практика. Разбор примеров и решений усложнённых задач с использованием операторов True, False, Break, Continue.

Тема 1.8. Вложенные циклы.

Теория. Использование циклов внутри тела других циклов. Внутренние и внешние циклы.

Практика. Решение задач по теме «Вложенные циклы»

Форма подведения итогов: самостоятельная работа по теме модуля.

Модуль 2. Множества, строки, списки и словари в Python.

Понятие о языке высокого уровня Python. Структура программы, строковые переменные и константы, работа с массивами, конкатенация в Python. Основные управляющие конструкции алгоритмов с ветвлением в Python. Принципы методов split и join. Примеры работы с ними.

Работа со списками, строками, множествами и кортежами в Python. Понятие словаря. Особенности работы со словарями в python. Решение задач.

Учащиеся должны знать:

– базовые конструкции языка программирования Python (операции

присваивания, ветвления, цикла, ввод\вывод, запись констант и выражений);

- основные методы сортировки списков;

- алгоритмы обработки списков (поиск элемента с заданными свойствами, удаление элементов, вставка элементов, замена элементов);

- способы организации процедур и функций.

Учащиеся должны уметь:

- записывать конструкции языка программирования Python;

- тестировать и осуществлять отладку программ;

- объективно оценивать результаты своей работы.

Формы занятий, используемые при изучении данного модуля:

- лекционная,

- групповая работа,

- групповые консультации,

- самостоятельная работа,

- контрольная работа.

Тема 2.1. Множества.

Теория. Структура множества. Неповторяющиеся элементы в случайном порядке.

Практика. Решение задач с использованием множеств.

Тема 2.2. Строки. Индексация.

Теория. Определение понятия строки. Посимвольная нумерация.

Практика. Решение задач по теме «Строки. Индексация.»

Тема 2.3. Строки. Срезы.

Теория. Понятие срезов строки. обращение к элементам по индексу.

Практика. Решение задач по теме «Строки. Срезы.»

Тема 2.4. Знакомство со списками.

Теория. Элементы списка. Создание и обращение к спискам.

Практика. Решение задач с использованием списков.

Тема 2.5. Кортежи. Преобразование коллекций.

Теория. Определение кортежа. Отличия от списков. Хранение

элементов различных типов

Практика. Решение задач на тему «Кортежи. Преобразование коллекций».

Тема 2.6. Методы Split и join. Списочные выражения.

Теория. Работа со списками. Применение методов split и join.

Практика. Решение задач на тему «Методы split и join».

Тема 2.7. Методы списков и строк.

Теория. Компьютерная графика и её виды. Кроссплатформенная библиотека для разработки графического интерфейса TkInter.

Теория. Генераторы списков. Методы find и rfind. Метод replace и count.

Практика. Решение задач на методы списков и строк.

Тема 2.8. Вложенные списки.

Теория. Обработка и вывод вложенных списков.

Практика. Решение задач на тему «Вложенные списки».

Тема 2.9. Знакомство со словарями.

Теория. Словари (dict) и работа с ними. Методы словарей.

Практика. Решение задач на тему «Знакомство со словарями».

Форма подведения итогов: самостоятельная работа по теме модуля.

Модуль 3. Функции в Python.

Понятие функции. Возвращаемые значения. Области видимости переменных. Определение объекта, принимающего аргументы и возвращающего значения.

Самостоятельная работа: работа с online-методичкой, в которой описываются способы решения задач, разработка решений по заданиям.

Учащиеся должны знать:

- основные понятия объектно-ориентированного проектирования;
- понятия ассоциативного массива, репозитория;
- модульный принцип компоновки программ;
- алгоритмы решения основных видов прикладных задач.

Учащиеся должны уметь:

- генерировать идеи;
- создавать и использовать стандартные функции языка Python;
- работать с документацией в интернете;
- объективно оценивать результаты своей работы.

Формы занятий, используемые при изучении данного модуля:

- лекционная,
- групповая работа,
- групповые консультации,
- самостоятельные работы,
- контрольная работа.

Тема 3.1. Функции.

Теория. Понятие функции. Способы определения функций.

Практика. Решение задач по теме урока.

Тема 3.2. Функции. Возвращение значений из функций.

Теория. Оператор Return. Ограничения на возвращаемые значения.

Практика. Решение задач по теме урока.

Тема 3.3. Области видимости переменных.

Теория. Локальная, глобальная и нелокальная области видимости

Практика. Решение задач по теме урока.

Тема 3.4. Функции: передача параметров.

Теория. Ключевые аргументы, область видимости аргументов.

Практика. Решение задач по теме урока.

Тема 3.5. Функции с переменным числом аргументов.

Теория. Изменяемое количество передаваемых аргументов при вызове функции.

Практика. Решение задач по теме урока.

Тема 3.6. Функции как объект. Лямбда-функции.

Теория. Создание и вызов функции в одной строке кода. Анонимные функции.

Практика. Решение задач по теме урока.

Тема 3.7. Обработка коллекций. Поточковый ввод `sys.stdin`.

Теория. Взаимодействие с интерпретатором python напрямую. Хранение набора значений одного или различных типов.

Практика. Решение задач по теме урока.

Форма подведения итогов: самостоятельная работа по теме модуля.

Модуль 4. Библиотеки Python.

Модули в Python. Подключение и использование модулей стандартной библиотеки. Модульный принцип компоновки программы. Работа с документацией в стандартной библиотеке. Понятие репозитория различных пакетов Python. Работа с внешними библиотеками Python и утилитой `pip`. Решение задач.

Учащиеся должны знать:

- стандартные подключаемые библиотеки и модули Python;
- модульный принцип компоновки программ;
- способы поиска и добавления библиотек python, не входящих в стандартный набор.

Учащиеся должны уметь:

- подключать и использовать модули стандартной библиотеки;
- работать с документацией в стандартной библиотеке;
- работать с внешними библиотеками и утилитой `pip`;
- объективно оценивать результаты своей работы.

Формы занятий, используемые при изучении данного модуля:

- лекционная,
- групповая работа,
- групповые консультации,
- самостоятельная работа.

Тема 4.1. `Random`.

Теория. Знакомство с модулем для генерации случайного набора

элементов. Управление случайной генерацией. Ограничение выборки и типов элементов.

Практика. Решение задач по теме урока.

Тема 4.2. Графика.

Теория. Визуализация математических вычислений. Создание анимированных изображений.

Практика. Решение задач по теме урока.

Тема 4.3. Графика + звук.

Теория. Построение спектрограммы. Визуализация звуковых волн.

Практика. Решение задач по теме урока.

Тема 4.4. Морфология.

Теория. Морфологический анализ языка. Определение характеристик слова на основе того, как это слово пишется.

Практика. Решение задач по теме урока.

Тема 4.5. Документы.

Теория. Открытие, закрытие, чтение и изменение документов.

Практика. Решение задач по теме урока.

Форма подведения итогов: самостоятельная работа по теме модуля.

Модуль 5. Объектно-ориентированное программирование.

Структурирование данных в виде объектов определённого типа. Формализация основных аспектов задачи для создания классов. Инкапсуляция логики работы программы внутрь объекта вместе с относящейся к нему информацией. Наследование логики и информации от одного или нескольких объектов.

Решение задач.

Учащиеся должны знать:

- основные понятия объектно-ориентированного проектирования;
- динамические, статические и классовые методы;
- способы переопределения класса его дочерними классами.

Учащиеся должны уметь:

- создавать экземпляры классов;
- определять атрибуты и методы внутри класса;
- вызывать определённый метод у объекта;
- учитывать предыдущие ошибки при работе.

Формы занятий, используемые при изучении данного модуля:

- лекционная,
- групповая работа,
- групповые консультации,
- самостоятельная работа.

Тема 5.1. Введение в ООП.

Теория. Понятия классов, объектов, методов. Преимущества использования ООП.

Практика. Решение задач по теме урока.

Тема 5.2. Полиморфизм.

Теория. Способность использовать в ООП общий интерфейс для нескольких форм (типов данных).

Практика. Решение задач по теме урока.

Тема 5.3. Определение операторов.

Теория. Реализации полиморфизма, когда можно задать отдельную реализацию какого-либо метода в своём классе.

Практика. Решение задач по теме урока.

Тема 5.4. Наследование.

Теория. Использование методов и свойств уже существующего объекта при создании нового.

Практика. Решение задач по теме модуля.

Модуль 6. Практическое применение языка Python.

Решение повседневных задач средствами Python. Автоматизация рутинных процессов. Разбор примеров олимпиадных задач и решений к ним.

Проектная работа: работа с конспектом, который описывает способы решения задач, поиск информации в интернете, самостоятельная работа над решением задач.

Тематика проектной работы:

- задачи, в которых по заданному алгоритму необходимо написать программу;

- задачи, в которых необходимо составить алгоритм решения и написать по нему программу;

- задачи, для решения которых необходимо найти некоторую информацию в Интернете.

Выполненная проектная работа презентуется перед одноклассниками.

Учащиеся должны знать:

- способы создания графического интерфейса программы;
- основные математические методы анализа данных;
- алгоритмы решения прикладных задач программными методами.

Учащиеся должны уметь:

- создавать программы с поточной обработкой данных;
- работать с функциями, вызывающими сами себя;
- искать необходимые для решения задач библиотеки и модули в интернете;
- адекватно воспринимать критику со стороны более опытных разработчиков.

Формы занятий, используемые при изучении данного модуля:

- лекционная,
- групповая работа,
- групповые консультации,
- итоговое выступление.

Тема 6.1. Основы ведения проектной деятельности и «Тайм-менеджмент».

Теория. Что такое проект. Виды проектов. Этапы проектной

деятельности. Теория управления временем.

Практика. Решение тематических задач.

Тема 6.2. Постановка проблемной ситуации и поиск путей решения.

Деление на команды.

Теория. Учимся искать пути решения, аргументировать свою точку зрения.

Практика. Мозговой штурм, решение командных задач.

Тема 6.3. Работа команд над реализацией своего проекта.

Теория. Консультация по теме задач проекта.

Практика. Решение задач по теме проекта.

Тема 6.4. Работа над возможными ошибками и недочётами в проекте.

Теория. Методы отладки и оптимизации кода.

Практика. Исправление неточностей и доработка проекта.

Тема 6.5. Подытожим и суммируем полученные навыки и знания по результатам работы над проектом.

Теория. Подводим итоги. Рефлексия.

Практика. Обсуждение полученных результатов. Внесение финальных изменений в проект.

Тема 6.6. Готовим презентацию для итоговой защиты.

Теория. Принципы хорошей презентации. Тренировка выступления.

Практика. Готовим проект и презентацию по полученным результатам.

Тема 6.7. Итоговая защита.

Теория. Формируем навыки публичного выступления. Умение усваивать информацию из выступления сотоварищей.

Практика. Представляем результат своей работы и полученных навыков. Анализируем полученный результат других выступающих.

ВТОРОЙ ГОД

РАЗДЕЛ «ОСНОВЫ ПРОМЫШЛЕННОГО ПРОГРАММИРОВАНИЯ»

Модуль 1. Разработка GUI приложений.

На этом этапе обучающиеся повторяют материал за прошлый учебный год, путем решения задач. В течение модуля ученики изучают процесс создания ПО с пользовательским интерфейсом, познакомятся с базами данных, научатся работать с файлами разных форматов. Также учащиеся знакомятся с основами работы в системе контроля версий GIT.

В течение модуля ученики решают большое количество задач различной сложности и пишут 2 самостоятельные работы, а также защищают проект, разработанный индивидуально.

Учащиеся должны знать:

- основные понятия концепции GUI;
- основные понятия концепции при работе с файловой системой и файлами;
- основные понятия концепции работы с базами данных;
- основы работы в системе контроля версий GIT;

Учащиеся должны уметь:

- составлять ПО с GUI;
- работать с различными файлами;
- работать с базой данных;
- решать тренировочные задачи;
- искать информацию в свободных источниках и структурировать ее;
- объективно оценивать результаты своей работы.

Формы занятий, используемые при изучении данного модуля:

- лекционная,
- групповая работа,
- групповые консультации,
- самостоятельная работа,

– защита проектов.

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

Практика. Решение задач на основные конструкции и структуры данных. Решение задач на классы и библиотеки.

Тема 1.2. Основные команды при одиночной работе с GIT.

Теория. Принципы работы с системой контроля версий Git из командной строки.

Практика. Решение задач на тему «Основные команды при одиночной работе с GIT».

Тема 1.3. Что такое QT и PyQT. QtDesigner, pyuic, два способа подключения uic-файла.

Теория. Основы разработки ПО с GUI в Python с использованием библиотеки PyQT.

Практика. Решение задач на тему «Что такое QT и PyQT. QtDesigner, pyuic, два способа подключения uic-файла».

Тема 1.4. Обработка исключений. Создание собственных исключений.

Теория. Основные концепции работы с исключениями в коде, создание обработки собственных исключений.

Практика. Решение задач на тему «Обработка исключений. Создание собственных исключений».

Тема 1.5. Файлы в Python. Работа с различными типами файлов. Использование библиотек.

Теория. Файловая структура, форматы файлов, работа с файлами в Python.

Практика. Решение задач на тему «Файлы в Python. Работа с различными типами файлов. Использование библиотек».

Тема 1.6. Диалоги, работа с изображениями.

Теория. Создание диалоговых окон в ПО. Работа с изображениями в GUI.

Практика. Решение задач на тему «Диалоги, работа с изображениями».

Тема 1.7. Самостоятельная работа на файлы.

Практика. Решение самостоятельной работы на тему «Файлы в Python. Работа с различными типами файлов. Использование библиотек».

Тема 1.8. Работа с простыми таблицами (csv). Фильтрация, сортировка, вывод. Работа с табличными данными в PyQT.

Теория. Работа с файлом типа csv, табличное представление данных из файла в GUI.

Практика. Решение задач на тему «Работа с простыми таблицами (csv). Фильтрация, сортировка, вывод. Работа с табличными данными в PyQT».

Тема 1.9. Введение в БД, работа с SQL-таблицами и отображение данных в PyQT.

Теория. Работа с базой данных sqlite. Отображение данных из базы в GUI.

Практика. Решение задач на тему «Введение в БД, работа с SQL-таблицами и отображение данных в PyQT».

Тема 1.10. Установка дополнительных компонентов. PyQTGraph.

Теория. Дополнительные компоненты для PyQT. Использование PyQTGraph.

Практика. Решение задач на тему «Установка дополнительных компонентов. PyQTGraph».

Тема 1.11. Обработка клавиатуры и курсора. Сборка независимого приложения.

Теория. Обработка событий ввода информации, создание независимого приложения.

Практика. Решение задач на тему «Обработка клавиатуры и курсора. Сборка независимого приложения».

Тема 1.11. Самостоятельная работа на SQL.

Практика. Решение самостоятельной работы по теме «SQL».

Тема 1.12. Работа над проектами.

Практика. Разработка индивидуальных проектов по теме модуля.

Тема 1.13. Защита проектов по модулю.

Практика. Подготовка к защите проекта. Защита проектов по модулю.

Форма подведения итогов: защита проектов, рефлексия работы над проектами.

Модуль 2. Pygame, создание игр на Python

В этом модуле ученики разбирают основы разработки игр на основе библиотеки PyGame. Также в теме продолжается разбор работы с использованием GIT, но уже в командной разработке.

В течение модуля школьники по традиции решают большое количество задач и создают командный проект.

Учащиеся должны знать:

- основные понятия игростроения;
- основные понятия при работе с графикой;
- библиотеку PyGame.

Учащиеся должны уметь:

- работать в команде;
- создавать игры;
- тестировать и осуществлять отладку программ;
- объективно оценивать результаты своей работы.

Формы занятий, используемые при изучении данного модуля:

- лекционная,
- групповая работа,
- групповые консультации,
- защита проектов.

Тема 2.1. Введение в репозитории. Подключение в PyCharm. Работа с удаленным репозиторием.

Теория. Работа с репозиторием из PyCharm, создание и подключение удаленных репозиторияев.

Практика. Решение задач на тему «Введение в репозитории. Подключение в PyCharm. Работа с удаленным репозиторием».

Тема 2.2. Введение в Pygame. Игровой цикл. События.

Теория. Основы разработки игр. Знакомство с библиотекой PyGame.

Практика. Решение задач на тему «Введение в Pygame. Игровой цикл.

События».

Тема 2.3. Клетчатое поле.

Теория. Основные концепции при разработке игра на клетчатом поле.

Практика. Решение задач на тему «Клетчатое поле».

Тема 2.4. Совместная работа над проектом, основные понятия. Работа с репозиториями в среде разработки.

Теория. Основные концепции при работе в команде, в удаленном репозитории командой.

Практика. Решение задач на тему «Совместная работа над проектом, основные понятия. Работа с репозиториями в среде разработки».

Тема 2.5. Изображения. Спрайты. Столкновения и другие взаимодействия.

Теория. Работа с графикой в PyGame.

Практика. Решение задач на тему «Изображения. Спрайты.

Столкновения и другие взаимодействия».

Тема 2.6. Разработка и украшение игры.

Теория. Правильный процесс разработки игр.

Практика. Решение задач на тему «Разработка и украшение игры».

Тема 2.7. Работа над проектами в команде.

Практика. Работа над проектами в команде.

Тема 2.8. Защита проектов по модулю.

Практика. Подготовка и защита проектов по модулю.

Форма подведения итогов: защита проектов, рефлексия итогов защиты.

Модуль 3. Web-разработка

В данном модули ученики знакомятся с основами веб-разработки, взаимодействия с сервером по API, созданием чат-ботов и навыка для

“Алисы”. Также ученики познакомятся с понятием периодических задач и аргументами командной строки. Ученики решают большое кол-во задач.

Учащиеся должны знать:

- основные понятия HTTP;
- основные этапы создания сайтов;
- основные этапы создания чат-ботов;
- основные этапы создания навыка для “Алисы”;
- основные этапы создания REST API.

Учащиеся должны уметь:

- разрабатывать сайты;
- разрабатывать REST API;
- разрабатывать чат-ботов;
- разрабатывать навыки для “Алисы”
- объективно оценивать результаты своей работы.

Формы занятий, используемые при изучении данного модуля:

- лекционная,
- групповая работа,
- групповые консультации,
- защита проектов.

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

Теория. Основы работы с популярными форматами файлов в веб-разработки - json, zip.

Практика. Решение задач на тему «Работа с популярными форматами файлов (json, zip)».

Тема 3.2. Знакомство с API.

Теория. Основы взаимодействия с внешними сервисами посредством API.

Практика. Решение задач на тему «Знакомство с API».

Тема 3.3. Работа с командной строкой (скрипты, аргументы).
Периодические задачи (модуль schedule).

Теория. Основы работы с аргументами программы, периодические задачи.

Практика. Решение задач на тему «Работа с командной строкой (скрипты, аргументы). Периодические задачи (модуль schedule)».

Тема 3.4. Библиотека argparse. Задачи на создание скриптов с ее помощью.

Теория. Разработка скриптов с использованием аргументов командной строки.

Практика. Решение задач на тему «Библиотека argparse. Задачи на создание скриптов с ее помощью».

Тема 3.5. Работа с протоколом HTTP.

Теория. Работа с протоколом HTTP.

Практика. Решение задач на тему «Работа с протоколом HTTP».

Тема 3.5. Работа с протоколом HTTP.

Теория. Углубляемся в работу с протоколом HTTP.

Практика. Решение задач на тему «Работа с протоколом HTTP».

Тема 3.6. Решение задач на API Яндекс.Карт.

Практика. Решение задач на тему «API Яндекс.Карт».

Тема 3.7. Самостоятельная работа на http, json и командную строку.

Практика. Решение самостоятельной работы.

Тема 3.8. Введение во flask. Обработка HTML-форм.

Теория. Введение в разработку сайтов с использованием микрофреймворка Flask.

Практика. Решение задач на тему «Введение во flask. Обработка HTML-форм».

Тема 3.9. Шаблоны. Flask-wtf.

Теория. Шаблонизаторы в Flask. Обработка форм с использованием модуля Flask-wtf.

Практика. Решение задач на тему «Шаблоны. Flask-wtf».

Тема 3.10. Знакомство с SQLAlchemy. Flask-sqlalchemy.

Теория. ORM. Работа с базой данных через абстракцию в коде.

Практика. Решение задач на тему «Знакомство с SQLAlchemy. Flask-sqlalchemy».

Тема 3.11. REST-API. Понятие. Flask-restfull.

Теория. Основы разработки собственных REST-API.

Практика. Решение задач на тему «REST-API. Понятие. Flask-restfull».

Тема 3.12. Разворачиваем проект в облаке.

Теория. Основные аспекты размещения проекта в облаке.

Практика. Решение задач на тему «Разворачиваем проект в облаке».

Тема 3.13. Цели и подходы к тестированию. Библиотеки unittest и pytest.

Теория. Основы тестирования кода.

Практика. Решение задач на тему «Цели и подходы к тестированию. Библиотеки unittest и pytest».

Тема 3.14. Чат-боты. Вконтакте, Telegram, Discrod.

Теория. Основные концепции разработки чат-ботов.

Практика. Решение задач на тему «Цели и подходы к тестированию. Библиотеки unittest и pytest».

Тема 3.14. Разработка навыка для “Алисы”.

Теория. Основные концепции разработки навыков для “Алисы”.

Практика. Решение задач на тему «Разработка навыка для “Алисы”».

Тема 3.15. Итоговая контрольная работа.

Практика. Решение задач по темам 3х модулей.

Тема 3.16. Работа над проектами в команде.

Практика. Работа над проектами в команде.

Тема 3.17. Защита проектов по модулю. Рефлексия.

Практика. Защита проектов по модулю. Рефлексия.

Форма подведения итогов. Публичная защита проекта.

**МЕТОДИЧЕСКОЕ ОБЕСПЕЧЕНИЕ
ПРОГРАММЫ «ПРОГРАММИРОВАНИЕ НА PYTHON»**

Тема модуля	Форма занятий	Приёмы и методы организации образовательного процесса	Дидактический материал. Электронные источники	Техническое оснащение и расходный материал	Форма подведения итогов
Раздел 1. «Основы программирования на языке Python»					
Модуль 1. Введение в программирование на Python	Комбинированная	Объяснительно-иллюстративный. Метод упражнений. Метод практических работ. Метод мозгового штурма. Проблемно-поисковый.	1. Яндекс.Контест, сервис, позволяющий программистам со всего мира соревноваться на предложенных задачах. Сервис способен одновременно обрабатывать терабайты данных, поэтому легко выдержит нагрузку в более чем тысячу участников. (https://contest.yandex.ru); 2. LMS AnyTask – система управления обучением, позволяющая автоматизировать большинство внутренних процессов Яндекс.Лицея (http://anytask.org); 3. Сервис PythonTutor, позволяющий визуализировать исполнение кода на языке Python (http://pythontutor.com); 4. Федеральный портал Единая коллекция образовательных ресурсов (http://school-collection.edu.ru); 6. Официальная документация языка Python (http://docs.python.org).	<ul style="list-style-type: none"> • Ноутбуки с мышкой и доступом к сети Интернет. • Презентационное оборудование. • Принтер. • Маркерная доска. 	Контрольная работа.

<p>Модуль 2. Множества, строки, списки и словари в Python.</p>	<p>Комбини рованная</p>	<p>Объяснительно- иллюстративный. Метод упражнений. Метод практических работ. Метод мозгового штурма. Проблемно-поисковый.</p>	<p>1. Яндекс.Контест, сервис, позволяющий программистам со всего мира соревноваться на предложенных задачах. Сервис способен одновременно обрабатывать терабайты данных, поэтому легко выдержит нагрузку в более чем тысячу участников. (https://contest.yandex.ru); 2. LMS AnyTask – система управления обучением, позволяющая автоматизировать большинство внутренних процессов Яндекс.Лицея (http://anytask.org); 3. Сервис PythonTutor, позволяющий визуализировать исполнение кода на языке Python (http://pythontutor.com); 4. Федеральный портал Единая коллекция образовательных ресурсов (http://school-collection.edu.ru); 6. Официальная документация языка Python (http://docs.python.org).</p>	<ul style="list-style-type: none"> • Ноутбуки с мышкой и доступом к сети Интернет. • Презентационное оборудование. • Принтер. • Маркерная доска. 	<p>Контрольная работа.</p>
--	-----------------------------	--	--	--	----------------------------

<p>Модуль 3. Функции в Python.</p>	<p>Комбинированная</p>	<p>Объяснительно-иллюстративный. Метод упражнений. Метод практических работ. Метод мозгового штурма. Проблемно-поисковый.</p>	<p>1. Яндекс.Контест, сервис, позволяющий программистам со всего мира соревноваться на предложенных задачах. Сервис способен одновременно обрабатывать терабайты данных, поэтому легко выдержит нагрузку в более чем тысячу участников. (https://contest.yandex.ru);</p> <p>2. LMS AnyTask – система управления обучением, позволяющая автоматизировать большинство внутренних процессов Яндекс.Лицея (http://anytask.org);</p> <p>3. Сервис PythonTutor, позволяющий визуализировать исполнение кода на языке Python (http://pythontutor.com);</p> <p>4. Федеральный портал Единая коллекция образовательных ресурсов (http://school-collection.edu.ru);</p> <p>6. Официальная документация языка Python (http://docs.python.org).</p>	<ul style="list-style-type: none"> • Ноутбуки с мышкой и доступом к сети Интернет. • Презентационное оборудование. • Принтер. • Маркерная доска. 	<p>Контрольная работа.</p>
--	------------------------	---	--	--	----------------------------

Модуль 4. Библиотеки Python.	Комбинированная	Объяснительно-иллюстративный. Метод упражнений. Метод практических работ. Метод мозгового штурма. Проблемно-поисковый.	<p>1. Яндекс.Контест, сервис, позволяющий программистам со всего мира соревноваться на предложенных задачах. Сервис способен одновременно обрабатывать терабайты данных, поэтому легко выдержит нагрузку в более чем тысячу участников. (https://contest.yandex.ru);</p> <p>2. LMS AnyTask – система управления обучением, позволяющая автоматизировать большинство внутренних процессов Яндекс.Лицея (http://anytask.org);</p> <p>3. Сервис PythonTutor, позволяющий визуализировать исполнение кода на языке Python (http://pythontutor.com);</p> <p>4. Федеральный портал Единая коллекция образовательных ресурсов (http://school-collection.edu.ru);</p> <p>6. Официальная документация языка Python (http://docs.python.org).</p>	<ul style="list-style-type: none"> • Ноутбуки с мышкой и доступом к сети Интернет. • Презентационное оборудование. • Принтер. • Маркерная доска. 	Контрольная работа.
Модуль 5. Объектно-ориентированное программирование.	Комбинированная	Объяснительно-иллюстративный. Метод упражнений. Метод практических работ. Метод мозгового штурма. Проблемно-поисковый.	<p>2. LMS AnyTask – система управления обучением, позволяющая автоматизировать большинство внутренних процессов Яндекс.Лицея (http://anytask.org);</p>	<ul style="list-style-type: none"> • Ноутбуки с мышкой и доступом к сети Интернет. • Презентационное оборудование. • Принтер. • Маркерная доска. 	Контрольная работа.

Модуль 6. Практическое применение языка Python.	Комбинированная	Объяснительно-иллюстративный. Метод мозгового штурма. Проблемно-поисковый.	3. Сервис PythonTutor, позволяющий визуализировать исполнение кода на языке Python (http://pythontutor.com);	<ul style="list-style-type: none"> • Ноутбуки с мышкой и доступом к сети Интернет. • Презентационное оборудование. • Принтер. • Маркерная доска. 	Контрольная работа.
Раздел 2. «Основы промышленного программирования»					
Модуль 1. Разработка GUI приложений	Комбинированная	Метод проектов. Объяснительно-иллюстративный. Метод мозгового штурма. Проблемно-поисковый.	<p>1. Яндекс.Контест, сервис, позволяющий программистам со всего мира соревноваться на предложенных задачах. Сервис способен одновременно обрабатывать терабайты данных, поэтому легко выдержит нагрузку в более чем тысячу участников. (https://contest.yandex.ru);</p> <p>2. LMS AnyTask – система управления обучением, позволяющая автоматизировать большинство внутренних процессов Яндекс.Лицея (http://anytask.org);</p> <p>3. Сервис PythonTutor, позволяющий визуализировать исполнение кода на языке Python (http://pythontutor.com);</p> <p>4. Федеральный портал Единая коллекция образовательных ресурсов (http://school-collection.edu.ru);</p> <p>6. Официальная документация языка Python (http://docs.python.org).</p>	Ноутбуки с мышкой и доступом к сети Интернет, на которых установлено следующие программное обеспечение: операционная система Windows (версия не ниже 7); Android Studio; Eclipse. Презентационное оборудование. Мобильные устройства на базе ОС Android.	Самостоятельная работа.

<p>Модуль 2. Pygame, создание игр на Python</p>	<p>Комбини рованная</p>	<p>Метод проектов. Объяснительно- иллюстративный. Метод мозгового штурма. Проблемно-поисковый.</p>	<p>1. Яндекс.Контест, сервис, позволяющий программистам со всего мира соревноваться на предложенных задачах. Сервис способен одновременно обрабатывать терабайты данных, поэтому легко выдержит нагрузку в более чем тысячу участников. (https://contest.yandex.ru); 2. LMS AnyTask – система управления обучением, позволяющая автоматизировать большинство внутренних процессов Яндекс.Лицея (http://anytask.org); 3. Сервис PythonTutor, позволяющий визуализировать исполнение кода на языке Python (http://pythontutor.com); 4. Федеральный портал Единая коллекция образовательных ресурсов (http://school-collection.edu.ru); 6. Официальная документация языка Python (http://docs.python.org).</p>	<p>Ноутбуки с мышкой и доступом к сети Интернет. Презентационное оборудование. Принтер. Маркерная доска.</p>	<p>Контрольная работа.</p>
---	-----------------------------	--	--	--	----------------------------

<p>Модуль 3. Web-разработка</p>	<p>Комбинированная</p>	<p>Метод проектов. Объяснительно-иллюстративный. Метод мозгового штурма. Проблемно-поисковый.</p>	<p>1. Яндекс.Контест, сервис, позволяющий программистам со всего мира соревноваться на предложенных задачах. Сервис способен одновременно обрабатывать терабайты данных, поэтому легко выдержит нагрузку в более чем тысячу участников. (https://contest.yandex.ru);</p> <p>2. LMS AnyTask – система управления обучением, позволяющая автоматизировать большинство внутренних процессов Яндекс.Лицея (http://anytask.org);</p> <p>3. Сервис PythonTutor, позволяющий визуализировать исполнение кода на языке Python (http://pythontutor.com);</p> <p>4. Федеральный портал Единая коллекция образовательных ресурсов (http://school-collection.edu.ru);</p> <p>6. Официальная документация языка Python (http://docs.python.org).</p>	<p>Ноутбуки с мышкой и доступом к сети Интернет. Презентационное оборудование. Принтер. Маркерная доска.</p>	<p>Защита проекта.</p>
-------------------------------------	------------------------	---	--	--	------------------------

СПИСОК ИСТОЧНИКОВ ИНФОРМАЦИИ

использованных при написании программы:

1. К. Ю. Поляков, Е. А. Еремин. Информатика. Углублённый уровень. Учебник для 10 класса в 2 частях. М.: БИНОМ. Лаборатория знаний, 2014.
2. М. Лутц. Изучаем Python. СПб.: Символ-Плюс, 2011.
3. Задачи по программированию. Под ред. С. М. Окулова, М.: БИНОМ. Лаборатория знаний, 2006.
4. С. М. Окулов. Основы программирования. М.: Бином. Лаборатория знаний, 2012.

рекомендованных обучающимся:

1. М. Лутц. Изучаем Python. СПб.: Символ-Плюс, 2011.
2. Информатика и ИКТ. Задачник-практикум в 2 частях. Под ред. И. Г. Семакина и Е. К. Хеннера. М.: БИНОМ. Лаборатория знаний, 2014.
3. Материалы и презентации к урокам в LMS Яндекс.Лицея.
4. Сайт pythonworld.ru — «Python 3 для начинающих».
5. Сайт pythontutor.ru — «Питонтьютор».
6. <https://www.youtube.com/playlist?list=PLJOzdkh8T5kpIBTG9mM2wVBjh-5OpdwBl> — Лекции А.В. Умнова, прочитанные в Школе Анализа Данных Яндекса.