

**МУНИЦИПАЛЬНОЕ БЮДЖЕТНОЕ УЧРЕЖДЕНИЕ  
ДОПОЛНИТЕЛЬНОГО ОБРАЗОВАНИЯ  
ГОРОДА РОСТОВА-НА-ДОНУ  
«ДВОРЕЦ ТВОРЧЕСТВА ДЕТЕЙ И МОЛОДЕЖИ»**

**ЦЕНТР ЦИФРОВОГО ОБРАЗОВАНИЯ ДЕТЕЙ «IT-куб»**

Принято  
педагогическим советом МБУ ДО ДТДМ  
Протокол №1 от 31.08.2023 г.  
Одобрено  
методическим советом МБУ ДО ДТДМ  
Протокол № 11 от 30.08.2023 г.

Утверждаю  
Директор МБУ ДО ДТДМ  
\_\_\_\_\_ Е.Э. Жихарцева  
Приказ № 789 от 31.08. 2023 г.

**ДОПОЛНИТЕЛЬНАЯ ОБЩЕОБРАЗОВАТЕЛЬНАЯ ПРОГРАММА**

**«Основы программирования на Python (углубленный уровень)»**

Возрастная категория: 14-16 лет.  
Срок реализации: 1 год.

Разработчик программы:  
**Пусева О.Н.,**  
педагог дополнительного образования  
Программу реализуют:  
**Буланов Д.П., Пусева О.Н., Серчаев Г.Г.,**  
педагоги дополнительного образования.  
Методическое сопровождение:  
**Букатова Е.В.,** методист.

г. Ростов-на-Дону  
2023 г.

## ОГЛАВЛЕНИЕ

I.	ПОЯСНИТЕЛЬНАЯ ЗАПИСКА.....	3
II.	УЧЕБНО-ТЕМАТИЧЕСКИЙ ПЛАН.....	10
III.	СОДЕРЖАНИЕ ПРОГРАММЫ.....	12
IV.	МЕТОДИЧЕСКОЕ ОБЕСПЕЧЕНИЕ.....	18
V.	ДИАГНОСТИЧЕСКИЙ ИНСТРУМЕНТАРИЙ.....	21
VI.	СПИСОК ЛИТЕРАТУРЫ.....	24
VII.	ПРИЛОЖЕНИЯ.....	26

## **I. ПОЯСНИТЕЛЬНАЯ ЗАПИСКА**

Дополнительная общеобразовательная общеразвивающая программа «Основы программирования на языке Python (углубленный уровень)» **технической направленности** разработана на основе рекомендаций партнера Яндекса (АНО ДПО «ШАД») на факультете компьютерных наук НИУ ВШЭ. с учетом развития науки, техники, культуры, экономики, технологий и социальной сферы.

В настоящее время мы переживаем большие изменения в развитии общества. В современную жизнь человека всё больше внедряются компьютеры и информационные технологии. Всё большее значение приобретает умение человека грамотно обращаться с компьютером, причём 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.

### **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.

### **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.

**Новизна данной образовательной программы:**

- обеспечивает знакомство с фундаментальными понятиями алгоритмизации и программирования на доступном уровне;
- имеет практическую направленность с ориентацией на реальные потребности, соответствующие возрасту ученика;
- охватывает как алгоритмическое направление, так и вопросы практического использования полученных знаний при решении задач из различных областей знаний;
- ориентирована на существующий парк вычислительной техники и дополнительные ограничения;
- допускает возможность варьирования в зависимости от уровня подготовки и интеллектуального уровня учащихся (как группового, так и индивидуального);
- предусматривает возможность индивидуальной работы с учащимися.

Практическая значимость курса заключается в том, что он способствует более успешному овладению знаниями и умениями по направлению «Программирование» через развитие самостоятельности обучающихся и оптимизацию средств и методов обучения.

Элементы программы курса могут быть рекомендованы для использования учителями информатики при проведении лабораторно-практических и практических занятий.

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

**Направленность** – техническая.

**Тип программы** - общеразвивающая.

**Вид программы** – модифицированная. Её содержание разработано в соответствии с требованиями актуальных нормативно-правовых документов в образовании: Национальной доктрины образования в Российской Федерации, Концепции дополнительного образования детей, Приказа Министерства просвещения РФ от 9 ноября 2018 г. № 196 «Об утверждении Порядка организации и осуществления образовательной деятельности по дополнительным общеобразовательным программам», Концепции развития системы образования в Ростовской области на период до 2025 г. и др.

При разработке содержания данной программы использована дополнительная общеобразовательная программа: «Программирование на Python»/Степанов Д.И., Елисеева А.Г.

**Уровень освоения** – углубленный.

**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;
- формировать и развивать навыки публичного выступления.

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

### **Прогнозируемые результаты освоения программы обучающимися по уровням**

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

В результате освоения программы обучающийся должен приобрести следующие компетенции:

- знание основ современных языков программирования;
- умение объяснять и использовать на практике как простые, так и сложные структуры данных и конструкции для работы с ними;
- умение искать и обрабатывать ошибки в коде;
- умение разбивать решение задачи на подзадачи;
- способность писать грамотный, красивый код;
- способность анализировать как свой, так и чужой код;
- способность работать с информацией: находить, оценивать и использовать информацию из различных источников, необходимую для решения профессиональных задач (в том числе на основе системного подхода);
- способность грамотно строить коммуникацию, исходя из целей и ситуации.

#### **Личностные результаты:**

- сформированность ответственного отношения к учению, способности довести до конца начатое дело аналогично завершённым творческим учебным проектам;
- сформированность способности к саморазвитию и самообразованию средствами информационных технологий на основе приобретённой благодаря иллюстрированной среде программирования мотивации к обучению и познанию;
- Получен опыт участия в социально значимых проектах, повышение уровня самооценки благодаря реализованным проектам;
- сформированность коммуникативной компетентности в общении и сотрудничестве со сверстниками в процессе образовательной, учебно-исследовательской и проектной деятельности, участия в конкурсах и конференциях различного уровня;
- сформированность целостного мировоззрения, соответствующего современному уровню развития информационных технологий;
- сформированность осознанного позитивного отношения к другому человеку, его мнению, результату его деятельности;
- сформированность ценности здорового и безопасного образа жизни; усвоение правил индивидуального и коллективного безопасного поведения

при работе с компьютерной техникой.

**Метапредметные результаты:**

- умение самостоятельно ставить и формулировать для себя новые задачи, развивать мотивы своей познавательной деятельности;
- умение самостоятельно планировать пути решения поставленной проблемы для получения эффективного результата; понимание, что в программировании длинная программа не всегда лучшая;
- умение критически оценивать правильность решения учебно-исследовательской задачи;
- умение корректировать свои действия, вносить изменения в программу и отлаживать её в соответствии с изменяющимися условиями;
- владение основами самоконтроля, способность к принятию решений;
- умение создавать, применять и преобразовывать знаки и символы, модели и схемы для решения учебно-исследовательских и проектных работ;
- сформированность и развита компетентность в области использования информационно-коммуникационных технологий (ИКТ-компетенция);
- умение организовывать учебное сотрудничество и совместную деятельность с учителем и сверстниками в процессе проектной и учебно-исследовательской деятельности.

**Предметные результаты:**

- умение определять результат выполнения алгоритма при заданных исходных данных, узнавать изученные алгоритмы обработки чисел и числовых последовательностей, создавать на их основе несложные программы анализа данных, читать и понимать несложные программы, написанные на выбранном для изучения универсальном алгоритмическом языке высокого уровня;
- сформированность представлений об основных предметных понятиях («информация», «алгоритм», «исполнитель», «модель») и их свойствах;
- способность к логическому и алгоритмическому мышлению, умение составить и записать алгоритм для конкретного исполнителя, знакомство с основными алгоритмическими структурами — линейной, условной и циклической;
- умение выполнять пошагово (с использованием компьютера или вручную) несложные алгоритмы управления исполнителями и анализа числовых и текстовых данных;
- навыки и опыт разработки программ в выбранной среде программирования, включая тестирование и отладку программ; умение использовать основные управляющие конструкции объектно-ориентированного программирования и библиотеки прикладных программ, выполнять созданные программы;
- умение разрабатывать и использовать компьютерно-математические модели, оценивать числовые параметры моделируемых объектов и процессов,

интерпретировать результаты, получаемые в ходе моделирования реальных процессов, анализировать готовые модели на предмет соответствия реальному объекту или процессу;

- умение соблюдать нормы информационной этики и права.

#### **Объем и срок освоения программы:**

Содержание программы рассчитано на 160 академических часа, которые включают вводный модуль на 16 часов и основной модуль на 144 часа. Программа содержит вводный модуль и основной. Необходимость вводного модуля диктуется углубленным уровнем программы, требующим отбора обучающихся и доведение их начального уровня до требуемого для перехода к основному модулю. В вводный модуль включены базовые понятия, но которых строится преподавание основного модуля.

Срок освоения программы – 1 год.

#### **Режим занятий:**

2 раза в неделю по 2 часа, всего 160 часов.

Занятия проводятся в соответствии с нормами СанПиН 2.4.3648–20.

**Наполняемость группы:** 12-15 человек.

#### **Формы организации деятельности обучающихся**

*Формы организации учебных занятий:* основной тип занятий — комбинированный, сочетающий в себе элементы теории и практики. Большинство заданий курса выполняется самостоятельно с помощью персонального компьютера и необходимых программных средств.

Единицей учебного процесса является блок уроков (модуль). Каждый такой блок охватывает отдельную информационную технологию или её часть. Внутри блоков разбивка по времени изучения производится учителем самостоятельно, но с учётом рекомендованного календарно-тематического плана. С учётом регулярного повторения ранее изученных тем, темп изучения отдельных разделов блока определяется субъективными и объективными факторами.

*Формы проведения занятий:* Каждая тема курса начинается с постановки задачи — характеристики предметной области или конкретной программы на языке Python, которую предстоит изучить. С этой целью учитель проводит демонстрацию презентации или показывает саму программу, а также готовые работы, выполненные в ней. Закрепление знаний проводится с помощью практики отработки умений самостоятельно решать поставленные задачи, соответствующих минимальному уровню планируемых результатов обучения.

Основные задания являются обязательными для выполнения всеми обучающимися в классе. Задания выполняются на компьютере с использованием интегрированной среды разработки. При этом ученики не только формируют новые теоретические и практические знания, но и приобретают новые технологические навыки.

При организации занятий по курсу «Основы программирования на языке Python (углублённый уровень)» для достижения поставленных целей и решения поставленных 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.

**Адресат программы.** Данная программа рассчитана на возраст с 14 до 16 лет. Зачисляются учащиеся, прошедшие отбор в два этапа: первый – дистанционное онлайн тестирование, второй – очное/дистанционное собеседование с преподавателем.

**Краткое описание возрастных психофизиологических особенностей детей, которым адресовано содержание программы.**

В социальном плане подростковая фаза – продолжение первичной социализации. Подростковое *чувство взрослости* – главным образом новый уровень притязаний, предвосхищающий положение, которого подросток фактически еще не достиг. Отсюда – типичные возрастные конфликты и их преломление в самосознании подростка. В подростковом возрасте происходят существенные изменения в развитии мыслительной деятельности учащихся, главным образом в процессе обучения. Содержание и логика изучаемых предметов, характер усвоения знаний у подростков требуют опоры на способность самостоятельно мыслить, сравнивать, делать выводы и обобщения.

Растущая заинтересованность обучающихся позволяет успешно осваивать новые научные категории, оперировать информацией, решать задачи с использованием высокоточного инновационного оборудования и специализированных компьютерных программ. Подростки испытывают большое эмоциональное удовлетворение от исследовательской деятельности. Им нравится мыслить, делать самостоятельные открытия. Программирование помогает детям воплощать в жизнь свои задумки, строить планы и фантазировать

Все вышеперечисленные особенности возраста были учтены при разработке содержания программы и технологий её реализации.

## II. УЧЕБНО-ТЕМАТИЧЕСКИЙ ПЛАН

### Вводный модуль

№ п/п	Тема	Количество часов	Теория	Практика
1	Диагностическая работа	2		2
2	Введение в программирование. Основные понятия	2	2	
3	Теория множеств. Булевы операции	2	1	1
4	Теория множеств. Булевы операции	2	1	1
5	Теория множеств. Булевы операции	2		2
6	Системы счисления	2	1	1
7	Системы счисления	2		2
8	Высказывания. Элементы логики	2	1	1
<b>Итого</b>		<b>16</b>	<b>6</b>	<b>10</b>

### Основной модуль

Модуль программы	Количество часов		
	Теория	Практика	Всего
<b>Модуль 1. Введение в программирование.</b>	<b>14</b>	<b>16</b>	<b>30</b>
Тема 1.1. Знакомство с системой Яндекс. Контекст. Понятия кода, интерпретатора, программы. Интегрированные среды, исполнение кода. Простейшие программы с выводом на экран.	2	1	3
Тема 1.2. Условный оператор. Переменные и арифметика.	2	1	3
Тема 1.3. Знакомство с циклом while.	2	2	4
Тема 1.4. Отладчик.	2	2	4
Тема 1.5. Знакомство с циклом for.	2	2	4
Тема 1.6. Погружение в условия. True, False, Break, Continue.	2	2	4
Тема 1.7. Знакомство со списками.	2	2	4
Тема 1.8. Решение задач по теме модуля 1.		4	4
<b>Модуль 2. Базовые конструкции в Python.</b>	<b>20</b>	<b>22</b>	<b>42</b>
Тема 2.1. Дополнительные возможности цикла for. Кортежи. Сортировки.	2	2	4
Тема 2.2. Знакомство со срезами и диапазонами. Равенство и совпадение объектов.	2	2	4
Тема 2.3. Списочные выражения. Методы split и join.	2	2	4
Тема 2.4. Другие методы списков и строк.	2	2	4
Тема 2.5. Функции.	4	2	6

Тема 2.6. Области видимости переменных.	4	2	6
Тема 2.7. Контрольная работа по темам 1 и 2 модулей.		2	2
Тема 2.8. Разбор контрольной работы.		2	2
Тема 2.9. Введение в компьютерную графику. Модуль TKInter.	4	2	6
Тема 2.10. Решение задач по теме модуля.		4	4
<b>Модуль 3. Решение прикладных задач в Python.</b>	<b>35</b>	<b>37</b>	<b>72</b>
Тема 3.1. Повторение материала 1 и 2 модулей.	2	2	4
Тема 3.2. Функции. Детальное изучение. Функциональная парадигма программирования.	10	6	16
Тема 3.3. Словари и множества.	8	4	12
Тема 3.4. Обзор стандартной библиотеки Python.	2	4	6
Тема 3.5. Дополнительные библиотеки Python.	8	4	12
Тема 3.6. Введение в ООП	5	5	10
Тема 3.7. Решение задач по теме модуля.		8	8
Тема 3.8. Итоговая контрольная работа.		2	2
Тема 3.9. Разбор контрольной работы.		2	2
<b>Итого</b>	<b>69</b>	<b>75</b>	<b>144</b>

### III. СОДЕРЖАНИЕ ПРОГРАММЫ

**ВВОДНЫЙ МОДУЛЬ (16 часов: теория – 6 часов, практика – 10 часов).**

**Диагностическая работа (2 часа).**

**Введение в программирование. Основные понятия (теория – 2 часа).**

Python как свободный интерпретируемый объектно-ориентированный расширяемый встраиваемый язык программирования очень высокого уровня. Синтаксис. Типы данных. Ввод данных. Условные инструкции.

**Теория множеств. Булевы операции (всего – 6 часов: теория - 2 часа, практика – 4 часа).**

Бесконечное счетное множество. Взаимно однозначное соответствие. Запись и операция. Краткий обзор операций, обозначений и диаграмм Венна. Универсальное множество. Мощность и показательные множества. Функции. Математическая логика. Основные разделы математической логики – логика высказываний и логика предикатов. Булевы функции. Алгебра логики как раздел математической логики. Существенная переменная, несущественная (фиктивная) переменная. Символика и математический язык описания булевых функций. Ознакомление с основами булевой алгебры. Изучение основных логических элементов. Решение задач анализа и синтеза логических функций и схем.

**Системы счисления (всего – 4 часов: теория – 1 часа, практика – 3 часа).**

Непозиционные и позиционные системы счисления. Основание системы счисления. Вес разряда. Нумерация разряда. Системы счисления с кратными основаниями

Типовые задания по теме «Системы счисления». Однородная и смешанная системы.

Перевод и преобразования в системах. Вещественные числа в различных системах счисления. Представление целых и вещественных чисел в ЭВМ. Сравнение систем. Таблицы истинности.

**Высказывания. Элементы логики (всего – 2 часов: теория – 1 час, практика – 1 час).**

Алгебра логики. Высказывание. Логическая операция. Конъюнкция. Дизъюнкция. Отрицание. Логическое выражение. Таблица истинности. Законы логики.

*Учащиеся должны знать:*

- Основные понятия программирования;
- Синтаксис.
- Типы данных.
- Ввод данных.
- Условные инструкции.
- Теорию множеств. Булевы операции.

*Учащиеся должны уметь:*

- Устанавливать необходимый инструмент для разработки приложений;
- Пользоваться символикой и математическим языком описания булевых функций
- Решать задачи анализа и синтеза логических функций и схем.

- 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.
- 

## **ОСНОВНОЙ МОДУЛЬ (144 часа: теория – 66 часов, практика – 78 часов).**

### **Модуль 1. Введение в программирование (всего 144 часа: теория – 14 часов, практика – 16 часов).**

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 решают большое количество задач: от самых простых до сложных.

*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).

*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;
- составлять программы обработки списков;
- 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.

**Тема 1.1.** Знакомство с системой Яндекс.Контест.

*Теория.* Понятия кода, интерпретатора, программы. Интегрированные среды, исполнение кода.

*Практика.* Простейшие программы с выводом на экран.

**Тема 1.2.** Условный оператор. Переменные и арифметика.

*Теория.* Алфавит и словарь языка Python. Объявление переменных. Типы данных. Арифметические операции. Операции с присваиванием. Линейные программы. Условный оператор.

*Практика.* Обзор типов данных. Вычисление математических выражений с помощью стандартных арифметических функций. Решение задач по теме.

**Тема 1.3.** Знакомство с циклом while.

*Теория.* Определение цикла. Оператор цикла с предусловием.

*Практика.* Решение задач по теме «Цикл while».

**Тема 1.4.** Отладчик.

*Теория.* Определение. Установка и запуск отладчика программного кода.

*Практика.* Решение задач.

*Форма подведения итогов:* самостоятельная работа на решение задач по теме 2.4.

**Тема 1.5.** Знакомство с циклом for.

*Теория.* Оператор цикла с постусловием. Оператор с известным числом повторов.

*Практика.* Решение задач по теме «Цикл for».

**Тема 1.6.** Погружение в условия. True, False, Break, Continue.

*Теория.* Принцип работы операторов True, False, Break, Continue.

*Практика.* Решение задач на тему «Погружение в условия».

**Тема 1.7.** Знакомство со списками.

*Теория.* Определение понятия «списки». Объявление списка. Действия над списками, над элементами списка.

*Практика.* Решение задач по теме «Списки»

*Форма подведения итогов:* самостоятельная работа по теме модуля.

**Модуль 2. Базовые конструкции языка Python (всего - 42 часа: теория – 20 часов, практика – 22 часа).**

Понятие о языке высокого уровня Python. Структура программы, переменные и константы, работа с числовыми переменными, арифметические операторы в Python. Основные управляющие конструкции алгоритмов с ветвлением в Python.

Устройство циклов for. Основные управляющие конструкции циклического алгоритма в Python. Простейшие циклы и циклы с переменными.

Работа со списками, строками, множествами и кортежами в Python. Понятие итератора.

Понятие подпрограммы, процедуры, функции. Функции 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.

**Тема 2.1.** Повторение. Дополнительные возможности цикла for. Кортежи. Сортировки.

*Теория.* Структура программы. Стандартные функции и арифметические выражения. Ветвления. Логические операции. Оператор цикла с известным количеством повторов, постусловием, предусловием. Оператор безусловного перехода.

*Практика.* Решение задач на тему «Цикл for».

**Тема 2.2.** Знакомство со срезами и диапазонами.

*Теория.* Понятие срезов и диапазонов. Равенство и совпадение объектов.

*Практика.* Решение задач по теме «Срезы и диапазоны»

**Тема 2.3.** Списочные выражения. Методы split и join.

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

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

**Тема 2.4.** Другие методы списков и строк.

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

*Практика.* Решение задач на методы списков и строк.

**Тема 2.5.** Знакомство с функциями.

*Теория.* Определение функции. Правила составления и вызова функций в программном коде. Локальные и глобальные переменные.

*Практика.* Решение задач на тему «Функции».

**Тема 2.6.** Области видимости переменных.

*Теория.* Локальная область видимости. Глобальная область видимости. Нелокальная область видимости

*Практика.* Решение задач на тему «Области видимости переменных».

**Тема 2.7.** Введение в компьютерную графику. Модуль TKInter.

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

*Практика.* Решение задач на тему «Модуль TKInter».

*Форма подведения итогов:* контрольная работа по теме модуля.

**Модуль 3. Решение прикладных задач в Python (всего – 72 часа: теория – 35 часов, практика – 37 часов).**

Понятие ассоциативного массива. Словари в 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.

Самостоятельная работа: работа с конспектом, который описывает способы решения задач, разработка проекта по индивидуальному заданию, составление отчёта о выполнении индивидуальной или совместной работы.

Тематика самостоятельных работ:

- 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.

Выполненные самостоятельные работы загружаются в автоматизированную систему проверки задач Яндекс. Контест для дальнейшей оценки.

Зачётное занятие: выполнение итогового индивидуального задания по предложенной теме.

*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;
- 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,
- контрольная работа.

**Тема 3.1.** Повторение материала 1-го и 2-го модулей.

*Теория.* Условные операторы. Операторы цикла. Работа со списками, их методами.

*Практика.* Решение задач.

**Тема 3.2.** Функции. Детальное изучение. Функциональная парадигма программирования.

*Теория.* Понятие функции. Именные функции, инструкция def. Аргументы



функции. Анонимные функции, инструкция lambda.

*Практика.* Решения задач на тему «Функции и функциональная парадигма программирования».

**Тема 3.3.** Словари и множества.

*Теория.* Словари (dict) и работа с ними. Методы словарей. Множества (set и frozenset).

*Практика.* Решение задач на тему «Словари и множества».

**Тема 3.4.** Обзор стандартной библиотеки Python.

*Теория.* Работа с модулями: создание, подключение инструкциями import и from. Модуль os, sys. Модуль contextlib. Модуль abc. Модуль datetime. Модуль collections.

*Практика.* Решение задач на тему «Стандартная библиотека Python».

**Тема 3.5.** Дополнительные библиотеки Python.

*Теория.* Модуль string, re, struct, CSV, smtplib. Утилита pip.

*Практика.* Решение задач на тему «Дополнительные библиотеки Python».

**Тема 3.6.** Введение в ООП.

*Теория.* Объектно-ориентированное программирование. Классы и объекты.

*Практика.* Решение задач на тему «Введение в ООП».

**Тема 3.7.** Итоговая контрольная работа по темам модулей 1, 2 и 3.

*Форма подведения итогов:* контрольная работа.

#### IV. МЕТОДИЧЕСКОЕ ОБЕСПЕЧЕНИЕ

Дополнительная общеобразовательная общеразвивающая программа «Основы программирования на языке Python (углубленный уровень)» разработана с учетом современных требований к организации образовательного процесса.

Программа основана на постулатах личностно-ориентированного образования, особенностью которого является актуализация гуманитарных, человекообразующих функций обучения: сохранение и восстановление телесного и духовного здоровья, личной свободы, нравственности; гуманный подход направлен не на формирование ребенка, а на обеспечение поддержки в его развитии, создание механизмов саморегуляции, саморазвития, адаптации, самозащиты, самовоспитания личности; акцентирование внимания не на вооружении ребенка знаниями и умениями, а его становлении, обретении им неповторимой индивидуальности, духовности, творческого начала; построение учебного процесса осуществляется на основе образовательных технологий, основными признаками которых является диалогичность, деятельностный творческий характер, поддержка индивидуального развития ребенка, предоставление ему необходимого пространства свободы, творчества и др.

При проектировании программы учтены следующие педагогические принципы:

- *научности и доступности обучения;*
- *упорядоченности педагогического процесса* (целостности, систематичности, последовательности), что означает достижение единства и взаимосвязи между всеми компонентами педагогического процесса;
- *природосообразности*, т.е. образовательный процесс проектируется с учетом возрастных, гендерных и индивидуальных особенностей обучающихся;
- *гуманизации* - педагогический процесс строится на полном признании гражданских прав ребенка и уважения к нему;
- *демократизации* - предоставление всем участникам образовательного процесса свободы для саморазвития, саморегуляции, самоопределения, самообучения, самовоспитания;
- *культуросообразности* - максимальное использование в образовательном процессе той культурной среды, в которой находится МБУ ДО ДТДМ;
- *принципа единства и непротиворечивости действий учебного заведения и образа жизни учащихся*, что способствует установлению взаимосвязей между сферами жизни обучающихся;
- *принцип дифференциации* реализуется на всех годах обучения посредством индивидуальных дифференцированных заданий и упражнений.

Программа построена так же на принципах *погружения в среду визуального программирования*. Обучающиеся изучают основы алгоритмизации и программирования затем, на практических примерах, разбирая реальные существующие задачи, выполняя на каждом занятии учебные проекты, и, наконец, получают практический опыт командной работы над итоговым проектом-приложением с последующей «публичной» защитой проекта.

Творческая деятельность обучающихся по решению задач позволяет наглядно увидеть результаты своей работы и оценить полезность и значимость

развития навыков программирования для жизни. Ребята осваивают конструкции языка Python и учатся решать возникающие задачи, как в одиночку, так и в команде. Учатся презентовать – защищать свой проект перед аудиторией.

В процессе реализации программы используются педагогические технологии личностно-ориентированного образования, сотрудничества, индивидуального, дифференцированного, компетентного подходов, проектно-исследовательская, проблемного обучения, рефлексивная, и др.

В целях обеспечения целостного характера педагогической деятельности при реализации данной образовательной программы, предусмотрены мероприятия психолого-педагогического сопровождения: диагностические исследования совместно с педагогами-психологами МБУ ДО ДТДМ.

**Таким образом, основными компонентами образовательного процесса при реализации данной образовательной программы являются:**

- практический компонент, направленный на формирование у учащихся практических знаний и умений;
- компонент самостоятельности, пронизывающий все сферы и виды деятельности, способствующий воспитанию самостоятельной личности;
- творческий компонент, направленный на развитие у учащихся творческих компетенций и творческого подхода к своей деятельности;
- исследовательский, направленный на формирование научного мировоззрения, исследовательского отношения к окружающему миру, развивающий способности к анализу, альтернативному мышлению в выборе различных способов деятельности.

Педагогические методы и приемы, применяемые при освоении данной программы, способствуют комплексному решению педагогических задач по созданию творческой, развивающей среды для обучающихся.

Тема модуля	Форма занятий	Приёмы и методы организации образовательного процесса	Дидактический материал. Электронные источники	Техническое оснащение и расходный материал	Форма подведения итогов
<b>«ОСНОВЫ ПРОГРАММИРОВАНИЯ НА ЯЗЫКЕ PYTHON»</b>					
Модуль 1. Введение в программирование	Комбинированная	Метод проектов. Объяснительно-иллюстративный. 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. ( <a href="https://contest.yandex.ru">https://contest.yandex.ru</a> ); 2. LMS AnyTask – система управления обучением, позволяющая автоматизировать большинство внутренних процессов Яндекс.Лицея ( <a href="http://anytask.org">http://anytask.org</a> ); 3. Сервис PythonTutor, позволяющий визуализировать исполнение кода на языке Python ( <a href="http://pythontutor.com">http://pythontutor.com</a> ); 4. Федеральный портал Единая коллекция образовательных ресурсов	- Ноутбуки с мышкой и доступом к сети Интернет. Презентационное оборудование. -Принтер. -Маркерная доска.	Контрольная работа.

			<a href="http://school-collection.edu.ru">http://school-collection.edu.ru</a> ); 6.Официальная документация языка Python ( <a href="http://docs.python.org">http://docs.python.org</a> ).		
Модуль 2. Базовые конструкции в Python	Комбинированная	Метод проектов. Объяснительно-иллюстративный. 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. ( <a href="https://contest.yandex.ru">https://contest.yandex.ru</a> ); 2. LMS AnyTask – система управления обучением, позволяющая автоматизировать большинство внутренних процессов Яндекс.Лицея ( <a href="http://anytask.org">http://anytask.org</a> ); 3. Сервис PythonTutor, позволяющий визуализировать исполнение кода на языке Python ( <a href="http://pythontutor.com">http://pythontutor.com</a> ); 4. Федеральный портал Единая коллекция образовательных ресурсов ( <a href="http://school-collection.edu.ru">http://school-collection.edu.ru</a> ); 6. Официальная документация языка Python ( <a href="http://docs.python.org">http://docs.python.org</a> ).	- Ноутбуки с мышкой и доступом к сети Интернет. - Презентационное оборудование. - Принтер. - Маркерная доска.	Контрольная работа.
Модуль 3. Решение прикладных задач в Python	Комбинированная	Метод проектов. Объяснительно-иллюстративный. 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. ( <a href="https://contest.yandex.ru">https://contest.yandex.ru</a> ); 2. LMS AnyTask – система управления обучением, позволяющая автоматизировать большинство внутренних процессов Яндекс.Лицея ( <a href="http://anytask.org">http://anytask.org</a> ); 3. Сервис PythonTutor, позволяющий визуализировать исполнение кода на языке Python ( <a href="http://pythontutor.com">http://pythontutor.com</a> ); 4. Федеральный портал Единая коллекция образовательных ресурсов ( <a href="http://school-collection.edu.ru">http://school-collection.edu.ru</a> ); 6. Официальная документация языка Python ( <a href="http://docs.python.org">http://docs.python.org</a> ).	- Ноутбуки с мышкой и доступом к сети Интернет. - Презентационное оборудование. - Принтер. - Маркерная доска.	Контрольная работа.

## V. ДИАГНОСТИЧЕСКИЙ ИНСТРУМЕНТАРИЙ

### Формы контроля результатов освоения программы

Методика обучения ориентирована на индивидуальный подход. Для того чтобы каждый ученик получил наилучший результат обучения, программой предусмотрены индивидуальные задания для самостоятельной работы на домашнем 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. Количество таких заданий в работе может варьироваться.

В ходе обучения проводится промежуточное 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.

*Формы и методы контроля:*

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

По окончании обучения проводится итоговая аттестация в форме публичной защиты проектов. Документальной формой подтверждения итогов промежуточной аттестации является документ об образовании установленного ИТ кубом образца и сертификат от компании-партнера Яндекс -лицей.

**Первичная и итоговая диагностика** проводится партнером программы Яндекс.Лицей на ресурсах партнера.

### Результативность

№ раздела	Тема	Максимальный балл
1	Диагностическая работа	10
2	Вводный модуль	10
3	Модуль 1. Введение в программирование	10
4	Модуль 2. Базовые конструкции в Python	10
5	Модуль 3. Решение прикладных задач в Python	10
6	Итоговая контрольная работа	10
		60 баллов

## Диагностика личностных и метапредметных образовательных результатов

Ожидаемый результат	Параметры	Критерии	Методы отслеживания
Умение школьников принимать неочевидные решения, видеть нестандартные ходы как в учебной деятельности, так и в повседневной жизни.	Изобретение школьниками способов решения проблем, по новизне превосходящих авторские (общепринятые)	Оригинальность таких решений	Анализ разрозненной информации
Значительное опережение сверстников в областях знаний, связанных с информатикой и WWW.	Наличие обращений за помощью по предмету	Количество обращений	Наблюдения, беседа
Умение эффективно работать над поставленной проблемой в коллективе.	Соотношение коллективного и индивидуальных результатов	Наличие и адекватность распределения ролей в коллективе в ходе совместного решения проблем. Сравнение коллективного и личных результатов	Наблюдение Беседа Эксперимент
Изменения круга общения ребенка		Рост количества друзей среди членов объединения	Социометрия Анкетирование Наблюдение Эксперимент
Устойчивый интерес к предмету и к внепрограммному материалу	Место учебного предмета в жизни ребенка	Приведение самостоятельных примеров на уроках	Беседа с родителями Наблюдение
	Обращение к педагогу по вопросам содержания, непосредственно не связанным с изучаемым материалом	Количество обращений. Характер вопросов и сообщений, глубина заинтересованности	Статистика (беседы при личной встрече, по телефону, e-mail)
Способность самостоятельно изучать материал	Наличие умения самостоятельно изучать трудные или значительные по объему темы	Степень самостоятельности (участие педагога) Качество усвоения	Самоанализ Беседа Проверка работ

Умение планировать свою деятельность	Развитие навыков планирования	Количество усвоенных компонент (построение сложных планов, учет взаимосвязей при «распараллеливании работы»)	Наблюдение Эксперимент Беседа с родителями
Способность к самоконтролю	Умение контролировать ход выполнения работ, требующих длительного времени	Эффективность и результативность контроля	Наблюдение Эксперимент Беседа с родителями
Умение составлять запросы	Успешность результатов	Уровень конкретики. Красота идей	Беседа
Наличие определенной культуры при решении задач	Умение понятно излагать свои мысли как устно, так и письменно	Отсутствие неверно понятых рассуждений сверстниками и взрослыми	Наблюдение Сравнение результатов на соревнованиях до и после апелляции с последующим выяснением причины в беседе с ребенком
Успешное выступление перед сверстниками	Рост успехов школьников (каждого в отдельности) и статистика по учебной группе	Сравнение уровня соревнований, набранных баллов, дипломов, мест	Анализ результатов соревнований

## VI. СПИСОК ЛИТЕРАТУРЫ

### Нормативная литература

1. Государственная программа Российской Федерации «Развитие образования» на 2019-2025 г, утвержденная Постановлением Правительства Российской Федерации от 26 декабря 2017 года № 1642 (ред. от 15.03.2021).
2. Приказ Министерства просвещения Российской Федерации №467 от 03.09.2019 г. «Об утверждении Целевой модели развития региональных систем дополнительного образования».
3. Распоряжение Министерства просвещения Российской Федерации №Р-126 от 21.06.2021 г. «Об утверждении ведомственной целевой программы «Развитие дополнительного образования детей, выявление и поддержка лиц, проявивших выдающиеся способности».
4. Государственная программа Ростовской области «Развитие образования», утверждена постановлением Правительства Ростовской области от 17.10.2018 № 646 (с изменениями на 28 декабря 2020 года).
5. Конвенция о правах ребенка (принята резолюцией 44/25 Генеральной Ассамблеи от 20 ноября 1989 г.) — URL: [http://www.un.org/ru/documents/decl\\_conv/conventions/childcon.shtml](http://www.un.org/ru/documents/decl_conv/conventions/childcon.shtml).
6. Концепция развития дополнительного образования детей до 2030 года, утвержденная распоряжением Правительства Российской Федерации от 31 марта 2022 г. № 678-р.
7. Национальный проект «Образование», утвержденный на заседании президиума Совета при Президенте Российской Федерации по стратегическому развитию и национальным проектам (протокол от 24 декабря 2018 г. № 16).
8. Постановление Правительства Российской Федерации от 31 октября 2018 г. № 1288 (ред. от 10.07.2020, № 1019) «Об организации проектной деятельности в Правительстве Российской Федерации».
9. Приказ Министерства просвещения Российской Федерации от 9 ноября 2018 г. № 196 (ред. от 30.09.2020 г.) «Об утверждении Порядка организации и осуществления образовательной деятельности по дополнительным общеобразовательным программам».
10. Приказ Министерства просвещения Российской Федерации от 30 сентября 2020 г. № 533 «О внесении изменений в Порядок организации и осуществления образовательной деятельности по дополнительным общеобразовательным программам, утвержденный приказом Министерства просвещения Российской Федерации от 9 ноября 2018 г. № 196».
11. Приказ Министерства труда и социальной защиты Российской Федерации от 22 сентября 2021 г. N 652н н «Об утверждении профессионального стандарта «Педагог дополнительного образования детей и взрослых».
12. Распоряжение Правительства Российской Федерации от 23.01.2021г. № 122-р «Об утверждении Плана основных мероприятий, проводимых в рамках Десятилетия детства, на период до 2027 года.



13. СанПиН 2.4.3648–20 «Санитарно-эпидемиологические требования к организациям воспитания, обучения, отдыха и оздоровления детей и молодежи», Утверждены постановлением Главного государственного санитарного врача Российской Федерации от 28.09.2020 № 28 (зарегистрировано Минюстом России 18.12.2020, регистрационный № 61573).

14. Стратегическая инициатива «Новая модель системы дополнительного образования», одобренная Президентом Российской Федерации 27 мая 2015 г.

15. Стратегия развития воспитания в Российской Федерации на период до 2025 года, утвержденная Распоряжением Правительства Российской Федерации от 29 мая 2015 г. № 996-р.

16. Указ Президента Российской Федерации от 29 мая 2017 г. № 240 «Об объявлении в Российской Федерации Десятилетия детства».

17. Указ Президента Российской Федерации от 21 июля 2020 г. № 474 «О национальных целях и стратегических задачах развития Российской Федерации на период до 2030 года».

18. Федеральный закон от 29.12.2012 N 273-ФЗ (ред. от 24.03.2021) «Об образовании в Российской Федерации».

19. Федеральный проект «Успех каждого ребенка», утвержденный президиумом Совета при Президенте Российской Федерации по стратегическому развитию и национальным проектам (протокол от 3 сентября 2018 года № 10).

### **Основная литература**

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.

### **Дополнительная литература**

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.

### **Электронные ресурсы**

1. Материалы и презентации к урокам в LMS Яндекс.Лицея.

2. Сайт [pythonworld.ru](http://pythonworld.ru) — «Python 3 для начинающих».

3. Сайт [pythontutor.ru](http://pythontutor.ru) — «Питонтьютор».

4. <https://www.youtube.com/playlist?list=PLJOzdkh8T5kpIBTG9mM2wVBjh-5OpdwBl> — Лекции А.В. Умнова, прочитанные в Школе Анализа Данных Яндекса.

## VII. ПРИЛОЖЕНИЕ

Приложение 1

### Календарный учебный график

№ п/п	Тема	Количество часов			Форма контроля
		теория	практика	всего	
<b>Вводный модуль</b>		<b>10</b>	<b>6</b>	<b>16</b>	
1.	Диагностическая работа	2		2	тестирование
2.	Введение в программирование. Основные понятия		2	2	
3.	Теория множеств. Булевы операции	1	1	2	
4.	Теория множеств. Булевы операции	1	1	2	
5.	Теория множеств. Булевы операции		2	2	
6.	Системы счисления	1	1	2	
7.	Системы счисления		2	2	
8.	Высказывания. Элементы логики	1	1	2	тестирование
<b>Основной модуль</b>		<b>66</b>	<b>78</b>	<b>144</b>	
9.	Знакомство с системой Яндекс. Контекст. Понятия кода, интерпретатора, программы. Интегрированные среды, исполнение кода. Простейшие программы с выводом на экран.	1	1	2	
10.	Условный оператор. Переменные и арифметика.	1	1	2	
11.	Условный оператор. Переменные и арифметика.	1	1	2	
12.	Знакомство с циклом while.	1	1	2	
13.	Знакомство с циклом while.	1	1	2	
14.	Отладчик.	1	1	2	
15.	Отладчик.	1	1	2	
16.	Знакомство с циклом for.	1	1	2	
17.	Знакомство с циклом for.	1	1	2	
18.	Погружение в условия. True, False, Break, Continue.	1	1	2	
19.	Погружение в условия. True, False, Break, Continue.	1	1	2	
20.	Знакомство со списками.	1	1	2	
21.	Знакомство со списками.	1	1	2	
22.	Решение задач по теме модуля 1.		2	2	
23.	Решение задач по теме модуля 1.		2	2	тестирование

24.	Дополнительные возможности цикла for. Кортежи. Сортировки.	1	1	2	
25.	Дополнительные возможности цикла for. Кортежи. Сортировки.	1	1	2	
26.	Знакомство со срезами и диапазонами. Равенство и совпадение объектов.	1	1	2	
27.	Знакомство со срезами и диапазонами. Равенство и совпадение объектов.	1	1	2	
28.	Списочные выражения. Методы split и join.	1	1	2	
29.	Списочные выражения. Методы split и join.	1	1	2	
30.	Другие методы списков и строк.	1	1	2	
31.	Другие методы списков и строк.	1	1	2	
32.	Функции.	2		2	
33.	Функции.	1	1	2	
34.	Функции.	1	1	2	
35.	Области видимости переменных.	2		2	
36.	Области видимости переменных.	1	1	2	
37.	Области видимости переменных.	1	1	2	
38.	Контрольная работа по темам 1 и 2 модулей.		2	2	тестирование
39.	Разбор контрольной работы.		2	2	
40.	Введение в компьютерную графику. Модуль TKInter.	2		2	
41.	Введение в компьютерную графику. Модуль TKInter.	1	1	2	
42.	Введение в компьютерную графику. Модуль TKInter.	1	1	2	
43.	Решение задач по теме модуля.		2	2	
44.	Решение задач по теме модуля.		2	2	тестирование
45.	Повторение материала 1 и 2 модулей.	1	1	2	
46.	Повторение материала 1 и 2 модулей.	1	1	2	
47.	Функции. Детальное изучение. Функциональная парадигма программирования.	2		2	

48.	Функции. Детальное изучение. Функциональная парадигма программирования.	1	1	2	
49.	Функции. Детальное изучение. Функциональная парадигма программирования.	1	1	2	
50.	Функции. Детальное изучение. Функциональная парадигма программирования.	1	1	2	
51.	Функции. Детальное изучение. Функциональная парадигма программирования.	1	1	2	
52.	Функции. Детальное изучение. Функциональная парадигма программирования.	1	1	2	
53.	Функции. Детальное изучение. Функциональная парадигма программирования.	1	1	2	
54.	Функции. Детальное изучение. Функциональная парадигма программирования.	1	1	2	
55.	Словари и множества.	2		2	
56.	Словари и множества.	1	1	2	
57.	Словари и множества.	1	1	2	
58.	Словари и множества.	1	1	2	
59.	Словари и множества.	1	1	2	
60.	Словари и множества.	1	1	2	
61.	Обзор стандартной библиотеки Python.	1	1	2	
62.	Обзор стандартной библиотеки Python.	1	1	2	
63.	Обзор стандартной библиотеки Python.		2	2	
64.	Дополнительные библиотеки Python.	2		2	
65.	Дополнительные библиотеки Python.	2		2	
66.	Дополнительные библиотеки Python.	1	1	2	
67.	Дополнительные библиотеки Python.	1	1	2	
68.	Дополнительные библиотеки Python.	1	1	2	
69.	Дополнительные библиотеки Python.	1	1	2	
70.	Введение в ООП	1	1	2	

71.	Введение в ООП	1	1	2	
72.	Введение в ООП	1	1	2	
73.	Введение в ООП	1	1	2	
74.	Введение в ООП	1	1	2	
75.	Решение задач по теме модуля.		2	2	
76.	Решение задач по теме модуля.		2	2	
77.	Решение задач по теме модуля.		2	2	
78.	Решение задач по теме модуля.		2	2	тестирование
79.	Итоговая контрольная работа.		2	2	тестирование
80.	Разбор контрольной работы.		2	2	
<b>Итого</b>		<b>76</b>	<b>84</b>	<b>160</b>	

## Диагностика эффективности образовательного процесса

### Классификация занятий по дидактической цели и форме

№	Классификация занятий по дидактической цели	Форма занятия
1.	Изучение и первичное закрепление нового учебного материала	Очная лекция
2.	Комплексное применение знаний	Практическое занятие по решению задач
3.	Обобщение и систематизация знаний	Постановка и решение социальных задач
4.	Актуализация знаний и умений	Самостоятельная работа обучающихся по решению задач с использованием WWW/
5.	Контроль и коррекция знаний и умений	Презентация полученных знаний

### Бланк наблюдения за обучающимися

Группа \_\_\_\_\_

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

За каждое согласие с утверждением - 1 балл.

Результаты контроля фиксируются в диагностической карте.

## Диагностическая карта

Уровни освоения программы	Параметры	Показатели
<b>Высокий уровень (80-100%)</b>	Теоретические знания.	Обучающийся освоил материал в полном объеме. Знает и понимает значение терминов, самостоятельно ориентируется в содержании материала по темам. Учащийся заинтересован, проявляет устойчивое внимание к выполнению заданий.
Практические умения и навыки.		Способен применять практические умения и навыки во время выполнения самостоятельных заданий. Работу аккуратно доводит до конца. Может оценить результаты выполнения своего задания и дать оценку работы своего товарища.
<b>Средний уровень (50-79%)</b>	Теоретические знания.	Учащийся освоил базовые знания, ориентируется в содержании материала по темам, иногда обращается за помощью к педагогу. Учащийся заинтересован, но не всегда проявляет устойчивое внимание к выполнению задания.
Практические умения и навыки.		Владеет базовыми навыками и умениями, но не всегда может выполнить самостоятельное задание, затрудняется и просит помощи педагога. В работе допускает небрежность, делает ошибки, но может устранить их после наводящих вопросов или самостоятельно. Оценить результаты своей деятельности может с подсказкой педагога.