

УТВЕРЖДЕНО:

И. о. директора ГАОУ ДО

«Центр для одаренных детей «Поиск»

О.А. Томиной,

приказ № 71 от 12 марта 2024 г.

Дополнительная общеобразовательная общеразвивающая программа  
**Яндекс Лицей**

Направление: наука

Возраст обучающихся: 14-18 лет

Объем программы: 280 часов.

Срок освоения: 2 года

Форма обучения: очная

Авторы программы: Бунина Е.И. Председатель Совета Автономной некоммерческой организации дополнительного профессионального образования «Школа анализа данных»,  
Умнов А.В., кандидат педагогических наук, методист Автономной некоммерческой организации дополнительного профессионального образования «Школа анализа данных»  
Паволоцкий А.В., кандидат педагогических наук, методист Автономной некоммерческой организации дополнительного профессионального образования «Школа анализа данных»  
Селезнёв А.О., методист Невинномысского филиала Центра «Поиск»

## ОГЛАВЛЕНИЕ

<b>ПОЯСНИТЕЛЬНАЯ ЗАПИСКА.....</b>	<b>3</b>
<b>1. ОСНОВНЫЕ ХАРАКТЕРИСТИКИ ПРОГРАММЫ.....</b>	<b>4</b>
<b>2. УЧЕБНЫЙ ПЛАН.....</b>	<b>11</b>
<b>3. РАБОЧАЯ ПРОГРАММА УЧЕБНОГО КУРСА «ОСНОВЫ ПРОГРАММИРОВАНИЯ НА PYTHON».....</b>	<b>12</b>
<b>4. РАБОЧАЯ ПРОГРАММА УЧЕБНОГО КУРСА «ОСНОВЫ ПРОМЫШЛЕННОГО ПРОГРАММИРОВАНИЯ».....</b>	<b>16</b>
<b>5. УЧЕБНО-МЕТОДИЧЕСКОЕ И ИНФОРМАЦИОННОЕ ОБЕСПЕЧЕНИЕ.....</b>	<b>21</b>

## ПОЯСНИТЕЛЬНАЯ ЗАПИСКА

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.

Опираясь на уникальный опыт преподавания программирования в Школе анализа данных Яндекса (АНО ДПО «ШАД») и на факультете компьютерных наук НИУ ВШЭ, была подготовлена данная программа. В ней большое внимание уделяется практической работе на компьютере, самостоятельному написанию кода.

Изучение основных принципов программирования невозможно без регулярной практики написания программ на каком-либо языке. Для обучения был выбран язык 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.

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

Знания и умения, приобретённые 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.

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

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

Важным аспектом программы является самостоятельная работа над заданиями: школьники учатся решать задачи без помощи преподавателя. Для этого в содержании курса фигурируют задания, в которых:

- для решения задачи необходимо найти какую-то информацию в сети Интернет;
- может потребоваться устранение ошибки, которую не так просто быстро обнаружить;
- условие сформулировано недостаточно прозрачно и ученику необходимо самостоятельно формализовать его (или задать правильные вопросы преподавателю).

**Вид программы** – модульная.

программа представляет собой совокупность 2-х самостоятельных логически завершённых курсов, которые реализуются в очной форме.

№	Название курса	Форма обучения	Класс обучающегося
1.	Основы программирования на языке Python (углубленный уровень)	очная	8-9 класс
2.	Основы промышленного программирования	очная	9-11 класс

## 1. ОСНОВНЫЕ ХАРАКТЕРИСТИКИ ПРОГРАММЫ

### 1.1. Направленность программы

Программа «Яндекс Лицей» имеет техническую направленность, в связи с этим рассматриваются следующие аспекты изучения:

- **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.
- **Общеразвивающий.** Обучение по данной программе создаёт благоприятные условия для интеллектуального воспитания личности ребенка, профессионального самоопределения, развития познавательной активности учащихся.
- **Общеобразовательный.** Содержание программы рассматривается как средство развития основных познавательных процессов, умения анализировать, выявлять сущности и отношения, описывать планы действий и делать логические выводы.

### 1.2. Новизна программы

Программа курса:

- 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 программы курса заключается в том, что она 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.

### **1.3. 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 получат мощный и удобный инструмент для решения как учебных, так и прикладных задач. Вместе с тем чистота и ясность его конструкций позволит учащимся потом с лёгкостью выучить любой другой язык программирования. Знания и умения, приобретенные в результате освоения курса, могут быть использованы учащимися при сдаче ЕГЭ, при участии в олимпиадах по программированию, 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.

### **1.4. Цель программы**

Целью программы является 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.

## 1.5. Задачи программы

Настоящая программа направлена на решение следующих задач:

- формирование и развитие навыков алгоритмического и логического мышления, грамотной разработки программ;
- знакомство с принципами и методами функционального программирования;
- знакомство с принципами и методами объектно-ориентированного программирования;
- приобретение навыков работы в интегрированной среде разработки на языке Python;
- изучение конструкций языка программирования Python;
- знакомство с основными структурами данных и типовыми методами обработки этих структур;
- приобретение навыков разработки эффективных алгоритмов и программ на основе изучения языка программирования Python;
- приобретение навыков поиска информации в сети Интернет, анализ выбранной информации на соответствие запросу, использование информации при решении задач;
- развитие у обучающихся интереса к программированию;
- формирование самостоятельности и творческого подхода к решению задач с использованием средств вычислительной техники;
- воспитание упорства в достижении результата;
- расширение кругозора обучающихся в области программирования.

**По окончании программы ученик приобретает 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.

## 1.6. 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.

### **1.7. Адресат программы**

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

Возраст обучающихся: 14 – 17 лет. Наполняемость группы: 12 – 14 человек.

### **1.8. Условия приема детей**

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

### **1.9. Объём и срок освоения программы**

Объём программы - 280 часов.

Срок реализации программы – 2 года.

1 год обучения: 136 часов в год,

2 год обучения: 144 часов в год.

### **1.10. Форма реализации программы**

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.

### **1.11. Формы организации деятельности обучающихся**

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.

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.

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.

### **1.12. Методы организации учебного процесса**

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.

#### **Формы и методы контроля:**

- тестирование;
- устный опрос;

- самостоятельные и контрольные работы;
- участие в проектной деятельности.

#### **Общая характеристика учебного процесса:**

- при изучении курса используются практические и самостоятельные работы;
- курс обучения заканчивается написанием программы для решения одной из задач.

### **1.13. Планируемые результаты освоения программы**

#### **Личностные результаты:**

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

- формирование и развитие компетентности 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;
- формирование представлений об основных предметных понятиях («информация», «алгоритм», «исполнитель», «модель») и их свойствах;
- развитие логических способностей и алгоритмического мышления, умения 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;
- формирование умения соблюдать нормы информационной этики и права.

## 2. УЧЕБНЫЙ ПЛАН

№	Наименование раздела, темы	Количество часов			Форма контроля
		Теория	Практика	Всего	
<b>1-й год обучения</b>					
1.	Модуль 1. Введение в программирование.	6	20	26	
2.	Модуль 2. Базовые конструкции в Python.	10	30	40	Контрольная работа
3.	Модуль 3. Решение прикладных задач в Python.	10	60	70	Контрольная работа
	<b>ИТОГО</b>	<b>26</b>	<b>110</b>	<b>136</b>	
<b>2-й год обучения</b>					
1.	Модуль 1. Программирование на Python.	6	28	34	Самостоятельная работа
2.	Модуль 2. Тестирование программ и командные методы работы	3	21	24	Контрольная работа
3.	Модуль 3. Проекты GUI, Игры и API.	14	38	72	Итоговый проект
	<b>ИТОГО</b>	<b>32</b>	<b>92</b>	<b>144</b>	
	<b>ИТОГО</b>	<b>58</b>	<b>202</b>	<b>280</b>	

### 3. РАБОЧАЯ ПРОГРАММА УЧЕБНОГО КУРСА «ОСНОВЫ ПРОГРАММИРОВАНИЯ НА PYTHON»

№	Наименование раздела, темы	Количество часов		
		Теория	Практика	Всего
	<b>Модуль1. Введение в программирование.</b>	<b>6</b>	<b>20</b>	<b>26</b>
1	Понятие кода.	1	1	2
2	Знакомство со средой.	1	1	2
3	Переменные, основные операторы. Базовые типы данных.		2	2
4	Условный оператор.	1	1	2
5	Простые встроенные функции.	1	1	2
6	Знакомство с циклом while.	1	1	2
7	Отладчик.		2	2
8	Знакомство с циклом for.	1	1	2
9	True и False, break и continue.		2	2
10	Вложенные циклы. Форматированный вывод.		2	2
11	Множества.		2	2
12	Строки. Индексация. Срезы.		2	2
13	Знакомство со списками.		2	2
	<b>Модуль 2. Базовые конструкции в Python</b>	<b>10</b>	<b>30</b>	<b>40</b>
14	Кортежи. Преобразование коллекций.	1	1	2
15	Методы split и join. Списочные выражения.	1	1	2
16	Методы списков и строк.	1	1	2
17	Пробная СР.		2	2
18	Резервное время.		2	2
19	Самостоятельная работа №1 на простые коллекции.		2	2
20	Вложенные списки.	1	1	2
21	Знакомство со словарями.	1	1	2
22	Практика по работе с графикой и tkinter.		2	2
23	Резервное время.		2	2

24	Резервное время.		2	2
25	Контрольная работа №1.		2	2
26	Функции.	1	1	2
27	Функции. Возвращение значений из функций.	1	1	2
28	Области видимости переменных.	1	1	2
29	Функции: передача параметров.	1	1	2
30	Функции с переменным числом аргументов.		2	2
31	Функции как объект. Лямбда-функции.	1	1	2
32	Обработка коллекций. Поточковый ввод sys.stdin.		2	2
33	Самостоятельная работа №2 на функции.		2	2
	<b>Модуль 3. Решение прикладных задач в Python</b>	<b>10</b>	<b>60</b>	<b>70</b>
34	Библиотеки Python. Встроенные модули.	1	1	2
35	Библиотеки Python. Работа с графическими файлами.		2	2
36	Библиотеки Python. Работа с графическими файлами и звуком.	1	1	2
37	Резервное время.		2	2
38	Библиотеки Python. Морфология.	1	1	2
39	Библиотеки Python. Работа с документами	1	1	2
40	Резервное время.		2	2
41	Переопределение функций и декораторы.		2	2
42	Самостоятельная работа №3 на библиотеки.		2	2
43	Введение в ООП.	1	1	2
44	Резервное время.		2	2
45	Введение в ООП. Полиморфизм.		2	2
46	ООП. Определение операторов.		2	2
47	ООП. Наследование.	1	1	2
48	ООП. Наследование. Часть №2.	1	1	2
49	Проектирование и разработка классов. Часть № 1.	1	1	2

50	Библиотеки Python. Numpy.		2	2
51	Проектирование и разработка классов. Часть № 2.		2	2
52	Итераторы и коллекции	1	1	2
53	Рекурсия	1	1	2
54	Резервное время		2	2
55	Решение задач по пройденным темам		2	2
56	Контрольная работа №2		2	2
57	Решение задач по пройденным темам		2	2
58	Решение задач по пройденным темам		2	2
59	Решение задач по пройденным темам		2	2
60	Решение задач по пройденным темам		2	2
61	Решение задач по пройденным темам		2	2
62	Решение задач по пройденным темам		2	2
63	Решение задач по пройденным темам		2	2
64	Решение задач по пройденным темам		2	2
65	Решение задач по пройденным темам		2	2
66	Решение задач по пройденным темам		2	2
67	Решение задач по пройденным темам		2	2
68	Работа над ошибками. Анализ результатов обучения.		2	2
	<b>ИТОГО</b>	<b>26</b>	<b>110</b>	<b>136</b>

# СОДЕРЖАНИЕ КУРСА «ОСНОВЫ ПРОГРАММИРОВАНИЯ НА ЯЗЫКЕ PYTHON»

## **Модуль 1. Введение в программирование**

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.

## **Модуль 2. Базовые конструкции языка Python**

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.

## **Модуль 3. Решение прикладных задач в Python**

Понятие ассоциативного массива. Словари в Python. Решение задач. 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. Основы ООП. Решение задач. 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.

**4. РАБОЧАЯ ПРОГРАММА УЧЕБНОГО КУРСА  
«ОСНОВЫ ПРОМЫШЛЕННОГО ПРОГРАММИРОВАНИЯ»**

№	Наименование раздела, темы	Количество часов		
		Теория	Практика	Всего
	<b>Модуль1. Программирование на Python.</b>	<b>6</b>	<b>28</b>	<b>34</b>
1	Повторение. Решение задач на основные конструкции и структуры данных. Решение задач на классы и библиотеки.	1	1	2
2	Повторение. Решение задач на классы и библиотеки.		2	2
3	Самостоятельная работа: Решение задач по теме «Повторение».		2	2
4	ООП. Наследование.	1	1	2
5	ООО. Наследование.		2	2
6	Самостоятельная работа: Решение задач по теме «ООП. Наследование».		2	2
7	Обработка исключительных ситуаций в Python.	1	1	2
8	Обработка исключительных ситуаций в Python.		2	2
9	Самостоятельная работа: Решение задач по теме «Обработка исключительных ситуаций в Python».		2	2
10	Рекурсия.	1	1	2
11	Самостоятельная работа: Решение задач по теме «Рекурсия».		2	2
12	Файлы в Python. Работа с различными типами файлов. Использование библиотек.	1	1	2
13	Файлы в Python. Работа с различными типами файлов. Использование библиотек.	1	1	2
14	Файлы в Python. Работа с различными типами файлов. Использование библиотек.		2	2
15	Самостоятельная работа: Решение задач по		2	2

	теме «Файлы в Python».			
16	Резервное время на решение задач.		2	2
17	Самостоятельная работа. Рубежный контроль.		2	2
	<b>Модуль 2. Тестирование программ и командные методы работы</b>	<b>3</b>	<b>21</b>	<b>24</b>
18	Подходы к тестированию. Создание простых тестов. Использование библиотек unittest и py.test.	1	1	2
19	Подходы к тестированию. Создание простых тестов. Использование библиотек unittest и py.test.		2	2
20	Самостоятельная работа: Решение задач по теме «Подходы к тестированию».		2	2
21	Работа с командной строкой. Библиотека argparse.	1	1	2
22	Работа с командной строкой. Библиотека argparse.		2	2
23	Самостоятельная работа: Решение задач по теме «Работа с командной строкой».		2	2
24	Репозитории. Совместная работа над проектами.	1	1	2
25	Репозитории. Совместная работа над проектами.		2	2
26	Самостоятельная работа: Решение задач по теме «Репозитории».		2	2
27	Резервное время на решение задач.		2	2
28	Самостоятельная работа. Рубежный контроль.		2	2
29	Контрольная работа по темам 1-го и 2-го модулей.		2	2
	<b>Модуль 3. Проекты GUI, Игры и API.</b>	<b>14</b>	<b>72</b>	<b>86</b>
30	Изучение виджетов.	1	1	2
31	Изучение виджетов.	1	1	2
32	Изучение виджетов.		2	2
33	Изучение виджетов.		2	2

34	Решение задач на тему «GUI».		2	2
35	Решение задач на тему «GUI».		2	2
36	Решение задач на тему «GUI».		2	2
37	Самостоятельная работа и её разбор.	1	1	2
38	Самостоятельная работа и её разбор.		2	2
39	Работа над проектом по теме «GUI» и его защита.	1	1	2
40	Работа над проектом по теме «GUI» и его защита.		2	2
41	Работа над проектом по теме «GUI» и его защита.		2	2
42	Работа над проектом по теме «GUI» и его защита.		2	2
43	Изучение игровых технологий и средств языка Python для их реализации.	1	1	2
44	Изучение игровых технологий и средств языка Python для их реализации.		2	2
45	Изучение игровых технологий и средств языка Python для их реализации.	1	1	2
46	Изучение игровых технологий и средств языка Python для их реализации.		2	2
47	Изучение игровых технологий и средств языка Python для их реализации.	1	1	2
48	Самостоятельная работа: Решение задач по теме «Игры».		2	2
49	Самостоятельная работа: Решение задач по теме «Игры».		2	2
50	Самостоятельная работа: Решение задач по теме «Игры».		2	2
51	Самостоятельная работа: Решение задач по теме «Игры».		2	2
52	Самостоятельная работа и её разбор.	1	1	2
53	Самостоятельная работа и её разбор.		2	2
54	Работа над проектом по теме «Игры» и его защита.	1	1	2

55	Работа над проектом по теме «Игры» и его защита.		2	2
56	Работа над проектом по теме «Игры» и его защита.		2	2
57	Работа над проектом по теме «Игры» и его защита.		2	2
58	Изучение некоторых API и работу с ними с помощью языка Python.	1	1	2
59	Изучение некоторых API и работу с ними с помощью языка Python.		2	2
60	Изучение некоторых API и работу с ними с помощью языка Python.	1	1	2
61	Изучение некоторых API и работу с ними с помощью языка Python.		2	2
62	Изучение некоторых API и работу с ними с помощью языка Python.	1	1	2
63	Самостоятельная работа: Решение задач по теме «API».		2	2
64	Самостоятельная работа: Решение задач по теме «API».		2	2
65	Самостоятельная работа: Решение задач по теме «API».		2	2
66	Самостоятельная работа: Решение задач по теме «API».		2	2
67	Самостоятельная работа: Решение задач по теме «API».		2	2
68	Самостоятельная работа и её разбор.	1	1	2
69	Работа над проектом по теме «API» и его защита.	1	1	2
70	Работа над проектом по теме «API» и его защита.		2	2
71	Работа над проектом по теме «API» и его защита.		2	2
72	Работа над проектом по теме «API» и его защита.		2	2
	<b>ИТОГО</b>	<b>23</b>	<b>121</b>	<b>144</b>

## **СОДЕРЖАНИЕ КУРСА «ОСНОВЫ ПРОМЫШЛЕННОГО ПРОГРАММИРОВАНИЯ»**

### **Модуль 1. Программирование на Python**

Объектно-ориентированное программирование. Наследование и исключения. Рекурсия. Работа с файлами. Файловые форматы: CSV, JSON, XML. На этом этапе обучающиеся продолжают осваивать язык Python и решать тренировочные задачи. В течение модуля ученики заканчивают изучение объектно-ориентированного подхода к программированию, рассматривают механизм поиска и обработки исключительных ситуаций (ошибок), знакомятся с различными файловыми. В течение модуля ученики решают большое количество задач различной сложности и пишут 2 самостоятельные работы.

### **Модуль 2. Тестирование программ и командные методы работы**

В этом небольшом по продолжительности модуле ученики разбирают устройство командной строки и принципы работы с ней, детально изучают принципы тестирования программ и соответствующие библиотеки языка Python, а также погружаются в методику групповой работы над проектом. В течение модуля школьники по традиции решают большое количество задач и пишут самостоятельную работу.

### **Модуль 3. Проекты GUI, Игры и API**

Модуль GUI открывает группу проектных тем курса. В теме «GUI» ученики рассматривают принципы создания пользовательских интерфейсов на языке Python с использованием соответствующих библиотек. Тема «Игры» предлагает к изучению различные виды компьютерных игр, технологии и алгоритмы для их проектирования. Тема «API» раскрывает способы интеграции различных сетевых приложений и сервисов. Каждая тема содержит мини-проекты и итоговый проект, который ученики по желанию защищают на итоговом занятии.

## **5. УЧЕБНО-МЕТОДИЧЕСКОЕ И ИНФОРМАЦИОННОЕ ОБЕСПЕЧЕНИЕ**

### **1. Методические учебные пособия:**

1. ФГОС. Примерные программы по информатике для основной и старшей школы. М.: БИНОМ. Лаборатория знаний.

### **2. Оборудование:**

1. Компьютерный класс (14 ученических ПК + 1 учительский ПК) с выходом в сеть Интернет
2. Маркерная доска
3. Проектор
4. Принтер

### **3. Цифровые образовательные ресурсы:**

1. Электронное приложение к учебникам К. Ю. Полякова Информатика и ИКТ. Базовый уровень. (10-11 кл.) набор цифровых ресурсов из единой коллекции цифровых образовательных ресурсов (дидактические материалы, интерактивные тесты, анимационные плакаты.) (<http://school-collection.edu.ru>).

### **4. Сетевые образовательные ресурсы:**

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. Всероссийский портал «Дистанционная подготовка по информатике» (<http://informatics.mcsme.ru>);
5. Федеральный портал Единая коллекция образовательных ресурсов (<http://school-collection.edu.ru>);
6. Официальная документация языка Python (<http://docs.python.org>).

### **5. Список литературы**

#### **Литература, использованная при подготовке 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.

### **Литература, рекомендованная 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.

### **Ресурсы в интернете**

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-OrdwB1>
5. OrdwB1 — Лекции А.В. Умнова, прочитанные в Школе Анализа Данных Яндекса.