



**МИНОБРНАУКИ РОССИИ**

**ФЕДЕРАЛЬНОЕ ГОСУДАРСТВЕННОЕ БЮДЖЕТНОЕ  
ОБРАЗОВАТЕЛЬНОЕ УЧРЕЖДЕНИЕ ВЫСШЕГО ОБРАЗОВАНИЯ  
«НАЦИОНАЛЬНЫЙ ИССЛЕДОВАТЕЛЬСКИЙ  
МОРДОВСКИЙ ГОСУДАРСТВЕННЫЙ УНИВЕРСИТЕТ  
им. Н.П. ОГАРЁВА»  
(ФГБОУ ВО «МГУ им. Н.П. Огарёва»)**



**УТВЕРЖДЕНО**

ученым советом

ФГБОУ ВО «МГУ им. Н.П. Огарёва»

(протокол № 20

от «11» августа 2022 г.)

Ректор

Д.Е. Глушко

**ДОПОЛНИТЕЛЬНАЯ ОБЩЕОБРАЗОВАТЕЛЬНАЯ  
ОБЩЕРАЗВИВАЮЩАЯ ПРОГРАММА  
ТЕХНИЧЕСКОЙ НАПРАВЛЕННОСТИ**

- **«Лицей Академии Яндекса. 2-й год. «Основы промышленного программирования»**

»

Форма обучения – очная

Нормативный срок освоения программы – 9 месяцев

Объем – 168 академических часа

САРАНСК 2022

## Структура программы

1. Пояснительная записка.	3
1.1 Направленность программы.	3
1.2 Актуальность программы.	3
1.3 Цель программы.	4
1.4 Задачи программы.	4
1.5 Отличительные особенности программы, новизна.	5
1.6 Нормативные правовые акты, на которых базируется разработка программы:	5
1.7 Адресат программы.	6
1.8 Объем программы	6
1.9 Срок освоения:	7
1.10 Форма обучения:	7
1.11 Особенности организации образовательного процесса.	7
1.12 Организационные формы обучения:	8
1.13 Режим занятий:	8
1.14 Планируемые результаты освоения программы	8
1.15 Документ об обучении, выдаваемый по окончании обучения, и условия его получения обучающимся.	9
2. Учебный план и учебно-тематический план	10
2.1 УЧЕБНЫЙ ПЛАН	10
2.2 УЧЕБНО-ТЕМАТИЧЕСКИЙ ПЛАН	11
3. Рабочие программы учебных предметов, курсов, дисциплин (модулей).	16
3.1 Программа модуля 1 «Повторение. Библиотека PyQT для создания GUI».	16
3.2 Программа модуля 2 «Библиотека PyGame»	19

3.3 Программа модуля 3 «Решение прикладных задач в Python»	22
4. Календарный учебный график дополнительной общеобразовательной общеразвивающей программы «Основы промышленного программирования»	27
5. Организационно-педагогические условия реализации программы	31
5.1 Кадровое обеспечение	31
5.2 Учебно-методическое и информационное обеспечение	32
5.3 Материально-техническое обеспечение	33
6. Нормативно-методическое обеспечение системы оценки качества освоения обучающимися дополнительной общеобразовательной общеразвивающей программы	34
6.1 Формы аттестации (указываются формы промежуточной и итоговой аттестации в соответствии с учебным планом)	34
6.2 Оценочные материалы	34
6.2.1 Комплект оценочных средств для проведения промежуточной аттестации	34
6.2.2 Комплект оценочных средств для проектной деятельности	35
6.2.3 Комплект оценочных средств для проведения итоговой аттестации	37
7. Сведения об обновлении программы	40

## **1. Пояснительная записка.**

### **1.1 Направленность программы.**

Техническая.

### **1.2 Актуальность программы.**

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 АНО ДПО «Образовательные технологии Яндекса» 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 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 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 курса «Основы промышленного программирования» 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).

### **1.3 Цель программы.**

Целью курса 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.

### **1.4 Задачи программы.**

- 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.

### **1.5 Отличительные особенности программы, новизна.**

- 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.

### **1.6 Нормативные правовые акты, на которых базируется разработка программы:**

- Федеральный закон от 29 декабря 2012 года №273-ФЗ «Об образовании в Российской Федерации»;
- – Постановление Правительства Республики Мордовия от 22 августа 2019 года № 352 «Об утверждении Порядка предоставления из республиканского бюджета Республики Мордовия гранта в форме субсидии некоммерческим организациям на обеспечение расходов по содержанию центров, реализующих дополнительные общеобразовательные программы, в организациях, осуществляющих образовательную деятельность по образовательным программам высшего образования, в том числе участвующих в создании научных и научно-

образовательных центров мирового уровня или обеспечивающих деятельность центров компетенций Национальной технологической инициативы»;

- Распоряжение Правительства РФ от 31 марта 2022 г. N 678-р «Об утверждении Концепции развития дополнительного образования детей до 2030 г. и плана мероприятий по ее реализации»;

- Приказ Министерства просвещения Российской Федерации от 27.07.2022 года № 629 «Об утверждении порядка организации и осуществления образовательной деятельности по дополнительным общеобразовательным программам»;

- Приказ Министерства образования и науки РФ от 23.08.2017 года № 816 «Об утверждении Порядка применения организациями, осуществляющими образовательную деятельность, электронного обучения, дистанционных образовательных технологий при реализации образовательных программ»;

- Приказ Министерства науки и высшего образования РФ и Министерства просвещения РФ от 5 августа 2020 г. N 882/391 «Об организации и осуществлении образовательной деятельности при сетевой форме реализации образовательных программ»;

- Приказ Министерства труда и социальной защиты Российской Федерации от 22 сентября 2021 г. N 652н «Об утверждении профессионального стандарта «Педагог дополнительного образования детей и взрослых»;

- Приказ Министерства образования Республики Мордовия от 4 марта 2019 года № 211 «Об утверждении Правил персонифицированного финансирования дополнительного образования детей»;

- Санитарные правила СП 2.4.3648-20 «Санитарно-эпидемиологические требования к организациям воспитания и обучения, отдыха и оздоровления детей и молодежи», утвержденные постановлением Главного государственного санитарного врача Российской Федерации от 28 сентября 2020 г. № 28 (зарегистрировано Министерством юстиции Российской Федерации 18 декабря 2020 г., регистрационный N 61573), действующие до 1 января 2027 года.

- Устав Университета и другие локальные нормативные акты Университета.

- 

## **1.7 Адресат программы.**

Дополнительная общеобразовательная (общеразвивающая) программа «Основы программирования на языке Python (углубленный уровень)» ориентирована на работу с учениками 9-го и 10-го классов школ разного уровня подготовки и с разной степенью мотивации.

## **1.8 Объем программы**

Продолжительность реализации всей программы -168 часов.

## **1.9 Срок освоения:**

Срок реализации программы - 8 месяцев.

## **1.10 Форма обучения:**

Очная.

## **1.11 Особенности организации образовательного процесса.**

Очная программа с применением ДОТ.

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 Организационные формы обучения:**

В процессе реализации программы используются формы: групповая, индивидуальная. Большинство занятий проводится в групповой форме. Группы, в которых проходят занятия могут быть разновозрастными.

### **1.13 Режим занятий:**

Занятия проводятся 2 раза в неделю по 2 часа (продолжительность учебного часа 45 минут).

### **1.14 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.

### **1.15 Документ об обучении, выдаваемый по окончании обучения, и условия его получения обучающимся.**

**Документ об обучении** – сертификат установленного образца (получают лица, освоившие программу в полном объеме и прошедшие итоговую аттестацию).

## 2. Учебный план и учебно-тематический план

### 2.1 УЧЕБНЫЙ ПЛАН

дополнительной общеобразовательной общеразвивающей программы  
«Основы программирования на языке Python (углубленный уровень)»

№	Наименование разделов/модулей	Всего часов	В том числе			Форма контроля
			Лекции	Практические занятия	Самостоятельная работа	
	Модуль 1. Повторение. Библиотека PyQT для создания GUI.	54	18	24	12	Наблюдение, анализ работ, устный опрос, контрольная работа, проектная деятельность
	Модуль 2. Библиотека PyGame.	37	13	15	9	Наблюдение, анализ работ, устный опрос, контрольная работа, проектная деятельность
	Модуль 2. Технологии для разработки WEB приложений.	77	27	33	17	Наблюдение, анализ работ, устный опрос, контрольная работа, проектная деятельность
	Итоговая аттестация					
	ИТОГО:	168	58	72	38	

## 2.2 УЧЕБНО-ТЕМАТИЧЕСКИЙ ПЛАН

### дополнительной общеобразовательной общеразвивающей программы «Основы промышленного программирования.»

№ п/п	Наименования Разделов/модулей и дисциплин/тем	Всего часов	В том числе:			Форма контроля
			лекции	практич еские занятия	самост оательн ая работа	
Модуль 1. _____ Повторение. Библиотека PyQT для создания GUI (наименование модуля)						
Всего: 54 часа (из них: 18 часов – лекционных, 24 часов – практических, 12 часов – самостоятельная работа)						
1.1	Повторение. Решение задач на основные конструкции данных	2	1	1	0	Наблюдение, опрос детей, анализ работ
1.2	Повторение. Решение задач на классы	2	1	1	0	Наблюдение, опрос детей, анализ работ
1.3	Повторение. Проектирование классов	2	1	1	0	Наблюдение, опрос детей, анализ работ
1.4	QT 1. Что такое QT и PyQT. Знакомство	2	1	1	0	Наблюдение, опрос детей, анализ работ
1.5	QT 2. QtDesigner, pyuic, два способа подключения uic-файла	2	1	1	0	Наблюдение, опрос детей, анализ работ
1.6	Резерв	3	1	1	1	Наблюдение, опрос детей, анализ работ
1.7	QT 3. Обработка исключений. Создание собственных исключений.	3	1	1	1	Наблюдение, опрос детей, анализ работ
1.8	QT 4. Файлы в Python. Типы файлов и работа с ними. Внутреннее устройство файлов	3	1	1	1	Наблюдение, опрос детей, анализ работ
1.9	Резерв	2	1	1	0	Наблюдение, опрос детей, анализ работ
1.10	QT 5. Диалоги, работа с изображениями	3	1	1	1	Наблюдение, опрос детей, анализ работ
1.11	Самостоятельная работа на файлы	2	0	2	0	Наблюдение, опрос детей,

						анализ работ
1.12	QT 6. Работа с простыми таблицами (csv). Работа с табличными данными в PyQT (начало работы над проектом PyQT)	3	1	1	1	Наблюдение, опрос детей, анализ работ
1.13	QT 7. Введение в БД, работа с SQL-таблицами и отображение данных в PyQT. Часть 1	3	1	1	1	Наблюдение, опрос детей, анализ работ
1.14	QT 8. Введение в БД, работа с SQL-таблицами и отображение данных в PyQT. Часть 2	3	1	1	1	Наблюдение, опрос детей, анализ работ
1.15	QT 9. Обработка клавиатуры и курсора. Сборка независимого приложения	3	1	1	1	Наблюдение, опрос детей, анализ работ
1.16	Резерв (PyQT. Установка дополнительных компонентов. PyQTgraph)	3	1	1	1	Наблюдение, опрос детей, анализ работ
1.17	Резерв	3	1	1	1	Наблюдение, опрос детей, анализ работ
1.18	Резерв	3	1	1	1	Наблюдение, опрос детей, анализ работ
1.19	Резерв	3	1	1	1	Наблюдение, опрос детей, анализ работ
1.20	Защита проекта QT	2	0	2	0	Наблюдение, опрос детей, анализ работ
1.21	Самостоятельная работа на SQL-запросы	2	0	2	0	Наблюдение, опрос детей, анализ работ
Модуль 2. _____ Библиотека PyGame. (наименование модуля)						
Всего: 37 часов (из них: 13 часов – лекционных, 15 часов – практических, 9 часов – самостоятельная работа)						
2.1	Введение в репозитории. Подключение в PyCharm. Работа с удаленным репозиторием	2	1	1	0	Наблюдение, опрос детей, анализ работ
2.2	Резерв (рекомендую доп. тему Основные команды	2	1	1	0	Наблюдение, опрос детей, анализ работ

	при одиночной работе с GIT)					
2.3	PyGame 1. Введение	2	1	1	0	Наблюдение, опрос детей, анализ работ
2.4	PyGame 2. Игровой цикл. События	2	1	1	0	Наблюдение, опрос детей, анализ работ
2.5	PyGame 3. Клетчатое поле	3	1	1	1	Наблюдение, опрос детей, анализ работ
2.6	PyGame 4. Классические игры на клетчатом поле	3	1	1	1	Наблюдение, опрос детей, анализ работ
2.7	Совместная работа над проектом, основные понятия и команды. Работа с репозиториями в среде разработки	3	1	1	1	Наблюдение, опрос детей, анализ работ
2.8	PyGame 5. Изображения. Спрайты	3	1	1	1	Наблюдение, опрос детей, анализ работ
2.9	PyGame 6. Столкновения и другие взаимодействия (начало работы над проектом PyGame)	3	1	1	1	Наблюдение, опрос детей, анализ работ
2.10	PyGame 7. Игра в целом	3	1	1	1	Наблюдение, опрос детей, анализ работ
2.11	PyGame 8. Украшения игры	3	1	1	1	Наблюдение, опрос детей, анализ работ
2.12	Резерв	3	1	1	1	Наблюдение, опрос детей, анализ работ
2.13	Резерв	3	1	1	1	Наблюдение, опрос детей, анализ работ
2.14	Защита проекта PyGame	2	0	2	0	Наблюдение, опрос детей, анализ работ
<b>Модуль 3. _____ Технологии для разработки WEB приложений.</b> (наименование модуля)						
<b>Всего: 77 часов (из них: 27 часов – лекционных, 33 часов – практических, 17 часов – самостоятельная работа)</b>						
3.1.	WEB. Работа с файловой системой и популярными форматами файлов: zip-архивами и json-файлами	2	1	1	0	Наблюдение, опрос детей, анализ работ
3.2.	WEB. Знакомство с API	2	1	1	0	Наблюдение, опрос детей, анализ работ

3.3.	WEB. Работа с командной строкой (скрипты, аргументы). Периодические задачи (модуль schedule)	2	1	1	0	Наблюдение, опрос детей, анализ работ
3.4.	WEB. Библиотека argparse. Задачи на создание скриптов с ее помощью	2	1	1	0	Наблюдение, опрос детей, анализ работ
3.5	WEB. Работа с протоколом HTTP	2	1	1	0	Наблюдение, опрос детей, анализ работ
3.6	WEB. Решение задач на API Яндекс.Карт	2	1	1	0	Наблюдение, опрос детей, анализ работ
3.7	WEB. Введение во flask. Обработка HTML-форм	2	1	1	0	Наблюдение, опрос детей, анализ работ
3.8	WEB. Шаблоны. flask-wtf	2	1	1	0	Наблюдение, опрос детей, анализ работ
3.9	Резерв	2	1	1	0	Наблюдение, опрос детей, анализ работ
3.10	Резерв	2	1	1	0	Наблюдение, опрос детей, анализ работ
3.11	Самостоятельная работа на http, json, и командную строку	2	0	2	0	Наблюдение, опрос детей, анализ работ
3.12	WEB. Знакомство с flask-sqlalchemy	3	1	1	1	Наблюдение, опрос детей, анализ работ
3.13	WEB. Flask-sqlalchemy	3	1	1	1	Наблюдение, опрос детей, анализ работ
3.14	WEB. REST-API. Понятие. Делаем простое Rest-api	3	1	1	1	Наблюдение, опрос детей, анализ работ
3.15	WEB. REST-API. Flask-restful	3	1	1	1	Наблюдение, опрос детей, анализ работ
3.16	WEB. Разворачиваем проект в облаке. Дорешка	3	1	1	1	Наблюдение, опрос детей, анализ работ
3.17	WEB. Урок №1 (Алиса)	3	1	1	1	Наблюдение, опрос детей, анализ работ
3.18	WEB. Урок №2 (Алиса)	3	1	1	1	Наблюдение, опрос детей, анализ работ

3.19	Резерв (начало работы над проектом WEB)	3	1	1	1	Наблюдение, опрос детей, анализ работ
3.20	Итоговая самостоятельная работа	2	0	2	0	Наблюдение, опрос детей, анализ работ
3.21	Резерв (Библиотеки unittest и pytest)	3	1	1	1	Наблюдение, опрос детей, анализ работ
3.22	Резерв (Чат-боты 1 (ВКонтакте))	3	1	1	1	Наблюдение, опрос детей, анализ работ
3.23	Резерв (Чат-боты 2 (Telegram))	3	1	1	1	Наблюдение, опрос детей, анализ работ
3.24	Резерв (Чат-боты 3 (Discord))	3	1	1	1	Наблюдение, опрос детей, анализ работ
3.25	Резерв	3	1	1	1	Наблюдение, опрос детей, анализ работ
3.26	Резерв	3	1	1	1	Наблюдение, опрос детей, анализ работ
3.27	Резерв	3	1	1	1	Наблюдение, опрос детей, анализ работ
3.28	Резерв	3	1	1	1	Наблюдение, опрос детей, анализ работ
3.29	Резерв	3	1	1	1	Наблюдение, опрос детей, анализ работ
3.30	Защита проекта Web	2	0	2	0	Наблюдение, опрос детей, анализ работ
	Итоговая аттестация	в соответствии с нормами времени				
	ИТОГО:	168	58	72	38	



### **3. Рабочие программы учебных предметов, курсов, дисциплин (модулей).**

#### **3.1 Программа модуля 1 «Повторение. Библиотека PyQt для создания GUI».**

##### **Образовательная задача модуля:**

Исключения. Работа с файлами. Файловые форматы: CSV, JSON, XML. Работа с базами данных. 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 форматами. Начинают работу над мини проектами.

В течение модуля ученики решают большое количество задач различной сложности и пишут 2 самостоятельные работы.

**Объем дисциплины (модуля):** 54 часов,

в т.ч. лекций – 18 часов;

практических – 24 часов;

самостоятельной работы 12 – часов.

##### **Содержание модуля:**

**Тема 1.1 Повторение. Решение задач на основные конструкции данных** (лекций 1 час, практических 1 час)

**Лекция:** Повторение основных конструкций данных.

**Практическое занятие:** Решение задач по теме урока.

**Тема 1.2 Повторение. Решение задач на классы** (лекций 1 час, практических 1 час)

**Лекция:** Классы, а именно: создание своих классов, атрибуты (в том числе методы) класса, «магические методы». Наследование, в том числе перегрузка методов и их дополнение с помощью обращения к методам базового класса через `super()`.

**Практическое занятие:** Решение задач по теме урока.

**Тема 1.3 Повторение. Проектирование классов** (лекций 1 час, практических 1 час)

**Лекция:** Повторение классов с акцентом на проектирование.

**Практическое занятие:** Решение задач по теме урока.

**Тема 1.4 QT 1. Что такое QT и PyQt. Знакомство** (лекций 1 час, практических 1 час)

**Лекция:** Рассказывается о графическом интерфейсе и разных способах его реализации, а также начинается знакомство с библиотекой PyQt5.

Разбирается настройка окружения и примеры работы с основными элементами интерфейса

**Практическое занятие:** Решение задач по теме урока.

**Тема 1.5 QT 2. QtDesigner, pyuic, два способа подключения uic-файла** (лекций 1 час, практических 1 час)

**Лекция:** Разбирается популярный способ создания графических интерфейсов — с помощью программы QtDesigner.

**Практическое занятие:** Решение задач по теме урока.

**Тема 1.6 Резерв (дорешка)** (лекций 1 час, практических 1 час)

**Лекция:** Повторение материала пройденных тем, разбор задач.

**Практическое занятие:** Решение не решенных задач по пройденным темам.

**Тема 1.7 QT 3. Обработка исключений. Создание собственных исключений.** (лекций 1 час, практических 1 час)

**Лекция:** Рассказывается о работе с исключениями в современных языках программирования, в частности, в языке Python. Сравнение методики исключений с методикой кодов возврата. Построение собственных классов исключений и их наследование, методики LBYL и EAFP.

**Практическое занятие:** Решение задач по теме урока.

**Тема 1.8 QT 4. Файлы в Python. Типы файлов и работа с ними. Внутреннее устройство файлов** (лекций 1 час, практических 1 час)

**Лекция:** Даются общие сведения о файлах и их хранении в современных ОС. Затрагиваются наиболее общие аспекты работы с файлами в Python: открытие, чтение/запись, закрытие текстовых и бинарных файлов. Обзорно рассмотрены вспомогательные функции («перемотка», работа с кодировками, построчное чтение).

**Практическое занятие:** Решение задач по теме урока.

**Тема 1.9 Резерв (дорешка)** (лекций 1 час, практических 1 час)

**Лекция:** Повторение материала пройденных тем, разбор задач.

**Практическое занятие:** Решение не решенных задач по пройденным темам.

**Тема 1.10 QT 5. Диалоги, работа с изображениями** (лекций 1 час, практических 1 час)

**Лекция:** Рассмотрим возможности PyQT по отображению графических данных, а также изучим возможности рисования непосредственно на виджетах библиотеки.

**Практическое занятие:** Решение задач по теме урока.

**Тема 1.11 Самостоятельная работа на файлы** (лекций 0 час, практических 2 часа)

**Практическое занятие:** Решение задач из самостоятельной работы, которая состоит из трёх задач, которые можно решать в любом порядке.

**Тема 1.12 QT 6. Работа с простыми таблицами (csv). Работа с табличными данными в PyQT (начало работы над проектом PyQT) (лекций 1 час, практических 1 час)**

**Лекция:** Технологии хранения однотипных записей в файлах, форматам с фиксированной и с произвольной длиной записи (DSV, TSV, CSV). Особое внимание уделено форматам с произвольной длиной записи, методам работы с ними при помощи строковых функций, а также специализированной библиотеки csv и PyQT.

**Практическое занятие:** Решение задач по теме урока.

**Тема 1.13 QT 7. Введение в БД, работа с SQL-таблицами и отображение данных в PyQT. Часть 1 (лекций 1 час, практических 1 час)**

**Лекция:** Начнем знакомство с базами данных и языком SQL. Это большая тема, которая будет сопровождать нас до конца обучения, периодически «обрастая» новыми подробностями.

**Практическое занятие:** Решение задач по теме урока.

**Тема 1.14 QT 8. Введение в БД, работа с SQL-таблицами и отображение данных в PyQT. Часть 2 (лекций 1 час, практических 1 час)**

**Лекция:** Продолжим знакомство с базами данных и работой с ними при помощи SQL и Python. Сделаем основной упор на управление данными: запись, изменение и удаление записей.

**Практическое занятие:** Решение задач по теме урока.

**Тема 1.15 QT 9. Обработка клавиатуры и курсора. Сборка независимого приложения (лекций 1 час, практических 1 час)**

**Лекция:** Научимся взаимодействовать с пользователем на новом уровне: добавим в программу работу с клавиатурой и мышкой. Рассмотрим, как собрать нашу программу в exe-файл. А также изучим основы создания SQLite-базы данных.

**Практическое занятие:** Решение задач по теме урока.

**Тема 1.16 Резерв (PyQT. Установка дополнительных компонентов. PyQTgraph) (лекций 1 час, практических 1 час)**

**Лекция:** Повторение материала пройденных тем, разбор задач. / Научимся пользоваться дополнительными компонентами PyQT на примере PyQTGraph — библиотеки для построения графиков. А еще научимся проигрывать музыку в своих приложениях.

**Практическое занятие:** Решение задач по пройденным темам. / Решение задач по теме урока.

**Тема 1.17 Резерв (подготовка проекта, дорешка)** (лекций 1 час, практических 1 час)

**Лекция:** Разбор домашних задач. Консультации по проектам.

**Практическое занятие:** Решение задач по пройденным темам. Доработка проектов.

**Тема 1.18 Резерв (подготовка проекта, дорешка)** (лекций 1 час, практических 1 час)

**Лекция:** Разбор домашних задач. Консультации по проектам.

**Практическое занятие:** Решение задач по пройденным темам. Доработка проектов.

**Тема 1.19 Резерв (подготовка проекта, дорешка)** (лекций 1 час, практических 1 час)

**Лекция:** Разбор домашних задач. Консультации по проектам.

**Практическое занятие:** Решение задач по пройденным темам. Доработка проектов.

**Тема 1.20 Защита проекта QT** (лекций 0 часов, практических 2 часа)

**Практическое занятие:** Защита проектов и их доработка.

**Тема 1.21 Самостоятельная работа на SQL-запросы** (лекций 0 час, практических 2 часа)

**Практическое занятие:** Решение задач из самостоятельной работы, которая состоит из трёх задач, которые можно решать в любом порядке.

**Задания для самостоятельной работы по модулю:** Решение задач по пройденным темам. Подготовка мини-проектов.

**Перечень ресурсов** Учебные материалы и задания по темам на платформе LMS Лицея Академии Яндекса <https://lyceum.yandex.ru/>.

## **3.2 Программа модуля 2 «Библиотека PyGame»**

### **Образовательная задача модуля:**

В этом модуле учащимся предстоит написать игру.

С одной стороны, игры — дело несерьезное. Конференции и конкурсы программных проектов очень редко рассматривают такие работы.

С другой, разработка игр — это сложная задача, включающая в себя много процессов и требующая использовать различные технологии.

Можно утверждать, что любая игра — это Объекты + Графика + Физика + ...

Для того чтобы было удобно программировать все это многообразие, используют фреймворки. Одним из самых известных и популярных фреймворков для разработки игр на Python является PyGame.

Решение задач.

**Объем дисциплины (модуля):** 37 часов,

в т.ч. лекций – 13 часов;  
практических – 15 часов;  
самостоятельной работы – 9 часов.

### **Содержание модуля**

**Тема 2.1 Введение в репозитории. Подключение в PyCharm. Работа с удаленным репозиторием** (лекций 1 час, практических 1 час)

**Лекция:** Рассматриваются цели систем контроля версий, принципы работы с ними и производится разбор системы Git. Научимся создавать локальные репозитории, работать с версиями, подключать удаленный репозиторий, клонировать удаленный репозиторий.

**Практическое занятие:** Решение задач по теме урока.

**Тема 2.2 Резерв (Основные команды при одиночной работе с GIT)** (лекций 1 час, практических 1 час)

**Лекция:** Рассматриваются принципы работы с системой контроля версий Git из командной строки. Мы повторим те же действия, что и на уроках по работе с Git из IDE, но в этот раз сделаем все «руками».

**Практическое занятие:** Решение задач по теме урока.

**Тема 2.3 PyGame 1. Введение** (лекций 1 час, практических 1 час)

**Лекция:** Знакомство с библиотекой PyGame.

**Практическое занятие:** Решение задач по теме урока.

**Тема 2.4 PyGame 2. Игровой цикл. События** (лекций 1 час, практических 1 час)

**Лекция:** Поговорим об игровом цикле. Обсудим работу со временем, кадрами и событиями.

**Практическое занятие:** Решение задач по теме урока.

**Тема 2.5 PyGame 3. Клетчатое поле** (лекций 1 час, практических 1 час)

**Лекция:** Начинаем рассматривать игры на клетчатом поле. Это занятие посвящено идеологии клетчатых игр и реализации простейших алгоритмов.

**Практическое занятие:** Решение задач по теме урока.

**Тема 2.6 PyGame 4. Классические игры на клетчатом поле** (лекций 1 час, практических 1 час)

**Лекция:** Занятие целиком состоит из задач. Цель — решать задачи на алгоритмы из популярных классических игр. Потом, при желании, можно будет доработать какую-то из них до итогового проекта.

**Практическое занятие:** Решение задач по теме урока.

**Тема 2.7 Совместная работа над проектом, основные понятия и команды. Работа с репозиториями в среде разработки (лекций 1 час, практических 1 час)**

**Лекция:** Рассматривается работа с ветками в Git, решение конфликтов изменений, а также основы командной разработки в Git.

**Практическое занятие:** Решение задач по теме урока.

**Тема 2.8 PyGame 5. Изображения. Спрайты (лекций 1 час, практических 1 час)**

**Лекция:** Разговор об изображениях и спрайтах — анимационных элементах игры.

**Практическое занятие:** Решение задач по теме урока.

**Тема 2.9 PyGame 6. Столкновения и другие взаимодействия (начало работы над проектом PyGame) (лекций 1 час, практических 1 час)**

**Лекция:** Так как элементами списка могут быть любые объекты — числа, строки, кортежи, множества и даже другие списки. Рассмотрим подробнее списки, элементами которых являются другие (вложенные) списки.

**Практическое занятие:** Решение задач по теме урока.

**Тема 2.10 PyGame 6. Столкновения и другие взаимодействия (начало работы над проектом PyGame) (лекций 1 час, практических 1 час)**

**Лекция:** Рассмотрим разные виды взаимодействия спрайтов друг с другом: пересечение по прямоугольнику, по окружности, по маске.

**Практическое занятие:** Решение задач по теме урока.

**Тема 2.10 PyGame 7. Игра в целом (лекций 1 час, практических 1 час)**

**Лекция:** Поговорим об игре в целом. Игра — это достаточно большая программа, и важно организовать ее правильно. Сегодня мы будем проектировать заготовку для игры Mario.

**Практическое занятие:** Решение задач по теме урока.

**Тема 2.11 PyGame 8. Украшения игры (лекций 1 час, практических 1 час)**

**Лекция:** Рассмотрим два способа на порядок улучшить впечатление от игры это анимация спрайтов и система частиц.

**Практическое занятие:** Решение задач по теме урока.

**Тема 2.12 Резерв (Цели и подходы к тестированию. Создание «самодельных» тестов (без библиотек)) (лекций 1 час, практических 1 час)**

**Лекция:** Повтор пройденного материала. Подготовка проектов. / Что такое тестирование и зачем оно нужно. Напишем несколько простых тестов и подготовимся к использованию более продвинутых инструментов (библиотек) для тестирования в Python.

**Практическое занятие:** Решение задач по пройденным темам.

**Тема 2.13 Резерв** (лекций 1 час, практических 1 час)

**Лекция:** Повтор пройденного материала. Подготовка проектов.

**Практическое занятие:** Решение задач по пройденным темам.

**Тема 2.14 Защита проекта PyGame** (лекций 0 часов, практических 2 часа)

**Практическое занятие:** Защита проектов и их доработка.

**Задания для самостоятельной работы по модулю:** Решение задач по пройденным темам. Подготовка мини-проектов.

**Перечень ресурсов:** Учебные материалы и задания по темам на платформе LMS Лицея Академии Яндекса <https://lyceum.yandex.ru/>.

### **3.3 Программа модуля 3 «Решение прикладных задач в Python»**

#### **Образовательная задача модуля:**

Мы уже научились создавать программы с графическим пользовательским интерфейсом с использованием виджетов PyQt, работать с базами данных, а также создавать игровые приложения с применением библиотеки PyGame. В этом полугодии мы будем рассматривать еще одну сложную и многообразную тему — работу с сетью Интернет. Периодически мы будем немного отвлекаться непосредственно от самого Интернета и говорить о смежных темах, которые сильно облегчат нам жизнь в дальнейшем.

Решение задач. Подготовка мини проекта.

**Объем дисциплины (модуля):** 77 часов,

в т.ч. лекций – 27 часов;

практических – 33 часа;

самостоятельной работы – 17 часов.

#### **Содержание модуля**

**Тема 3.1 WEB. Работа с файловой системой и популярными форматами файлов: zip-архивами и json-файлами** (лекций 1 час, практических 1 час)

**Лекция:** Рассказывается об основных средствах манипуляции с файлами и папками в Python, а также о работе с файлами специализированных форматов: zip-архивах и JSON.

**Практическое занятие:** Решение задач по теме урока.

**Тема 3.2 WEB. Знакомство с API** (лекций 1 час, практических 1 час)

**Лекция:** Рассказывается о том, что такое API, почему и для чего создается API, как изучать новый API и работать с ним. Разбирается HTTP-API на базе StaticMapsAPI и Geocoder API. Изучается принцип работы с HTTP-API на языке Python.

На этом и последующих уроках для работы всех примеров необходимо подключение к сети Интернет.

**Практическое занятие:** Решение задач по теме урока.

**Тема 3.3 WEB. Работа с командной строкой (скрипты, аргументы).  
Периодические задачи (модуль schedule) (лекций 1 час, практических 1 час)**

**Лекция:** Систематизируем уже накопленные к этому времени знания о командной строке как самом простом, с точки зрения реализации, интерфейсе пользователя, и о создании скриптов для командной строки на языке Python. Разбирается простой способ работы с параметрами командной строкой через список `sys.argv` из модуля `sys`.

**Практическое занятие:** Решение задач по теме урока.

**Тема 3.4 WEB. Библиотека argparse. Задачи на создание скриптов с ее помощью (лекций 1 час, практических 1 час)**

**Лекция:** Продолжим работать с командной строкой и разберем возможности библиотеки `argparse`.

**Практическое занятие:** Решение задач по теме урока.

**Тема 3.5 WEB. Работа с протоколом HTTP (лекций 1 час, практических 1 час)**

**Лекция:** Использование HTTP-API. Показан пример поискового приложения, формирующего запросы к HTTP-API на основании результатов других запросов. Рассматривается API поиска по организациям с примерами задач.

**Практическое занятие:** Решение задач по теме урока.

**Тема 3.6 WEB. Решение задач на API Яндекс.Карт (лекций 1 час, практических 1 час)**

**Лекция:** Знакомство с общими принципами работы со сторонними API и концентрация внимания на Яндекс.Картах.

**Практическое занятие:** Решение задач по теме урока.

**Тема 3.7 WEB. Введение во flask. Обработка HTML-форм (лекций 1 час, практических 1 час)**

**Лекция:** Будем использовать не чужие веб-приложения, а писать свои. И уже очень скоро мы убедимся, что с помощью Python это делать достаточно просто (особенно с использованием дополнительных сторонних библиотек). Рассмотрим микрофреймворк Flask.

**Практическое занятие:** Решение задач по теме урока.

**Тема 3.8 WEB. Шаблоны. flask-wtf (лекций 1 час, практических 1 час)**

**Лекция:** Избавимся от необходимости создавать HTML-разметку непосредственно в коде, а также рассмотрим возможность создания форм с использованием объектного подхода с помощью библиотеки `flask-wtf`.

**Практическое занятие:** Решение задач по теме урока.

**Тема 3.9 Резерв (дорешка) (лекций 1 час, практических 1 час)**



**Лекция:** Повтор пройденного материала, разбор с учениками задач из классных и домашних работ.

**Практическое занятие:** Решение задач по пройденным темам

**Тема 3.10 Резерв (дорешка/ подготовка к СР) (лекций 1 час, практических 1 час)**

**Лекция:** Повтор пройденного материала, разбор с учениками задач из классных и домашних работ.

**Практическое занятие:** Решение задач по пройденным темам.

**Тема 3.11 Самостоятельная работа на http, json, и командную строку (лекций 0 час, практических 2 часа)**

**Практическое занятие:** Решение задач из самостоятельной работы, которая состоит из трёх задач, которые можно решать в любом порядке.

**Тема 3.12 WEB. Знакомство с flask-sqlalchemy (лекций 1 час, практических 1 час)**

**Лекция:** Что такое ORM (Object-Relational Mapping — объектно-реляционное отображение). Создадим полноценное веб-приложение с регистрацией и авторизацией пользователей, работающее с базой данных через ORM.

**Практическое занятие:** Решение задач по теме урока.

**Тема 3.13 WEB. Flask-sqlalchemy (лекций 1 час, практических 1 час)**

**Лекция:** Доделаем процесс авторизации пользователей в веб-приложении, а также рассмотрим еще несколько важных моментов, связанных с работой с библиотеками flask и sqlalchemy.

**Практическое занятие:** Решение задач по теме урока.

**Тема 3.14 WEB. REST-API. Понятие. Делаем простое Rest-api (лекций 1 час, практических 1 час)**

**Лекция:** Научимся создавать собственное API для наших веб-приложений.

**Практическое занятие:** Решение задач по теме урока.

**Тема 3.15 WEB. REST-API. Flask-restful (лекций 1 час, практических 1 час)**

**Лекция:** Как создавать RESTful-сервисы с помощью библиотеки flask-restful.

**Практическое занятие:** Решение задач по теме урока.

**Тема 3.16 WEB. Разворачиваем проект в облаке. Дорешка (лекций 1 час, практических 1 час)**

**Лекция:** Как разместить свой проект в Интернете, чтобы к нему могли получить доступ все желающие, а также изучим несколько сервисов, которые смогут нам в этом помочь.

**Практическое занятие:** Решение задач по пройденным темам.

**Тема 3.17 WEB. Урок №1 (Алиса)** (лекций 1 час, практических 1 час)

**Лекция:** Работа с голосовым помощником от Яндекса — Алисой.

**Практическое занятие:** Решение задач по теме урока.

**Тема 3.18 WEB. Урок №2 (Алиса)** (лекций 1 час, практических 1 час)

**Лекция:** Продолжим разрабатывать навыки для Алисы: научим ее знакомиться с пользователем и взаимодействовать с внешними API. Как разместить свой навык в Яндекс.Облаке.

**Практическое занятие:** Решение задач по теме урока.

**Тема 3.19 Резерв (начало работы над проектом WEB)** (лекций 1 час, практических 1 час)

**Лекция:** Повтор пройденного материала. Начало работы над проектом WEB.

**Практическое занятие:** Решение задач по пройденным темам.

**Тема 3.20 Итоговая самостоятельная работа** (лекций 0 часов, практических 2 часа)

**Практическое занятие:** Решение задач из самостоятельной работы, которая состоит из пяти задач, которые можно решать в любом порядке.

**Тема 3.21 Резерв (Введение в асинхронное программирование)** (лекций 1 час, практических 1 час)

**Лекция:** Что такое асинхронное программирование, чем оно отличается от синхронного, как писать асинхронные программы на python.

**Практическое занятие:** Решение задач по теме урока.

**Тема 3.22 Резерв (Библиотеки unittest и pytest)** (лекций 1 час, практических 1 час)

**Лекция:** Рассмотрим инструменты языка Python для тестирования: библиотеки unittest и pytest. С их помощью мы научимся писать гибкие и автоматизированные тесты.

**Практическое занятие:** Решение задач по теме урока.

**Тема 3.23 Резерв (Чат-боты 1 (ВКонтакте))** (лекций 1 час, практических 1 час)

**Лекция:** Поговорим о создании чат-ботов для различных популярных платформ и начнем, пожалуй, с самой популярной социальной сети в России и странах СНГ — vk.com.

**Практическое занятие:** Решение задач по теме урока.

**Тема 3.24 Резерв (Чат-боты 2 (Telegram))** (лекций 1 час, практических 1 час)

**Лекция:** Рассказывается про реализацию telegram API в виде библиотеки на примере telegram-bot-API. Мы будем учиться на конкретных задачах: разберем устройство telegram-бота и узнаем об основных возможностях API, создав несколько простых ботов.

**Практическое занятие:** Решение задач по теме урока.

**Тема 3.25 Резерв (Чат-боты 3 (Discord))** (лекций 1 час, практических 1 час)

**Лекция:** Повтор пройденного материала. Разбор с учениками задач из классных и домашних работ по темам из будущей СР. / Этот урок посвящен вычислительным мощностям Python и библиотеке Numpy. Научимся замерять время выполнения программы. Затронем работу с многомерными массивами и изучим (или вспомним) немного линейной алгебры. А еще, возможно, напишем игру.

**Практическое занятие:** Решение задач по теме урока.

**Тема 3.26 Резерв (Чат-боты 3 (Discord))** (лекций 1 час, практических 1 час)

**Лекция:** Создадим бота для Discord, а также немного поговорим о таком понятии, как параллельное программирование.

**Практическое занятие:** Решение задач по теме урока.

**Тема 3.27 Резерв** (лекций 1 час, практических 1 час)

**Лекция:** Повтор пройденного материала. Подготовка мини проекта.

**Практическое занятие:** Решение задач по пройденным темам.

**Тема 3.28 Резерв** (лекций 1 час, практических 1 час)

**Лекция:** Повтор пройденного материала. Подготовка мини проекта.

**Практическое занятие:** Решение задач по пройденным темам.

**Тема 3.29 Резерв** (лекций 1 час, практических 1 час)

**Лекция:** Повтор пройденного материала. Подготовка мини проекта.

**Практическое занятие:** Решение задач по пройденным темам.

**Тема 3.30 Защита проекта Web** (лекций 0 часов, практических 2 часа)

**Практическое занятие:** Защита проектов и их доработка.

**Задания для самостоятельной работы по модулю:** Решение задач по пройденным темам.

**Перечень ресурсов** Учебные материалы и задания по темам на платформе LMS Лицея Академии Яндекса <https://lyceum.yandex.ru/>.

#### 4. Календарный учебный график дополнительной общеобразовательной общеразвивающей программы «Основы промышленного программирования»

Начало и окончание реализации программы: сентябрь – апрель.

Каникулы: не предусмотрены.

Продолжительность занятий: 45 минут

Продолжительность перемен: 10 минут

Образовательная недельная нагрузка на обучающихся: ~ 5 часов.

Наименование модуля (раздела) / темы	Неделя	Кол-во часов
<b>Модуль 1. «Повторение. Библиотека PyQt для создания GUI»</b>		
Тема 1.1 Повторение. Решение задач на основные конструкции данных	1	2
Тема 1.2 Повторение. Решение задач на классы	1	2
Тема 1.3 Повторение. Проектирование классов	2	2
Тема 1.4 QT 1. Что такое QT и PyQt. Знакомство	2	2
Тема 1.5 QT 2. QtDesigner, pyuic, два способа подключения uic-файла	3	2
Тема 1.6 Резерв	3	3
Тема 1.7 QT 3. Обработка исключений. Создание собственных исключений.	4	3
Тема 1.8 QT 4. Файлы в Python. Типы файлов и работа с ними. Внутреннее устройство файлов	4	3
Тема 1.9 Резерв	5	2
Тема 1.10 QT 5. Диалоги, работа с изображениями	5	3
Тема 1.11 Самостоятельная работа на файлы	6	2
Тема 1.12 QT 6. Работа с простыми таблицами (csv). Работа с табличными данными в PyQt (начало работы над проектом PyQt)	6	3

Наименование модуля (раздела) / темы	Неделя	Кол-во часов
Тема 1.13 QT 7. Введение в БД, работа с SQL-таблицами и отображение данных в PyQT. Часть 1	7	3
Тема 1.14 QT 8. Введение в БД, работа с SQL-таблицами и отображение данных в PyQT. Часть 2	7	3
Тема 1.15 QT 9. Обработка клавиатуры и курсора. Сборка независимого приложения	8	3
Тема 1.16 Резерв (PyQT. Установка дополнительных компонентов. PyQTgraph)	8	3
Тема 1.17 Резерв	9	3
Тема 1.18 Резерв	9	3
Тема 1.19 Резерв	10	3
Тема 1.20 Защита проекта QT	10	2
Тема 1.21 Самостоятельная работа на SQL-запросы	11	2
<b>Модуль 2. «Библиотека PyGame»</b>		
Тема 2.1 Введение в репозитории. Подключение в PyCharm. Работа с удаленным репозиторием	11	2
Тема 2.2 Резерв (Основные команды при одиночной работе с GIT)	12	2
Тема 2.3 PyGame 1. Введение	12	2
Тема 2.4 PyGame 2. Игровой цикл. События	13	2
Тема 2.5 PyGame 3. Клетчатое поле	13	3
Тема 2.6 PyGame 4. Классические игры на клетчатом поле	14	3
Тема 2.7 Совместная работа над проектом, основные понятия и команды. Работа с репозиториями в среде разработки	14	3
Тема 2.8 PyGame 5. Изображения. Спрайты	15	3
Тема 2.9 PyGame 6. Столкновения и другие взаимодействия (начало работы над проектом PyGame)	15	3
Тема 2.10 PyGame 7. Игра в целом	16	3
Тема 2.11 PyGame 8. Украшения игры	16	3
Тема 2.12 Резерв	17	3
Тема 2.13 Резерв	17	3

Наименование модуля (раздела) / темы	Неделя	Кол-во часов
Тема 2.14 Защита проекта PyGame	18	2
<b>Модуль 3. «Технологии для разработки WEB приложений»</b>		
Тема 3.1 WEB. Работа с файловой системой и популярными форматами файлов: zip-архивами и json-файлами	18	2
Тема 3.2 WEB. Знакомство с API	19	2
Тема 3.3 WEB. Работа с командной строкой (скрипты, аргументы). Периодические задачи (модуль schedule)	19	2
Тема 3.4 WEB. Библиотека argparse. Задачи на создание скриптов с ее помощью	20	2
Тема 3.5 WEB. Работа с протоколом HTTP	20	2
Тема 3.6 WEB. Решение задач на API Яндекс.Карт	21	2
Тема 3.7 WEB. Введение во flask. Обработка HTML-форм	21	2
Тема 3.8 WEB. Шаблоны. flask-wtf	22	2
Тема 3.9 Резерв	22	2
Тема 3.10 Резерв	23	2
Тема 3.11 Самостоятельная работа на http, json, и командную строку	23	2
Тема 3.12 WEB. Знакомство с flask-sqlalchemy	24	3
Тема 3.13 WEB. Flask-sqlalchemy	24	3
Тема 3.14 WEB. REST-API. Понятие. Делаем простое Rest-api	25	3
Тема 3.15 WEB. REST-API. Flask-restful	25	3
Тема 3.16 WEB. Разворачиваем проект в облаке. Дорешка	26	3
Тема 3.17 WEB. Урок №1 (Алиса)	26	3
Тема 3.18 WEB. Урок №2 (Алиса)	27	3
Тема 3.19 Резерв (начало работы над проектом WEB)	27	3
Тема 3.20 Итоговая самостоятельная работа	28	2
Тема 3.21 Резерв (Введение в асинхронное программирование)	28	3
Тема 3.22 Резерв (Библиотеки unittest и pytest)	29	3
Тема 3.23 Резерв (Чат-боты 1 (ВКонтакте))	29	3

Наименование модуля (раздела) / темы	Неделя	Кол-во часов
Тема 3.24 Резерв (Чат-боты 2 (Telegram))	30	3
Тема 3.25 Резерв (Чат-боты 3 (Discord))	30	3
Тема 3.26 Резерв	31	3
Тема 3.27 Резерв	31	3
Тема 3.28 Резерв	32	3
Тема 3.29 Резерв	32	3
Тема 3.30 Защита проекта Web	33	2

## 5. Организационно-педагогические условия реализации программы

### 5.1 Кадровое обеспечение

Дается характеристика кадрового обеспечения образовательного процесса в рамках ДООП, соответствие требованиям по профессии.

Название дисциплины / модуля / практики	ФИО преподавателя	Квалификация преподавателей (образование, ученая степень, ученое звание, награды, звания); квалификация преподавателей, привлекаемых к проведению занятий	Опыт профессиональной деятельности (преподавательской деятельности) (стаж работы)
«Основы промышленного программирования»	Попов Александр Александрович	Образование – высшее, инженер по специальности промышленная электроника. Педагог дополнительного образования детей ДНК им. Академика Е. М. Дианова, инженер научно-исследовательской лаборатории вентильных электрических машин, младший научный сотрудник Института корпоративного обучения и непрерывного образования	5



## 5.2 Учебно-методическое и информационное обеспечение

### Основная

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

5. М. Лутц. Изучаем Python. СПб.: Символ-Плюс, 2011.
6. 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».

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

### 5.3 Материально-техническое обеспечение

Наименование специализированных учебных помещений	Вид занятий	Наименование оборудования, программного обеспечения
1	2	3
Аудитория 401, 201	Лекция, практика	Ноутбуки с доступом к сети интернет, операционная система Windows, программное обеспечение PyCharm, наличие доступа к сети Интернет, наличие кабинета

## **6. Нормативно-методическое обеспечение системы оценки качества освоения обучающимися дополнительной общеобразовательной общеразвивающей программы**

### **6.1 Формы аттестации (указываются формы промежуточной и итоговой аттестации в соответствии с учебным планом)**

Название	Форма аттестации
Модуль 1 «Повторение. Библиотека PyQT для создания GUI»	Самостоятельные работы, Мини проект
Модуль 2 «Библиотека PyGame»	Самостоятельные работы, Мини проект
Модуль 3 «Технологии для разработки WEB приложений»	Самостоятельные работы, Мини проект
Итоговая аттестация	Сумма баллов за курс

### **6.2 Оценочные материалы**

#### **6.2.1 Комплект оценочных средств для проведения промежуточной аттестации**

Модуль	Перечень контрольных заданий (вопросов, практических заданий, задач, ситуаций, тестовых заданий и др. )	Критерии оценки
Модуль 1 «Введение в программирование»	3-4 задачи на пройденные темы модуля, с 2-3 тестами для самостоятельной проверки правильности решения.	Проверка программного кода на соответствие стандарту PEP8, количеству пройденных автотестов в тестирующей системе.
Модуль 2 «Базовые конструкции в Python»	3-4 задачи на пройденные темы модуля, с 2-3 тестами для	Проверка программного кода на

	самостоятельной проверки правильности решения.	соответствие стандарту PEP8, количеству пройденных автотестов в тестирующей системе.
Модуль 3 «Решение прикладных задач в Python»	3-4 задачи на пройденные темы модуля, с 2-3 тестами для самостоятельной проверки правильности решения.	Проверка программного кода на соответствие стандарту PEP8, количеству пройденных автотестов в тестирующей системе.

## 6.2.2 Комплект оценочных средств для проектной деятельности

### Общие сведения.

Во время обучения в Лицее Академии Яндекса вам предстоит написать несколько проектов. Вы можете выполнять их самостоятельно или в группе (не больше 2-х человек).

Как результат выполнения проекта должно быть представлено:

- Задание проекта — его следует согласовать с преподавателем до начала работы;
- Программный код с необходимыми комментариями, пояснениями и ресурсами;
- Пояснительную записку — документ, описывающий суть проекта, его особенности и прочие моменты;
- Презентацию — если вы проводите защиту проекта;
- Видео — при необходимости, например, для проекта PyGame.

Оценка за проект учитывает как саму реализацию задачи, так и оформление и представление проекта.

## Оценка проекта.

Критерий	Оценка	Комментарий
Объём программного кода	15	15 баллов даётся за 500 и более строк. 12 — за 300. Далее пропорционально. Учитывается только содержательный код, в оценку не входят, например, карты уровней, если они включены в код.
Чистота кода	5	Оценивается PEP 8, константы в коде, сору-paste, говорящие имена переменных и так далее.
Качество проектирования	20	Используются ли классы, модули и прочее. Если код написан очень примитивно, то оценка снижается.
Применённые технологии	20	Реализованы ли практически все изученные в модуле технологии.
Оригинальная идея	10	
Соблюдение сроков	15	Соблюдение сроков каждого этапа проекта. Даже учитывая сжатые сроки, просим вас разумно распределить своё время для работы над проектом, не откладывая всё на последний день.
Защита	10	Качество защиты проекта. Допуск к защите возможен только при условии выполнения всех этапов проекта до даты защиты.
Премия	5	
Работоспособность	[0.25, 0.5, 0.75, 1]	Значение коэффициента выбирается из предложенного списка в зависимости от того, работает ли проект, реализует ли все заявленные функции и т.д.

### 6.2.3 Комплект оценочных средств для проведения итоговой аттестации

Рейтинг — это показатель успешности ученика в Лицее Академии Яндекса. Рейтинг обновляется с каждой зачтенной задачей. Стоимость задачи назначает методист курса.

Максимальный рейтинг — 120 баллов. Чтобы перейти на второй курс, нужно набрать 45 и более баллов.

Соответствие рейтинга и оценки:

Рейтинг	Оценка
<45	Удовлетворительно
45–79	Хорошо
80–99	Отличник
100 и больше	Суперотличник

**Как вычисляется рейтинг:**

**Шаг 1. Устанавливаются максимально возможные баллы за каждый тип задач:**

Все задачи курса делятся на пять типов: домашние, дополнительные и задачи классных, самостоятельных, контрольных работ.

Максимальные баллы:

- 10 баллов за решение всех задач классных работ ( $MAX_{кл}$ ),
- 10 баллов — домашних задач ( $MAX_{дом}$ )
- 40 баллов — дополнительных задач ( $MAX_{доп}$ )
- 40 баллов — задач контрольных работ ( $MAX_{контр}$ )
- 20 баллов — задач самостоятельных работ ( $MAX_{сам}$ ).



Решение дополнительных задач, задач самостоятельных и контрольных работ вносит самый большой вклад в рейтинг:

## Шаг 2. Вычисляется количество уроков с задачами каждого типа

В курсах может быть разное количество задач каждого типа. Например, на первом курсе основного образования ученики решают четыре самостоятельные работы, а на первом курсе дополнительного образования — три.


Поэтому для каждого типа задач рассчитывается  $K_{min}$  — количество уроков курса, в которых есть хотя бы одна задача данного типа.

Например,  $K_{min} = 2$  означает, что в курсе есть два урока хотя бы с одной задачей классной работы.

## Шаг 3. Суммируются первичные баллы решенных задач

Первичные баллы — это баллы, указанные в тестирующей системе для каждой задачи.

# Попугай


 Класная работа


макс. 16 балл.




Дедлайн: 1 окт. 23:30

Условие

ID 32524072  
25 апр. 12:57


 Доработать  
25 апр. 12:58

Отправить решение 


```
1 print('Моя первая программа напечатала "Привет, Яндекс!" :')
2 sdfsd
3 file()sdf
4
5 sdf
6 sd
7 file()sd
8 file()sdf
9
10 sdf
```

История решений



ID 21031080 


Отправлено на проверку

Дедлайн: 1 окт. 23:30

ID 32524072 

Отправлено на проверку

 Вердикт: ошибка оформления кода. Подробнее 

 Статус изменен: Доработать

Введем обозначения:

- $ПБ_{кл} = \sum ПБ_{РешенныхКлассных}$  ,
- $ПБ_{дом} = \sum ПБ_{РешенныхДомашних}$  ,
- $ПБ_{доп} = \sum ПБ_{РешенныхДополнительных}$  ,
- $ПБ_{контр} = \sum ПБ_{РешенныхКонтрольных}$  ,
- $ПБ_{сам} = \sum ПБ_{РешенныхСамостоятельных}$  .

За выполнение всех задач одного типа дается максимум 100 первичных баллов. Например, если ученик решит все домашние задачи урока, то  $ПБ_{дом} = 100$ .

#### **Шаг 4. Вычисляются вторичные баллы**

Вторичные баллы вычисляются по формуле:

$$ВБ_{ТипЗадачи} = \frac{ПБ_{ТипЗадачи}}{100} \cdot \frac{МАХ_{ТипЗадачи}}{K_{ТипЗадачи}}$$

Первый множитель в формуле — доля набранных первичных баллов из максимально возможных. Второй множитель — стоимость одного урока в баллах.

Введем обозначения:

- $ВБ_{кл} = \frac{ПБ_{кл}}{100} \cdot \frac{МАХ_{кл}}{K_{кл}}$
- $ВБ_{дом} = \frac{ПБ_{дом}}{100} \cdot \frac{МАХ_{дом}}{K_{дом}}$
- $ВБ_{доп} = \frac{ПБ_{доп}}{100} \cdot \frac{МАХ_{доп}}{K_{доп}}$
- $ВБ_{контр} = \frac{ПБ_{контр}}{100} \cdot \frac{МАХ_{контр}}{K_{контр}}$
- $ВБ_{сам} = \frac{ПБ_{сам}}{100} \cdot \frac{МАХ_{сам}}{K_{сам}}$

#### **Шаг 5. Вторичные баллы суммируются**

Рейтинг вычисляется по формуле:

$$P = ВБ_{кл} + ВБ_{дом} + ВБ_{доп} + ВБ_{контр} + ВБ_{сам}$$



## 7. Сведения об обновлении программы

Программа обновлена решением Ученого совета Университета:

№	Прилагаемый к ДООП документ, содержащий текст обновления	Решение об обновлении ДООП	
		дата	протокол №
1.	Приложение № 1	____. ____ 20__ г.	
2.	Приложение № 2	____. ____ 20__ г.	
3.	Приложение № 3	____. ____ 20__ г.	
4.	Приложение № 4	____. ____ 20__ г.	