



Государственное бюджетное учреждение культуры
Иркутская областная государственная универсальная научная библиотека
им. И. И. Молчанова-Сибирского

Принята на заседании Методического
совета ИОГУНБ им. И. И. Молчанова-
Сибирского от «07» августа 2024 г.
Протокол № 7

Утверждена приказом ГБУК ИОГУНБ
от «15» августа 2024 г.
№ 157-д

ДОПОЛНИТЕЛЬНАЯ ОБЩЕОБРАЗОВАТЕЛЬНАЯ ПРОГРАММА

Основы промышленного программирования
(наименование программы)

Направленность: техническая
Возраст обучающихся: 14 – 19 лет
Срок реализации программы: 1 год
Форма обучения: очная
ФИО, должность автора:
М.Г. Рейнгольд, ведущий методист СУЦ НМО
А.В. Осипова, ведущий методист СУЦ НМО
А.В. Васильев, ведущий методист СУЦ НМО

ОГЛАВЛЕНИЕ

1.	Пояснительная записка	3
2.	Содержание программы	8
3.	Методическое обеспечение	23
4.	Организационно-педагогические условия реализации дополнительной общеразвивающей программы	24
5.	Список литературы	26

1. ПОЯСНИТЕЛЬНАЯ ЗАПИСКА

1.1. Общая характеристика дополнительной общеразвивающей программы

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

В обязательной школьной программе информатики программирование нередко представлено лишь на элементарном уровне, на это выделяется недостаточное количество часов. Лишь немногие школы могут себе позволить преподавать программирование на достойном уровне. Следствием этого является формальное восприятие учащимися основ современного программирования и неумение применять полученные знания на практике.

Изучение основных принципов программирования невозможно без регулярной практики написания программ на каком-либо языке. 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 выучивание 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 до 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 конструкции позволит учащимся потом с легкостью выучить любой другой язык программирования.

Содержание дополнительной общеобразовательной программы «Основы промышленного программирования» (далее ДОП, общеобразовательная программа) включает 3 модуля:

1. модуль № 1. Программирование на Python.
2. модуль № 2. Тестирование программ и командные методы работы.
3. модуль № 3. Проекты GUI, Игры и API

В основу программы 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: обучающиеся учатся решать задачи без помощи преподавателя. Для этого в содержании программы 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 обучающемуся необходимо самостоятельно формализовать его (или задать правильные вопросы педагогу).

ДОП «Основы промышленного программирования» рассчитан на 112 учебных часов и 56 часов самостоятельной работы и предназначен для учеников 8-11 классов школ разного уровня подготовки, а также студентов 1-2 курсов средних специальных учебных заведений.

1.2. Нормативные документы, регламентирующие разработку дополнительной образовательной программы

Нормативно–правовую базу разработки ДОП составляют:

- Федеральный закон от 29.12.2012 № 273–ФЗ «Об образовании в Российской Федерации»;
- приказ Министерства просвещения Российской Федерации от 27.12.2022 № 629 «Об утверждении порядка организации и осуществления образовательной деятельности по дополнительным общеобразовательным программам»;
- письмо Минобрнауки России от 18.11.2015 № 09–3242 «О направлении информации» (вместе с «Методическими рекомендациями по проектированию дополнительных общеразвивающих программ (включая разноуровневые программы)»);
- локальные нормативные акты ГБУК ИОГУНБ, регламентирующие образовательную деятельность по дополнительным образовательным программам.

1.3. Актуальность программы

Знания и умения, приобретенные в результате освоения программы, 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.

Новизна данной образовательной программы:

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

Практическая значимость программы заключается в том, что она способствует более успешному овладению знаниями и умениями в области программирования через развитие самостоятельности обучающихся и оптимизацию средств и методов обучения.

1.4. Направленность программы: техническая.

1.5. Объемы и сроки освоения программы

Форма обучения: очная

Срок реализации программы: 1 год

Объем программы: 168 академических часа

Адресат программы: учащиеся 14-19 лет общеобразовательных организаций и организаций среднего профессионального образования.

Максимальная недельная нагрузка: 4 академических часа в неделю.

Режим занятий: два раза в неделю по 2 академических часа.

Продолжительность академического часа – 45 минут. Между сдвоенными занятиями предусмотрен перерыв 10 минут.

1.6. Цель, задачи и планируемые результаты освоения программы

Цель: создание условий для изучения основ промышленного программирования;

рассмотрение различных парадигм программирования, предлагаемых этим языком (процедурная, функциональная, объектно-ориентированная); подготовка к использованию как языка программирования, так и методов программирования на Python в учебной и последующей профессиональной деятельности в различных предметных областях.

Задачи:

- 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.

Программа направлена на формирование (совершенствование) следующих компетенций:

- 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.

1.7. Личностные, метапредметные результаты освоения программы:

Личностные результаты:

- формирование ответственного отношения к учению, способности довести до конца начатое дело аналогично завершенным творческим учебным проектам;
- формирование способности к саморазвитию и самообразованию средствами информационных технологий на основе приобретенной 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, участие 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 ее 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, умение составить и записать алгоритм для конкретного исполнителя, знакомство с основными алгоритмическими структурами – линейной, условной и циклической;
- умение выполнять пошагово (с использованием компьютера или вручную) несложные алгоритмы управления исполнителями и анализа числовых и текстовых данных;
- навыки и опыт разработки программ в выбранной среде программирования, включая тестирование и отладку программ; умение использовать основные управляющие конструкции объектно-ориентированного программирования и библиотеки прикладных программ, выполнять созданные программы;
- умение разрабатывать и использовать компьютерно-математические модели, оценивать числовые параметры моделируемых объектов и процессов, интерпретировать результаты, получаемые в ходе моделирования реальных процессов, анализировать готовые модели на предмет соответствия реальному объекту или процессу;
- формирование умения соблюдать нормы информационной этики и права.

Требования к результатам освоения программы

В результате освоения программы обучающийся должен знать о:

- основах современных языков программирования;
- возможностях нахождения, оценивания и использования информации из различных источников, необходимой для решения профессиональных задач (в том числе на основе системного подхода);
- библиотеках PyQT, Pygame, а также фреймворке Flask;
- том, как работает протокол HTTP;

- основных SQL-запросах;
- обработке исключений, создании собственных исключений.

В результате освоения программы обучающийся должен уметь:

- объяснять и использовать на практике как простые, так и сложные структуры данных и конструкции для работы с ними;
- искать и обрабатывать ошибки в коде;
- разбивать решение задачи на подзадачи;
- писать грамотный, красивый код;
- анализировать как свой, так и чужой код;
- грамотно строить коммуникацию, исходя из целей и ситуации;
- создавать приложения с графическим интерфейсом; работать в команде;
- использовать базы данных в проектах.

1.8. Методы организации учебного процесса

При организации занятий по ДОП 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.

1.9. Формы и методы контроля:

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

1.10. Документ о квалификации

Контроль качества освоения Программы включает в себя промежуточную и итоговую аттестацию обучающихся.

По окончании обучения при условии успешного освоения Программы выдается сертификат о прохождении обучения.

Лицам, не прошедшим итоговую аттестацию, а также лицам, освоившим часть Программы и отчисленным из ГБУК ИОГУНБ, выдается справка об обучении или периоде обучения по установленному образцу.

2. СОДЕРЖАНИЕ ПРОГРАММЫ

2.1. Учебный план

№	Наименование и содержание учебного модуля	Обязательные учебные занятия, ч			Самостоятельная подготовка	Формы промежуточной аттестации
		всего часов	в том числе			
			лекций	практические занятия		
1.	Модуль № 1. Программирование на Python	12	3	9		
1.1.	Тема 1. Повторение. Решение задач на основные конструкции данных	4	1	3		
1.2.	Тема 2. Повторение. Решение задач на классы	4	1	3		
1.3.	Тема 3. Повторение. Проектирование классов	4	1	3		
2.	Модуль № 2. Тестирование программ и командные методы работы. PyQT	49	15	34		Защита проекта и контрольная работа
2.1.	Тема 1. QT 1. Что такое QT и PyQT. Знакомство	3	1	2		
2.2.	Тема 2. QT 2. QtDesigner, ruic, два способа подключения uic файла	3	1	2		
2.3.	Тема 3. PyQT. Установка дополнительных компонентов. PyQTgraph	3	1	2		
2.4.	Тема 4. QT 3. Обработка исключений. Создание собственных исключений	3	1	2		
2.5.	Тема 5. QT 4. Файлы в Python. Типы файлов и работа с ними. Внутреннее устройство Файлов	3	1	2		
2.6.	Тема 6. QT 5. Диалоги, работа с изображениями	3	1	2		
2.7.	Тема 7. Контрольная работа на файлы	2		2		
2.8.	Тема 8. QT 6. Работа с простыми таблицами (csv). Работа с табличными	3	1	2		

	данными вPyQT					
2.9	Тема 9. QT 7. Введение в БД, работа с SQL-таблицами и отображение данных в PyQT. Часть 1	3	1	2		
2.10	Тема 10. QT 8. Введение в БД, работа с SQL - таблицами и отображение данных вPyQT. Часть 2	3	1	2		
2.11	Тема 11. QT 9. Обработка клавиатуры и курсора. Сборка независимого приложения	3	1	2		
2.12	Тема 12. Защита проекта QT	2		2		
2.13	Тема 13. Основные команды при одиночной работе с GIT	3	1	2		
2.14	Тема 14. Цели и подходы к тестированию. Создание «самодельных» тестов (без библиотек)	3	1	2		
2.15	Тема 15. Контрольная работа на SQL-запросы	2		2		
2.16	Тема 16. Введение в репозитории. Подключение в PyCharm. Работа с удаленным репозиторием	3	1	2		
3.	Модуль № 3. Проекты Игры и API	79	26	53		Защита проекта
3.1	Тема 1. PyGame 1. Введение	3	1	2		
3.2	Тема 2. PyGame 2. Игровой цикл. События	3	1	2		
3.3	Тема 3. PyGame 3. Клетчатое поле	3	1	2		
3.4	Тема 4. PyGame 4. Классические игры на клетчатом поле	3	1	2		
3.5	Тема 5. Совместная работа над проектом, основные понятия и команды. Работа с репозиториями в среде разработки	3	1	2		
3.6	Тема 6. PyGame 5. Изображения. Спрайты	3	1	2		
3.7	Тема 7. PyGame 6. Столкновения и другие взаимодействия	3	1	2		
3.8	Тема 8. PyGame 7. Игра в	3	1	2		

	целом					
3.9	Тема 9. PyGame 8. Украшения игры	3	1	2		
3.10	Тема 10. Защита проекта PyGame	2		2		
3.11	Тема 11. WEB. Работа с файловой системой и популярными форматами файлов: zip- архивами и json-файлами	3	1	2		
3.12	Тема 12. WEB. Знакомство с API	3	1	2		
3.13	Тема 13. WEB. Работа с командной строкой (скрипты, аргументы). Периодические задачи (модуль schedule)	3	1	2		
3.14	Тема 14. WEB. Библиотека argparse. Задачи на создание скриптов с ее помощью	3	1	2		
3.15	Тема 15. WEB. Работа с протоколом HTTP	3	1	2		
3.16	Тема 16. WEB. Решение задач на API Яндекс.Карт	3	1	2		
3.17	Тема 17. Контрольная работа на http, json, и командную строку	2		2		
3.18	Тема 18. WEB. Введение во flask. Обработка HTML-форм	3	1	2		
3.19	Тема 19. WEB. Шаблоны. flask-wtf	3	1	2		
3.20	Тема 20. WEB. Знакомство с flask-sqlalchemy	3	1	2		
3.21	Тема 21. WEB. Flask-sqlalchemy	3	1	2		
3.22	Тема 22. WEB. REST-API. Понятие. Делаем простое Rest-api	3	1	2		
3.23	Тема 23. WEB. REST-API. Flask-restful	3	1	2		
3.24	Тема 24. WEB. Разворачиваем проект в облаке. Дорешка	3	1	2		
	Итоговая аттестация. Защита проекта WEB	2		2		2
Итого		142	44	98		
*Промежуточная аттестация проводится за счет времени, отведенного на изучение модуля						

**2.2. Календарный учебный график
Для 1 группы:**

№	Наименование и содержание учебного модуля	День для 1 группы	Обязательные учебные занятия, ч		Самостоятельная работа
			лекции	практические занятия	
1.	Модуль № 1. Программирование на Python		3	9	
1.1.	Тема 1. Повторение. Решение задач на основные конструкции данных	09.09 11.09	1	3	
1.2	Тема 2. Повторение. Решение задач на классы	16.09. 18.09	1	3	
1.3	Тема 3. Повторение. Проектирование классов	23.09 25.09	1	3	
2.	Модуль № 2. Тестирование программ и командные методы работы. PyQT		15	34	
2.1.	Тема 1. QT 1. Что такое QT и PyQT. Знакомство	30.09 02.10	1	2	
2.2	Тема 2. QT 2. QtDesigner, ruic, два способа подключения uic файла	07.10 09.10	1	2	
2.3	Тема 3. PyQT. Установка дополнительных компонентов. PyQTgraph	14.10 16.10	1	2	
2.4	Тема 4. QT 3. Обработка исключений. Создание собственных исключений	21.10 23.10	1	2	
2.5	Тема 5. QT 4. Файлы в Python. Типы файлов и работа с ними. Внутреннее устройство Файлов	28.10 30.10	1	2	
2.6	Тема 6. QT 5. Диалоги, работа с изображениями	04.11 06.11	1	2	
2.7	Тема 7. Контрольная работа на файлы	08.11		2	
2.8	Тема 8. QT 6. Работа с простыми таблицами (csv). Работа с табличными данными вPyQT	11.11 13.11	1	2	

2.9	Тема 9. QT 7. Введение в БД, работа с SQL-таблицами и отображение данных в PyQT. Часть 1	18.11 20.11	1	2	
2.10	Тема 10. QT 8. Введение в БД, работа с SQL - таблицами и отображение данных в PyQT. Часть 2	25.11 27.11	1	2	
2.11	Тема 11. QT 9. Обработка клавиатуры и курсора. Сборка независимого приложения	02.12 04.12	1	2	
2.12	Тема 12. Защита проекта QT	06.12		2	
2.13	Тема 13. Основные команды при одиночной работе с GIT	13.01 15.01	1	2	
2.14	Тема 14. Цели и подходы к тестированию. Создание «самодельных» тестов (без библиотек)	20.01 22.01	1	2	
2.15	Тема 15. Контрольная работа на SQL-запросы	24.01		2	
2.16	Тема 16. Введение в репозитории. Подключение в PyCharm. Работа с удаленным репозиторием	27.01 29.01	1	2	
3.	Модуль № 3. Проекты Игры и API		26	53	
3.1	Тема 1. PyGame 1. Введение	03.02 05.02	1	2	
3.2	Тема 2. PyGame 2. Игровой цикл. События	10.02 12.02	1	2	
3.3	Тема 3. PyGame 3. Клетчатое поле	17.02 19.02	1	2	
3.4	Тема 4. PyGame 4. Классические игры на клетчатом поле	24.02 26.02	1	2	
3.5	Тема 5. Совместная работа над проектом, основные понятия и команды. Работа с репозиториями в среде разработки	03.03 05.03	1	2	
3.6	Тема 6. PyGame 5. Изображения. Спрайты	10.03 12.03	1	2	
3.7	Тема 7. PyGame 6. Столкновения и другие взаимодействия	17.03 19.03	1	2	
3.8	Тема 8. PyGame 7. Игра в целом	24.03 26.03	1	2	

3.9	Тема 9. PyGame 8. Украшения игры	31.03 02.04	1	2	
3.10	Тема 10. Защита проекта PyGame	04.04		2	
3.11	Тема 11. WEB. Работа с файловой системой и популярными форматами файлов: zip- архивами и json-файлами	07.04 09.04	1	2	
3.12	Тема 12. WEB. Знакомство с API	14.04 16.04	1	2	
3.13	Тема 13. WEB. Работа с командной строкой (скрипты, аргументы). Периодические задачи (модуль schedule)	21.04 23.04	1	2	
3.14	Тема 14. WEB. Библиотека argparse. Задачи на создание скриптов с ее помощью	28.04 30.04	1	2	
3.15	Тема 15. WEB. Работа с протоколом HTTP	05.05 07.05	1	2	
3.16	Тема 16. WEB. Решение задач на API Яндекс.Карт	12.05 14.05	1	2	
3.17	Тема 17. Контрольная работа на http, json, и командную строку	16.05		2	
3.18	Тема 18. WEB. Введение во flask. Обработка HTML-форм	19.05 21.05	1	2	
3.19	Тема 19. WEB. Шаблоны. flask-wtf	26.05 28.05	1	2	
3.20	Тема 20. WEB. Знакомство с flask-sqlalchemy	02.06 04.06	1	2	
3.21	Тема 21. WEB. Flask-sqlalchemy	09.06 11.06	1	2	
3.22	Тема 22. WEB. REST-API. Понятие. Делаем простое Rest-api	16.06 18.06	1	2	
3.23	Тема 23. WEB. REST-API. Flask-restful	23.06 25.06	1	2	
3.24	Тема 24. WEB. Разворачиваем проект в облаке. Дорешка	30.06. 01.07	1	2	
	Итоговая аттестация. Защита проекта WEB	03.07		2	
Итого			44	98	

Для 2 группы:

№	Наименование и содержание учебного модуля	День для 2 группы	Обязательные учебные занятия, ч		Самостоятельная работа
			лекции	практические занятия	
1.	Модуль № 1. Программирование на Python		3	9	
1.1.	Тема 1. Повторение. Решение задач на основные конструкции данных	10.09 12.09	1	3	
1.2	Тема 2. Повторение. Решение задач на классы	17.09. 19.09	1	3	
1.3	Тема 3. Повторение. Проектирование классов	24.09 26.09	1	3	
2.	Модуль № 2. Тестирование программ и командные методы работы. PyQT		15	34	
2.1.	Тема 1. QT 1. Что такое QT и PyQT. Знакомство	01.10 03.10	1	2	
2.2	Тема 2. QT 2. QtDesigner, ruic, два способа подключения uic файла	08.10 10.10	1	2	
2.3	Тема 3. PyQT. Установка дополнительных компонентов. PyQTgraph	15.10 17.10	1	2	
2.4	Тема 4. QT 3. Обработка исключений. Создание собственных исключений	22.10 24.10	1	2	
2.5	Тема 5. QT 4. Файлы в Python. Типы файлов и работа с ними. Внутреннее устройство Файлов	29.10 31.10	1	2	
2.6	Тема 6. QT 5. Диалоги, работа с изображениями	05.11 07.11	1	2	
2.7	Тема 7. Контрольная работа на файлы	08.11		2	
2.8	Тема 8. QT 6. Работа с простыми таблицами (csv). Работа с табличными данными в PyQT	12.11 14.11	1	2	

2.9	Тема 9. QT 7. Введение в БД, работа с SQL-таблицами и отображение данных в PyQT. Часть 1	19.11 21.11	1	2	
2.10	Тема 10. QT 8. Введение в БД, работа с SQL - таблицами и отображение данных в PyQT. Часть 2	26.11 28.11	1	2	
2.11	Тема 11. QT 9. Обработка клавиатуры и курсора. Сборка независимого приложения	03.12 05.12	1	2	
2.12	Тема 12. Защита проекта QT	06.12		2	
2.13	Тема 13. Основные команды при одиночной работе с GIT	10.12 12.12	1	2	
2.14	Тема 14. Цели и подходы к тестированию. Создание «самодельных» тестов (без библиотек)	17.12 19.12	1	2	
2.15	Тема 15. Контрольная работа на SQL-запросы	20.12		2	
2.16	Тема 16. Введение в репозитории. Подключение в PyCharm. Работа с удаленным репозиторием	24.12 26.12	1	2	
3.	Модуль № 3. Проекты Игры и API		26	53	
3.1	Тема 1. PyGame 1. Введение	14.01 16.01	1	2	
3.2	Тема 2. PyGame 2. Игровой цикл. События	21.01 23.01	1	2	
3.3	Тема 3. PyGame 3. Клетчатое поле	29.01 30.01	1	2	
3.4	Тема 4. PyGame 4. Классические игры на клетчатом поле	04.02 06.02	1	2	
3.5	Тема 5. Совместная работа над проектом, основные понятия и команды. Работа с репозиториями в среде разработки	11.02 13.02	1	2	
3.6	Тема 6. PyGame 5. Изображения. Спрайты	18.02 20.02	1	2	
3.7	Тема 7. PyGame 6. Столкновения и другие взаимодействия	25.02 27.02	1	2	
3.8	Тема 8. PyGame 7. Игра в целом	04.03 06.03	1	2	

3.9	Тема 9. PyGame 8. Украшения игры	11.03 13.03	1	2	
3.10	Тема 10. Защита проекта PyGame	18.03 20.03		2	
3.11	Тема 11. WEB. Работа с файловой системой и популярными форматами файлов: zip- архивами и json-файлами	25.03 27.03	1	2	
3.12	Тема 12. WEB. Знакомство с API	08.04 10.04	1	2	
3.13	Тема 13. WEB. Работа с командной строкой (скрипты, аргументы). Периодические задачи (модуль schedule)	15.04 17.04	1	2	
3.14	Тема 14. WEB. Библиотека argparse. Задачи на создание скриптов с ее помощью	22.04 24.04	1	2	
3.15	Тема 15. WEB. Работа с протоколом HTTP	29.04	1	2	
3.16	Тема 16. WEB. Решение задач на API Яндекс.Карт	05.05	1	2	
3.17	Тема 17. Контрольная работа на http, json, и командную строку	06.05		2	
3.18	Тема 18. WEB. Введение во flask. Обработка HTML-форм	07.05 09.05	1	2	
3.19	Тема 19. WEB. Шаблоны. flask-wtf	13.05 15.05	1	2	
3.20	Тема 20. WEB. Знакомство с flask-sqlalchemy	20.05 22.05	1	2	
3.21	Тема 21. WEB. Flask-sqlalchemy	27.05 29.05	1	2	
3.22	Тема 22. WEB. REST-API. Понятие. Делаем простое Rest-api	03.06 05.06	1	2	
3.23	Тема 23. WEB. REST-API. Flask-restful	10.06 12.06	1	2	
3.24	Тема 24. WEB. Разворачиваем проект в облаке. Дорешка	17.06 19.06	1	2	
3.25	Подготовка к защите проекта	24.06 26.06 03.07	2	3	
Итоговая аттестация		03.07		2	
Итого			44	98	

2.3. Содержание учебных модулей

Рабочая программа Модуля № 1. «Программирование на Python»

1. Объем учебного модуля и виды учебной работы

Виды учебной работы	Кол-во учебных часов
Лекции	3
Практические занятия, в т. ч. контрольная работа	9
Самостоятельная работа	-
Всего	12

2. Пояснительная записка

Цель изучения: повторить ранее изученный материал

После освоения учебного модуля у слушателей будут сформированы **знания:**

- что из себя представляет базовый синтаксис программ на языке Python;
- что из себя представляют классы и функции;
- как правильно проектировать классы.

Умения:

- работать в среде разработки, использовать отладку для исправления ошибок;
- создавать собственные классы и функции;
- проектировать архитектуру программы;
- использовать простые встроенные функции;
- организовывать взаимосвязь классов и функций в программе;
- гибко применять данные знания на практике для решения конкретных задач.

3. Тематический план и содержание Модуля № 1

Тема 1. Повторение. Решение задач на основные конструкции данных.

Лекция (1 час). Повторение материала прошлого года. Базовый синтаксис языка программирования Python. Простые встроенные функции. Написание собственных функций.

Практическое занятие (1 час). Решение задач разного уровня на платформе «Яндекс.Лицея» модуля «Решение задач на основные конструкции данных».

Тема 2. Повторение. Решение задач на классы.

Лекция (1 час): Повторение материала прошлого года; Роль классов в программе; Атрибуты и методы; Встроенные методы; Написание собственных классов.

Практическое занятие (1 час): Решение задач разного уровня на платформе «Яндекс.Лицея» модуля «Решение задач на классы».

Тема 3. Повторение. Проектирование классов.

Лекция (1 час). Повторение материала прошлого года; Принципы проектирования классов; Архитектура программы; Примеры решения задач с использованием классов.

Практическое занятие (1 час): Решение задач разного уровня на платформе «Яндекс.Лицея» модуля «Проектирование классов».

Рабочая программа Модуля № 2. «Тестирование программ и командные методы работы PYQT»

1. Объем учебного модуля и виды учебной работы

Виды учебной работы	Кол-во учебных часов
Лекции	15
Практические занятия, в т. ч. контрольная работа	34
Самостоятельная работа	-
Всего	49

2. Пояснительная записка

Цель изучения: сформировать представление о том, как проводить тестирование программ, какие методы командной работы используются в промышленном программировании. Также в этом модуле будет работа с использованием PyQt для создания приложения.

После освоения учебного модуля у обучающихся будут сформированы знания:

- что такое QT и PyQt;
- какие способы тестирования программ существуют;
- что такое базы данных и язык SQL, для каких задач это используется;
- какие инструменты для работы в команде используются.

Умения:

- создание компьютерного приложения с использованием PyQt;
- сборка независимого приложения;
- работа с системой контроля версий GIT;
- работа с базой данных.

3. Тематический план и содержание Модуля № 2

Тема 1. QT 1. Что такое QT и PyQt. Знакомство.

Лекция (1 час). В уроке 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 (1 час). Решение задач разного уровня на платформе «Яндекс.Лицея» модуля «QT 1. Что такое QT и PyQt. Знакомство».

Тема 2. QT 2. QtDesigner, pyuis, два способа подключения uic файла.

Лекция (1 час). В этом уроке разбирается популярный способ создания графических интерфейсов – с помощью программы QtDesigner.

Практическое занятие (1 час). Решение задач разного уровня на платформе «Яндекс.Лицея» модуля «QT 2. QtDesigner, pyuis, два способа подключения uic файла».

Тема 3. PyQt. Установка дополнительных компонентов PyQtgraph.

Лекция (1 час). Установка и технология работы с PyQtgraph.

Практическое занятие (1 час). Решение задач разного уровня на платформе «Яндекс.Лицея» модуля «PyQt. Установка дополнительных компонентов. PyQtgraph».

Тема 4. QT 3. Обработка исключений. Создание собственных исключений.

Лекция (1 час). Урок рассказывает 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 (1 час). Решение задач разного уровня на платформе «Яндекс.Лицея» модуля «QT 3. Обработка исключений. Создание собственных исключений».

Тема 5. QT 4. Файлы в Python. Типы файлов и работа с ними.

Внутреннее устройство файлов

Лекция (1 час). В уроке 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 (1 час). Решение задач разного уровня на платформе «Яндекс.Лицея» модуля «QT 4. Файлы в Python. Типы файлов и работа с ними. Внутреннее устройство файлов».

Тема 6. QT 5. Диалоги, работа с изображениями

Лекция (1 час). На этом уроке рассматриваются возможности PyQt по отображению графических данных, а также возможности рисования непосредственно на виджетах библиотеки.

Практическое занятие: (1 час). решение задач разного уровня на платформе «Яндекс.Лицея» модуля «QT 5. Диалоги, работа с изображениями».

Тема 7. QT 6. Работа с простыми таблицами (csv).

Работа с табличными данными в PyQT

Лекция (1 час). Урок посвящен 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 (1 час). решение задач разного уровня на платформе «Яндекс.Лицея» модуля «QT 6. Работа с простыми таблицами (csv). Работа с табличными данными в PyQT».

Тема 8. QT 7. Введение в БД, работа с SQL-таблицами и отображение данных в PyQT.

Лекция (1 час). На уроке начнется работа с базами данных и языком SQL. Это большая тема, которая будет сопровождать нас до конца обучения, периодически «обрастая» новыми подробностями.

Практическое занятие (1 час). Решение задач разного уровня на платформе «Яндекс.Лицея» модуля «QT 7. Введение в БД, работа с SQL-таблицами и отображение данных в PyQT. Часть 1».

Тема 9. QT 8. Введение в БД, работа с SQL-таблицами и отображение данных в PyQT.

Лекция (1 час). В уроке продолжается знакомство с базами данных и работой с ними при помощи SQL и Python. Большое внимание уделяется управлению данными: запись, изменение и удаление записей.

Практическое занятие (1 час). Решение задач разного уровня на платформе «Яндекс.Лицея» модуля «QT 8. Введение в БД, работа с SQL-таблицами и отображение данных в PyQT. Часть 2».

Тема 10. QT 9. Обработка клавиатуры и курсора. Сборка независимого приложения.

Лекция (1 час). В этом уроке показывается, как взаимодействовать с пользователем на новом уровне: добавление в программу работы с клавиатурой и мышкой. Также будет рассказано, как собрать нашу программу в exe-файл. А также изучаются основы создания SQLite-базы данных.

Практическое занятие: (1 час). Решение задач разного уровня на платформе «Яндекс.Лицея» модуля «QT 9. Обработка клавиатуры и курсора. Сборка независимого приложения».

Тема 11. Подготовка к защите проекта.

Лекция (2 часа). На уроке разбираются основные ошибки при защите и разработке проекта.

Практическое занятие (4 часа). На уроке разбираются основные ошибки при защите и разработке проекта. Работа над собственным проектом на защиту.

Тема 12. Основные команды при одиночной работе с GIT.

Лекция (1 час). Отработка основных команд GIT.

Практическое занятие (1 час). Решение задач разного уровня на платформе «Яндекс.Лицея» модуля «Основные команды при одиночной работе с GIT».

Тема 13. Цели и подходы к тестированию.

Создание «самодельных» тестов (без библиотек)

Лекция (1 час). Инструменты и технологии разработки тестов.

Практическое занятие (1 час). Решение задач разного уровня на платформе «Яндекс.Лицея» модуля «Цели и подходы к тестированию. Создание "самодельных" тестов (без библиотек)».

Тема 14. Введение в репозитории. Подключение в PyCharm.

Работа с удаленным репозиторием.

Лекция (1 час). Технология работы с репозиторием.

Практическое занятие (1 час). Решение задач разного уровня на платформе «Яндекс.Лицея» модуля «Введение в репозитории. Подключение в PyCharm. Работа с удаленным репозиторием».

Рабочая программа Модуля № 3 «Проекты Игры и API»

1. Объем учебного модуля и виды учебной работы

Виды занятий	Кол-во учебных часов
Лекции	26
Практические занятия, в т. ч. контрольная работа	53
Самостоятельная работа	-
Всего	79

2. Пояснительная записка

Цель изучения: повторить ранее изученный материал.

Умения:

- приобретение навыков поиска информации в сети Интернет, анализ выбранной информации на соответствие запросу, использование информации при решении задач;
- приобретение навыков работы с библиотеками;
- приобретение навыков решения прикладных задач;
- приобретение навыков разработки классов;
- приобретение навыков организации времени и планирования.

3. Тематический план и содержание учебного Модуля № 3 «Проекты Игры и API»

Тема 1. PyGame 1. Введение.

Лекция (2 часа). В этом уроке мы поговорим о том, как создаются игры на языке Python. Рассмотрим возможности библиотеки PyGame, научимся подключать ее и запускать приложение.

Практическое занятие (2 часа). Решение задач разного уровня на платформе «Яндекс.Лицея» модуля «PyGame 1. Введение».

Тема 2. PyGame 2. Игровой цикл. События.

Лекция (1 час). На уроке рассматривается взаимодействие программы и пользователя. Рассказывается о том, как работает игровой цикл, что из себя представляют события.

Практическое занятие (1 час). Решение задач разного уровня на платформе «Яндекс.Лицея» модуля «Игровой цикл. События».

Тема 3. PyGame 3. Клетчатое поле.

Лекция (2 часа). В данном уроке мы поговорим о том, как реализуется клетчатое поле – основа для будущей игры.

Практическое занятие (2 часа). Решение задач разного уровня на платформе «Яндекс.Лицея» модуля «Клетчатое поле».

Тема 4. PyGame 4. Классические игры на клетчатом поле.

Лекция (2 часа). В данном уроке рассматриваются простейшие игры на клетчатом поле. Приводится пример создания игры на языке Python.

Практическое занятие (2 часа). Решение задач разного уровня на платформе «Яндекс.Лицея» модуля «Функции с переменным числом аргументов» и модуля «Классические игры на клетчатом поле».

Тема 5. PyGame 5. Изображения. Спрайты.

Лекция (1 час). На уроке мы поговорим о том, как добавить графику в игру, а также о том, что такое спрайты, научимся создавать их на основе собственных изображений.

Практическое занятие: (1 час). Решение задач разного уровня на платформе «Яндекс.Лицея» модуля «Изображения. Спрайты».

Тема 6. PyGame 6. Столкновения и другие взаимодействия.

Лекция (2 часа). На занятии мы научимся управлять взаимодействиями между игровыми объектами. Рассмотрим то, как реализуется столкновение между объектами, и т.д.

Практическое занятие (2 часа). Решение задач разного уровня на

платформе «Яндекс.Лицея» модуля «Столкновения и другие взаимодействия».

Тема 7. PyGame 7. Игра в целом.

Лекция (2 часа). Мы применим полученные на прошлых уроках знания, чтобы сделать работоспособную игру. Рассмотрим составляющие игры и этапы, необходимые для ее создания.

Практическое занятие (2 часа). Решение задач разного уровня на платформе «Яндекс.Лицея» модуля «Игра в целом».

Тема 8. PyGame 8. Украшения игры.

Лекция (2 часа). В этом уроке мы завершим работу над игрой. К основному функционалу добавим детали, позволяющие украсить нашу игру.

Практическое занятие (2 часа). Решение задач разного уровня на платформе «Яндекс.Лицея» модуля «Украшения игры».

Тема 9. WEB. Работа с файловой системой и популярными форматами файлов: zip-архивами и json-файлами.

Лекция (2 часа). В этом уроке даются знания о работе с файловой системой и популярными форматами файлов, такими как zip-архивы и json-файлы.

Практическое занятие (2 часа). Решение задач разного уровня на платформе «Яндекс.Лицея» модуля «Работа с файловой системой и популярными форматами файлов».

Тема 10. WEB. Знакомство с API.

Лекция (2 часа). В этом уроке мы переходим к изучению WEB-разработки. Мы узнаем, что такое API, где и зачем применяется эта технология, рассмотрим несколько примеров использования.

Практическое занятие (2 часа). Решение задач разного уровня на платформе «Яндекс.Лицея» модуля «Знакомство с API».

Тема 11. WEB. Работа с командной строкой (скрипты, аргументы). Периодические задачи (модуль schedule).

Лекция (2 часа): В этом уроке мы рассмотрим работу с командной строкой. Узнаем, что такое скрипты и аргументы. Также мы научимся запускать периодические задачи, которые должны выполняться по расписанию. Познакомимся с модулем schedule.

Практическое занятие (2 часа). Решение задач разного уровня на платформе «Яндекс.Лицея» модуля «Работа с командной строкой».

Тема 12. WEB. Библиотека argparse. Задачи на создание скриптов с ее помощью.

Лекция (2 часа). На этом занятии мы продолжим работать с командной строкой. Рассмотрим способы упрощения этого процесса, познакомимся с библиотекой argparse.

Практическое занятие (2 часа). Решение задач разного уровня на платформе «Яндекс.Лицея» модуля «Библиотека argparse».

Тема 13. WEB. Работа с протоколом HTTP.

Лекция (2 часа). На этом занятии мы познакомимся с протоколом HTTP. Эти знания являются фундаментальными при построении WEB-приложений. Также рассмотрим модель OSI и другие протоколы, входящие в нее.

Практическое занятие (2 часа). Решение задач разного уровня на платформе «Яндекс.Лицея» модуля «Проектирование и разработка классов. Часть 1» и модуля «Проектирование и разработка классов. Часть 1».

Тема 14. WEB. Решение задач на API Яндекс.Карт.

Лекция (2 часа). На этом занятии мы узнаем, как использовать стороннее API на примере сервиса «Яндекс.Карты». Мы научимся создавать программы, которые позволят получать информацию об объекте по его координатам, а также получать изображение карты по описанию объекта.

Практическое занятие (2 часа). Решение задач разного уровня на платформе «Яндекс.Лицея» модуля «Решение задач на API Яндекс.Карт».

Тема 15. WEB. Введение во flask. Обработка HTML-форм.

Лекция (2 часа). На этом занятии мы познакомимся с библиотекой flask,

позволяющей создавать полноценные WEB-приложения. Также рассмотрим обработку HTML-форм с помощью flask.

Практическое занятие (2 часа). Решение задач разного уровня на платформе «Яндекс.Лицея» модуля «Введение во flask. Обработка HTML-форм».

Тема 16. WEB. Шаблоны. flask-wtf.

Лекция (2 часа). На этом занятии мы продолжим работать с flask. Рассмотрим мощный инструмент шаблонов, позволяющий динамически менять содержимое веб-страницы. Познакомимся с модулем flask-WTF для удобной работы с формами.

Практическое занятие (2 часа). Решение задач разного уровня на платформе «Яндекс.Лицея» модуля «Шаблоны. flask-wtf».

Тема 3.17. WEB. Знакомство с flask-sqlalchemy.

Лекция (2 часа). В этом уроке мы начнем работать с базами данных и языком SQL, а также познакомимся с модулем flask-sqlAlchemy – набором SQL-инструментов для Python, научимся его устанавливать и подключать к проекту.

Практическое занятие (2 часа). Решение задач разного уровня на платформе «Яндекс.Лицея» модуля «Знакомство с flask-sqlalchemy».

Тема 18. WEB. Flask-sqlalchemy.

Лекция (2 часа). На этом занятии мы продолжим работу с flask-sqlalchemy. Познакомимся с инструментом объектно-реляционного отображения (ORM), который предоставляет разработчикам всю мощь и гибкость SQL.

Практическое занятие (2 часа). Решение задач разного уровня на платформе «Яндекс.Лицея» модуля «Flask-sqlalchemy».

Тема 19. WEB. REST-API. Понятие. Делаем простое Rest-api.

Лекция (2 часа). На этом занятии мы рассмотрим наиболее популярный способ взаимодействия компонентов веб-приложения – REST-API. Познакомимся с его основными принципами и научимся добавлять REST-API в существующее приложение.

Практическое занятие (2 часа). Решение задач разного уровня на платформе «Яндекс.Лицея» модуля «REST-API. Понятие. Делаем простое Rest-api».

Тема 20. WEB. REST-API. Flask-restful.

Лекция (2 часа). В этом уроке мы продолжим работать с REST-API. Мы познакомимся с модулем Flask-restful, позволяющим серьезно упростить написание веб-приложений.

Практическое занятие (2 часа). Решение задач разного уровня на платформе «Яндекс.Лицея» модуля «REST-API. Flask-restful».

Тема 21. WEB. Разворачиваем проект в облаке. Дорешка.

Лекция (2 часа). На этом занятии мы завершаем создание WEB-приложения. Мы научимся разворачивать веб-приложение в облаке для того, чтобы сделать его общедоступным.

Практическое занятие (2 часа). Решение задач разного уровня на платформе «Яндекс.Лицея» модуля «Разворачиваем проект в облаке. Дорешка».

3. МЕТОДИЧЕСКОЕ ОБЕСПЕЧЕНИЕ ПРОГРАММЫ

3.1. Формы занятий, планируемых по каждому разделу

Основной тип занятий — комбинированный, сочетающий в себе элементы теории и практики. Большинство заданий ДОП 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 ДОП 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.

3.2. Приемы и методы организации учебно-воспитательного процесса

При организации занятий по ДОП 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.

Общая характеристика учебного процесса:

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

4. ОРГАНИЗАЦИОННО-ПЕДАГОГИЧЕСКИЕ УСЛОВИЯ РЕАЛИЗАЦИИ ПРОГРАММЫ

4.1. Материально-техническое обеспечение

№	Наименование учебного модуля в соответствии с учебным планом	Наименование специализированных аудиторий, кабинетов, лабораторий и пр. с перечнем основного оборудования	Форма владения, пользования (собственность, оперативное управление, аренда и т. п.)
1	Программирование Python	Кабинет № 617 персональный компьютер – 26 шт., принтер – 1 шт., мультимедийный проектор – 1 шт., доска-флипчарт – 1 шт., модуль беспроводного доступа – 1 шт., стенд-выставка на колесах – 2 шт., телевизор MISTERY – 1 шт., читальный аппарат INDUS – 2 шт., стол компьютерный – 4 шт., стол письменный однотумбовый (преподавательский) – 1 шт., стул подъемно-поворотный – 1 шт., стол читательский одноместный – 27 шт., стул рабочий – 30 шт., кафедра – 1 шт.	Оперативное управление по договору Собственность
2	Тестирование программы командные методы работы	Кабинет № 617 персональный компьютер – 26 шт., принтер – 1 шт., мультимедийный проектор – 1 шт., доска-флипчарт – 1 шт., модуль беспроводного доступа - 1 шт., стенд-выставка на колесах – 2 шт., телевизор MISTERY – 1 шт., читальный аппарат INDUS – 2 шт., стол компьютерный – 4 шт., стол письменный однотумбовый (преподавательский) – 1 шт., стул подъемно-поворотный – 1 шт., стол читательский одноместный – 27 шт., стул рабочий – 30 шт., кафедра – 1 шт.	Оперативное управление по договору Собственность

3	Проекты Игры и API	Кабинет № 617 персональный компьютер – 26 шт., принтер – 1 шт., мультимедийный проектор – 1 шт., доска-флипчарт – 1 шт., модуль беспроводного доступа – 1 шт., стенд-выставка на колесах – 2 шт., телевизор MISTERY – 1 шт., читальный аппарат INDUS – 2 шт., стол компьютерный – 4 шт., стол письменный одностумбовый (преподавательский) – 1 шт., стул	Оперативное управление по договору Собственность
---	--------------------	--	---

4.2. Обеспеченность учебными материалами, техническими и электронными средствами обучения и контроля знаний

№ п/п	Наименование	Вид	Форма доступа	Количество
1.	Интернет-ресурсы. Статьи	Электронный	Кабинеты	617
2.	Мультимедийные презентации лекций	Электронный	Кабинеты	617
3.	Методические материалы	Электронный	Кабинеты	617

4.3. Кадровое обеспечение

Преподавательский состав

Ведущими преподавателями программы являются специалисты с высшим техническим образованием, имеющие опыт работы в преподавании и IT-индустрии, ведущие методисты сектора Учебный центр научно-методического отдела ГБУК ИОГУНБ

5. СПИСОК ЛИТЕРАТУРЫ

1. Информатика : учебное пособие / под ред. Б. Е. Одинцова, А. Н. Романова ; Финансовый университет при Правительстве Российской Федерации. – 2-е издание, переработанное и дополненное. – Москва : Вузовский учебник : Инфра-М, 2016. – 409 с.
2. Кольцов, Д. В. Python: создаем программы и игры / Д. Кольцов. – Санкт-Петербург : Наука и техника, 2017. – 395 с. : ил. – (Серия «Просто о сложном»).
3. Мэтиз, Э. Изучаем Python. Программирование игр, визуализация данных, веб- приложения / Эрик Мэтиз ; [пер. с англ. Е. Матвеева]. – Санкт-Петербург [и др.] : Питер, 2017. – 492 с. : ил. – (Библиотека программиста).
4. Любанович, Б. Простой Python. Современный стиль программирования / Билл Любанович ; [пер. с англ. Е. Зазноба]. – Санкт-Петербург [и др.] : Питер, 2019. – 476 с. : ил.,табл. – (Бестселлеры O'Reilly).
5. Ночка, Е. И. Основы алгоритмизации и программирования (на языке Питон) : учебник для студентов среднего профессионального образования по направлениям подготовки 09.02.01 «Компьютерные системы и комплексы», 09.02.04 «Информационные системы (по отраслям)» / Е. И. Ночка. – Москва : КУРС, 2019. – 203 с. : ил.
6. Седер, Н. Python : экспресс-курс / Наоми Седер ; перевел с английского Е. Матвеев. –3-е издание. – Санкт-Петербург [и др.] : Питер, 2019. – 480 с.
7. Сэнд, Уоррен. Hello World! Занимательное программирование / Уоррен Сэнд, КартерСэнд ; [пер. И. Рузмайкина]. – Санкт-Петербург [и др.] : Питер, 2017. – 400 с.