

Муниципальное автономное общеобразовательное учреждение города
Новосибирска
«Инженерный лицей Новосибирского государственного технического
университета»

Рекомендовано решением
педагогического совета МАОУ
«Инженерный лицей НГТУ»
Протокол № 1 от 26.08. 2025

Утверждаю
Директор МАОУ
«Инженерный лицей НГТУ»
Безлекина М.А.
Приказ №228 от 26.08.2025

**Рабочая программа по курсу внеурочной
деятельности «Программирование Python»**

название учебного предмета, курса, дисциплины (модуля)

для класса(ов) 7-9

Количество часов: 2 часа в неделю

Всего: 204 часа

Из них: 7 класс 68 часа

8 класс 68 часа

9 класс 68 часа

Разработчики программы:

Муль Павел Фридрихович учитель информатики высшей квалификационной категории,
Гурина Анна Анатольевна учитель информатики высшей квалификационной категории,
Цысс Евгения Сергеевна учитель информатики высшей квалификационной категории,
Орлова Анна Алексеевна учитель информатики первой квалификационной категории.

г. Новосибирск
2025

Программа обсуждалась на заседании кафедры / методического объединения
учителей математики и информатики МАОУ «Инженерный лицей НГТУ»
Протокол заседания № 1 от 26.08.2025 г.

Подолян Е.В / Подолян Е.В
Ф.И.О. руководителя кафедры/МО

Пояснительная записка

Одновременно с революционным развитием аппаратного и программного обеспечения и оснащением современной компьютерной техникой учебных заведений курс информатики претерпел существенные изменения. Наиболее яркая характеристика такого изменения - вымывание программирования из школьного курса информатики. Основное внимание стало уделяться освоению современных информационных технологий. Но, как заметил А.Г. Гейн, «...очевидно, что именно алгоритмизация с самого начала вытянула на школьную арену курс информатики и ныне во многих реально существующих курсах информатики позволяет уйти от умных, но пустоватых разговоров к конкретному делу».

Данный курс является вводным курсом по программированию, дающий представление о базовых понятиях структурного программирования (данных, операциях, переменных, ветвлениях в программе, циклах и функциях). Выбор Python обусловлен тем, что это язык, обладающий рядом преимуществ перед другими языками для начинающих изучать программирование - ясность кода, быстрота реализации. По результатам исследования компании JetBrains (<https://www.jetbrains.com/ru-ru/lp/devecosystem-2019/>) язык Python – самый изучаемый и один из самых востребованных на рынке труда.

Изучение Python в школе откроет ученикам возможности дальнейшего развития в области ИТ и поможет профориентации в старших классах, пригодится в олимпиадах по программированию и решении заданий ЕГЭ.

Курс предполагает смешанный формат обучения: очное и онлайн-обучение. На очных занятиях в группах по 10-15 человек используются задания на онлайн-платформе с автоматизированной проверкой. Теоретические блоки и задания на платформе доступны ученикам для самостоятельного изучения в любое время. В ходе курса запланирован плавный переход к формату “перевернутого класса”: на первых пяти занятиях ученики знакомятся с онлайн-платформой и основными инструментами программирования на Python. Начиная с шестого урока организуются различные формы самостоятельной деятельности на уроке, с групповой взаимопроверкой. После десятого урока иногда ученикам предлагается самостоятельное изучение новой темы, с опорой на конспект онлайн-курса, а учителям является консультантом и модератором. Последние пять уроков отведены самостоятельную работу над проектами. Сочетание групповой работы с учителем в классе и индивидуальной работы в личном кабинете на онлайн-платформе Stepik позволяет ученикам выработать не только технические навыки программирования, но и навыки социального взаимодействия при работе над финальным проектом курса, а главное – научиться самостоятельно выстраивать свое профессиональное развитие.

Отличительная особенность данной программы от существующих образовательных программ заключается в том, что изучается материал, слабо представленный и не представленный в программе основного курса информатики и ИКТ. Он систематизирован, доступно и логично излагается, подкреплен мощным дидактическим материалом, направлен на практику программирования и подготовку к олимпиадам, на развитие творчества и самостоятельности обучающихся. На занятиях организуется деятельность, создающая условия для творческого развития воспитанников на различных возрастных этапах и предусматривающая их дифференциацию по степени одаренности. Данный курс позволяет повысить результативность обучения информатики и ИКТ при параллельном преподавании школьного основного курса и данного дополнительного, благодаря авторскому подходу к организации образовательного процесса типовой программы Благотворительного фонда развития образования “АйкьюОпшн” (“Возможность Интеллекта”).

Основной целью курса является – формирование у обучающихся навыков программирования на языке Python и самонаправленного обучения.

Достижение цели предполагает решение следующих задач:

1. Сформировать основные навыки программирования на языке Python.
2. Научить применять полученные знания для решения практических задач.
3. Повысить уровень самостоятельности в обучении (по четырехступенчатой шкале Г. Гроу)

4. Научить декомпозировать задачи и работать над ними в небольших командах, с разделением ответственности и ролей.

Рабочая программа разработана на основе следующих нормативно-правовых документов:

- Федеральный закон от 29.12.2012 года № 273-ФЗ «Об образовании в Российской Федерации» с изм. от 27.12.2019 г.
- Федеральный государственный образовательный стандарт среднего общего образования (с изменениями и дополнениями), утвержденный приказом Министерства образования и науки РФ от 17 мая 2012 г. N 413.
- Примерная рабочая программа курса внеурочной деятельности «Основы программирования на Python» для 7-9 классов образовательных организаций. ФГБНУ Институт стратегии развития образования, Москва 2022г.
- Основная образовательная программа основного общего образования МАОУ «Инженерный лицей НГТУ».
- Программа воспитания МАОУ «Инженерного лицея НГТУ».
- Учебный план МАОУ «Инженерный лицей НГТУ».

Учебно-методическое и материально-техническое обеспечение образовательного процесса

Литература

1. Васильев А. Н. Python на примерах: практикум /А. Н. Васильев - Наука и Техника, 2019 - 432 с.
2. Прохоренок Н. А. Python 3: самое необходимое: практикум / Н. А. Прохоренок, В. А. Дронов - БХВ-Петербург, 2019 - 608 с.
3. Гэддис Т. Начинаем программировать на Python: учебник/Т. Гэддис - БХВ-Петербург, 2019 - 768 с.
4. Седжвик Р. Программирование на языке Python: /Р. Седжвик, К., Уэйн Р., Дондеро - Вильямс, 2017 - 736 с.
5. Харрисон М. Как устроен Python.: практикум / М. Харрисон - Питер, 2002 - 272 с.

Технические средства

Требования к комплектации компьютерного класса

Для реализации учебного курса необходимо наличие компьютерного класса в соответствующей комплектации. Наиболее рациональным является установка в компьютерном классе 13–15 компьютеров (рабочих мест) для школьников и одного компьютера (рабочего места) для педагога. Предполагается объединение компьютеров в локальную сеть с возможностью выхода в Интернет, что позволяет использовать сетевые цифровые образовательные ресурсы.

Минимальные требования к техническим характеристикам каждого компьютера следующие:

- процессор – не ниже Celeron с тактовой частотой 2,8 ГГц;
- оперативная память – не менее 2Гб;
- жидкокристаллический монитор с диагональю не менее 15 дюймов;
- жёсткий диск – не менее 250 Гб;
- клавиатура;
- мышь;
- устройство для чтения компакт-дисков (желательно).

Кроме того, в кабинете информатики должны быть:

- принтер на рабочем месте учителя;
- проектор и аудиоколонки на рабочем месте учителя;
- устройства, обеспечивающие подключение к сети;

- интернет-соединение, скорость загрузки не менее 2 Мбит/сек;
- устройства вывода звуковой информации;
- устройства для ручного ввода текстовой информации и манипулирования экранными элементами.

Требования к программному обеспечению компьютеров:

На компьютерах должна быть установлена операционная система Windows7 или моложе /MacOS/Unix-based системы с поддержкой протокола HTML5, а также необходимое программное обеспечение:

- файловый менеджер (в составе операционной системы или др.);
- антивирусная программа;
- программа-архиватор;
- мультимедиа-проигрыватель (входит в состав операционных систем или др.);
- браузер Google Chrome;
- интерактивная оболочка (бесплатная IDEWing101 или аналог)
- текстовый редактор Brackets, или аналог;
- графический редактор Gimp, или аналог;
- текстовый процессор (MS Word или OpenOffice.org Writer).

Электронные образовательные ресурсы

1. Курс Поколение Python: курс для начинающих на платформе Stepik - <https://stepik.org/course/58852/syllabus>. (для 7 класса)
2. Курс Поколение Python: курс для продвинутых на платформе Stepik - <https://stepik.org/course/68343/syllabus>.(для 8 класса)
3. Курс Поколение Python: курс для продвинутых на СДО лицея НГТУ (Moodle). (для 9 класса)
4. Интерактивные проверочные задания в приложениях Kahoot и LearningApps.

Используемые методические материалы

1. Типовая рабочая программа курса «Python для начинающих» - Белых Р., Гуев Т., Скобина О. (авторский коллектив благотворительного фонда развития образования “АйкьюОпшн”, г.Санкт-Петербург 2022;
2. Типовая рабочая программа курса «Поколение Python: курс для продвинутых» - Белых Р., Гуев Т., Скобина О. (авторский коллектив благотворительного фонда развития образования “АйкьюОпшн” г.Санкт-Петербург 2022;
3. Методические указания для учителя - Белых Р., Гуев Т., Скобина О. (авторский коллектив благотворительного фонда развития образования “АйкьюОпшн” г.Санкт-Петербург 2023;
4. Презентации с иллюстративным изложением теоретического материала- Белых Р., Гуев Т., Скобина О. (авторский коллектив благотворительного фонда развития образования “АйкьюОпшн”, г.Санкт-Петербург 2023;
5. Инструкции подвижных игр, направленных на закрепление знаний, полученных на занятиях- Белых Р., Гуев Т., Скобина О. (авторский коллектив благотворительного фонда развития образования “АйкьюОпшн”, г.Санкт-Петербург 2022;
6. Инструкции для проведения рефлексии процесса обучения с учениками- Белых Р., Гуев Т., Скобина О. (авторский коллектив благотворительного фонда развития образования “АйкьюОпшн”, г.Санкт-Петербург 2022;
7. Авторские проектные задания для создания учащимися игр на языке Python - Белых Р., Гуев Т., Скобина О. (авторский коллектив благотворительного фонда развития образования “АйкьюОпшн”, г.Санкт-Петербург 2022.
8. Клейнберг Дж. Алгоритмы: разработка и применение. СПб: Питер, 2016. - 800 с.

9. Златопольский Д. М. Основы программирования на языке Python. – М.: ДМК Пресс, 2017. – 284 с.: ил.
10. Бхаргава А. Грокаем алгоритмы. Иллюстрированное пособие для программистов и любопытствующих. - СПб.: Питер, 2017. - 288 с.
11. Златопольский Д.М. Сборник задач по программированию. СПб: БХВ-Петербург, 2011. - 295 с.

**Планируемые результаты освоения курса внеурочной деятельности
«Программирование Python»**

ЛИЧНОСТНЫЕ РЕЗУЛЬТАТЫ

Патриотическое воспитание:

- ценностное отношение к отечественному культурному, историческому и научному наследию;
- понимание значения информатики как науки в жизни современного общества.

Духовно-нравственное воспитание:

- ориентация на моральные ценности и нормы в ситуациях нравственного выбора;
- готовность оценивать своё поведение и поступки, а также поведение и поступки других людей с позиции нравственных и правовых норм с учётом осознания последствий поступков;
- активное неприятие асоциальных поступков, в том числе в Интернете.

Гражданское воспитание:

- представление о социальных нормах и правилах межличностных отношений в коллективе, в том числе в социальных сообществах;
- соблюдение правил безопасности, в том числе навыков безопасного поведения в интернет-среде;
- ориентация на совместную деятельность при выполнении учебных и познавательных задач, создания учебных проектов;
- стремление оценивать своё поведение и поступки своих товарищей с позиции нравственных и правовых норм с учётом осознания последствий поступков.

Ценность научного познания:

- наличие представлений об информации, информационных процессах и информационных технологиях, соответствующих современному уровню развития науки и общественной практики;
- интерес к обучению и познанию;
- любознательность;
- стремление к самообразованию;
- овладение начальными навыками исследовательской деятельности, установка на осмысление опыта, наблюдений, поступков и стремление совершенствовать пути достижения индивидуального и коллективного благополучия;
- наличие базовых навыков самостоятельной работы с учебными текстами, справочной литературой, разнообразными средствами информационных технологий, а также умения самостоятельно определять цели своего обучения, ставить и формулировать для себя новые задачи в учёбе и познавательной деятельности, развивать мотивы и интересы своей познавательной деятельности.

Формирование культуры здоровья:

- установка на здоровый образ жизни, в том числе и за счёт освоения и соблюдения требований безопасной эксплуатации средств ИКТ.

Трудовое воспитание:

- интерес к практическому изучению профессий и труда в сферах деятельности, связанных с информатикой, программированием и информационными технологиями, основанными на достижениях науки информатики и научно-технического прогресса.

Экологическое воспитание:

- наличие представлений о глобальном характере экологических проблем и путей их решения, в том числе с учётом возможностей ИКТ.

Адаптация обучающегося к изменяющимся условиям социальной среды:

- освоение обучающимися социального опыта, основных социальных ролей, соответствующих ведущей деятельности возраста, норм и правил общественного поведения, форм социальной жизни в группах и сообществах, в том числе в виртуальном пространстве.

МЕТАПРЕДМЕТНЫЕ РЕЗУЛЬТАТЫ

Универсальные познавательные действия

Базовые логические действия:

- умение определять понятия, создавать обобщения, устанавливать аналогии, классифицировать, самостоятельно выбирать основания и критерии для классификации, устанавливать причинно-следственные связи, строить логические рассуждения, делать умозаключения (индуктивные, дедуктивные и по аналогии) и выводы;
- умение создавать, применять и преобразовывать знаки и символы, модели и схемы для решения учебных и познавательных задач;
- самостоятельно выбирать способ решения учебной задачи (сравнивать несколько вариантов решения, выбирать наиболее подходящий с учётом самостоятельно выделенных критериев).

Базовые исследовательские действия:

- формулировать вопросы, фиксирующие разрыв между реальным и желательным состоянием ситуации, объекта, и самостоятельно устанавливать искомое и данное;
- оценивать применимость и достоверность информации, полученной в ходе исследования;
- прогнозировать возможное дальнейшее развитие процессов, событий и их последствия в аналогичных или сходных ситуациях, а также выдвигать предположения об их развитии в новых условиях и контекстах.

Работа с информацией:

- выявлять дефицит информации, данных, необходимых для решения поставленной задачи;
- применять основные методы и инструменты при поиске и отборе информации из источников с учётом предложенной учебной задачи и заданных критериев;
- выбирать, анализировать, систематизировать и интерпретировать информацию различных видов и форм представления;
- выбирать оптимальную форму представления информации и иллюстрировать решаемые задачи несложными схемами, диаграммами, иными графическими объектами и их комбинациями;
- оценивать достоверность информации по критериям, предложенным учителем или сформулированным самостоятельно;
- запоминать и систематизировать информацию.

Универсальные коммуникативные действия

Общение:

- сопоставлять свои суждения с суждениями других участников диалога, обнаруживать различие и сходство позиций;
- публично представлять результаты выполненного опыта (исследования, проекта);
- выбирать формат выступления с учётом задач презентации и особенностей аудитории и в соответствии с ним составлять устные и письменные тексты с использованием иллюстративных материалов.

Совместная деятельность (сотрудничество):

- понимать и использовать преимущества командной и индивидуальной работы при решении конкретной проблемы, в том числе при создании информационного продукта;
- принимать цель совместной информационной деятельности по сбору, обработке, передаче, формализации информации; коллективно строить действия по её достижению: распределять роли, договариваться, обсуждать процесс и результат совместной работы;

- выполнять свою часть работы с информацией или информационным продуктом, достигая качественного результата по своему направлению и координируя свои действия с другими членами команды;
- оценивать качество своего вклада в общий информационный продукт по критериям, самостоятельно сформулированным участниками взаимодействия;
- сравнивать результаты с исходной задачей и вклад каждого члена команды в достижение результатов, разделять сферу ответственности и проявлять готовность к предоставлению отчёта перед группой.

Универсальные регулятивные действия

Самоорганизация:

- выявлять в жизненных и учебных ситуациях проблемы, требующие решения;
- составлять алгоритм решения задачи (или его часть), выбирать способ решения учебной задачи с учётом имеющихся ресурсов и собственных возможностей, аргументировать выбор варианта решения задачи;
- составлять план действий (план реализации намеченного алгоритма решения), корректировать предложенный алгоритм с учётом получения новых знаний об изучаемом объекте.

Самоконтроль (рефлексия):

- владеть способами самоконтроля, самомотивации и рефлексии;
- учитывать контекст и предвидеть трудности, которые могут возникнуть при решении учебной задачи, адаптировать решение к меняющимся обстоятельствам;
- вносить корректизы в деятельность на основе новых обстоятельств, изменившихся ситуаций, установленных ошибок, возникших трудностей;
- оценивать соответствие результата цели и условиям.

Эмоциональный интеллект:

- ставить себя на место другого человека, понимать мотивы и намерения другого.

Принятие себя и других:

- осознавать невозможность контролировать всё вокруг даже в условиях открытого доступа к любым объёмам информации;
- осознанно относиться к другому человеку, его мнению.

ПРЕДМЕТНЫЕ РЕЗУЛЬТАТЫ

7 класс

К концу обучения в 7 классе обучающийся научится:

- соблюдать требования безопасности при работе за компьютером;
- объяснять, что такое алгоритм, язык программирования, программа;
- навыкам алгоритмического мышления и пониманием необходимости формального описания алгоритмов.
- использовать переменные различных типов при написании программ на Python;
- использовать оператор присваивания при написании программы на Python;
- объяснять, что такое логическое выражение, вычислять значение логического выражения, записывать логическое выражение на Python;
- реализовывать алгоритмы в виде программ и программных систем.
- владеть элементарными навыками документирования программ.
- сопоставлять математические модели задач и их компьютерные аналоги, анализировать полученные результаты с точки зрения соответствия объекту и целям моделирования.
- анализировать разные способы записи алгоритмов с позиции того, что они являются информационными моделями.
- владеть универсальным языком программирования высокого уровня, представлениями о базовых типах данных и структурах данных.
- использовать основные управляющие конструкции языка.
- производить отладку программ с помощью встроенного отладчика.

8 класс

К концу обучения в 8 классе обучающийся научится:

- соблюдать требования безопасности при работе на компьютере;
- основам практических базисных знаний программирования в Python, и их применению;
- писать программы на Python для рисования геометрических фигур, используя модуль Turtle;
- основам практических базисных навыков разработки игр с помощью черепашьей графики (библиотека turtle);
- понимать различия локальных и глобальных переменных;
- решать задачи с использованием глобальных переменных на Python;
- основам практических базисных навыков по созданию мемов (библиотека pillow);
- основам практических базисных навыков по созданию телеграмм-бота (библиотека telebot);
- сопоставлять математические модели задач и их компьютерные аналоги, анализировать полученные результаты с точки зрения соответствия объекту и целям моделирования;
- анализировать разные способы записи алгоритмов с позиции того, что они являются информационными моделями;
- писать свои функции на Python;
- разбивать задачи на подзадачи;

9 класс

К концу обучения в 9 классе обучающийся научится:

- соблюдать требования безопасности при работе на компьютере;
- планировать действия, необходимые для решения задач высокого и повышенного уровня сложности с использованием языка программирования, оценивать результаты работы;
- реализовывать сложные алгоритмы по обработке файлов в виде программ и программных систем;
- определять необходимый тип данных, с учетом особенностей устройства памяти.
- организовывать рекурсивную обработку данных;
- сопоставлять математические модели задач и их компьютерные аналоги, анализировать полученные результаты с точки зрения соответствия объекту и целям моделирования;
- анализировать разные способы записи алгоритмов с позиции того, что они являются информационными моделями;
- владеть универсальным языком программирования высокого уровня на базисном уровне.

Содержание курса

7 класс:

Знакомство с Python.

Переменные и выражения.

Условные предложения.

Циклы.

Работа со строками.

Работа со списками.

Функции.

Локальные и глобальные переменные.

Работа над итоговым проектом.

8 класс:

Повторение основных конструкций Python .

Тип данных bool и NoneType .

Вложенные списки .

Кортежи.

Множества.

Словари.

Модули decimal, fraction и другие.

Функции.

Работа с файлами.

Работа над итоговым проектом.

9 класс

Повторение основных конструкций Python .

Работа с датой и временем.

Работа с файлами.

Устройство памяти и особенности типов данных.

Дополнительные типы коллекций.

Обработка исключений.

Рекурсия.

Функции.

Итераторы и генераторы.

Регулярные выражения.

Тематическое планирование 7 класс 68 часов

№ пп	Название раздела	Количе ство часов	Основные виды деятельности обучающихся
1.	Знакомство с Python	2	<p>Знать структуру программы на Python, режимы работы с Python.</p> <p>Выполнять установку среды программирования.</p> <p>Уметь составлять простейшую программу, запускать ее в интерактивной среде.</p> <p>Писать комментарии в программе.</p>
2.	Переменные и выражения	6	<p>Знать операторы ввода-вывода.</p> <p>Уметь работать со справочной системой среды программирования.</p> <p>Решать задачи на элементарные действия с числами.</p> <p>Уметь пользоваться интерфейсом среды программирования Python.</p> <p>Использовать команды редактора программы.</p> <p>Организовывать ввод и вывод данных.</p> <p>Записывать арифметические выражения, учитывая приоритет действий.</p>
3.	Условные предложения	6	<p>Знать назначение условного оператора, способы записи условного оператора.</p> <p>Знать и использовать логические операторы or, and, not.</p> <p>Создавать сложные условия с помощью логических операторов.</p> <p>Уметь решать задачи по теме "Условные операторы".</p> <p>Составлять программы с конструкциями ветвления, используя простые и сложные условия.</p>
4.	Циклы	10	<p>Знать правила записи циклов условием, уметь использовать данные правила при написании программ.</p> <p>Знать и использовать назначение и особенности использования цикла с параметром, его форматы записи.</p> <p>Использовать при решении задач операторы for и while.</p> <p>Уметь реализовывать циклические алгоритмы, составлять программы с циклом.</p> <p>Уметь определять вид цикла, наиболее удобный для решения поставленной задачи.</p>
5.	Работа со строками	10	Знать назначение строкового типа данных.

			<p>Использовать операторы для работы со строками.</p> <p>Знать процедуры и функции для работы со строками.</p> <p>Уметь использовать операции со строками при решении задач:</p> <ul style="list-style-type: none"> – описывать строки; – соединять строки; – находить длину строки; – вырезать часть строки; – находить подстроку в строке; – находить количество слов в строке; – находить количество символов в строке; – находить заданный символ в строке.
6.	Работа со списками	10	<p>Знать основы работы со списками.</p> <p>Использовать методы списков при решении задач.</p> <p>Уметь выполнять вывод элементов списка.</p> <p>Уметь использовать и различать строковые методы <code>split()</code> и <code>join()</code></p> <p>Уметь решать задачи со списочными выражениями.</p>
7.	Функции	10	<p>Знать способы описания функции.</p> <p>Уметь использовать при решении задач принципы структурного программирования.</p> <p>Знать понятие формальных и фактических параметров подпрограмм, уметь использовать различные способы передачи параметров подпрограмм.</p> <p>Уметь создавать и использовать функции простые и рекурсивные и использовать их при решении задач.</p> <p>Уметь использовать механизм параметров для передачи значений в программах.</p>
8.	Локальные и глобальные переменные	6	<p>Различать понятия локальных и глобальных переменных.</p> <p>Уметь писать функции возвращающие значения. Уметь использовать локальные и глобальные переменные в подпрограммах.</p> <p>Уметь использовать при решении задач принципы структурного программирования.</p> <p>Уметь использовать механизм параметров для передачи значений в программах.</p>

9.	Работа над итоговым проектом	8	<p>Применять предметные знания и способы действий.</p> <p>Уметь самостоятельно планировать и управлять своей познавательной деятельностью во времени.</p> <p>Уметь использовать ресурсные возможности для достижения целей.</p> <p>Осуществлять выбор конструктивных стратегий в трудных ситуациях.</p> <p>Уметь ясно изложить и оформить выполненную работу, представить её результаты, аргументировано ответить на вопросы.</p>
----	------------------------------	---	---

Тематическое планирование 8 класс – 68 часов

№ пп	Название раздела	Количе ство часов	Основные виды деятельности обучающихся
1.	Повторение основных конструкций Python	6	<p>Знать структуру программы на Python, · режимы работы с Python.</p> <p>Использовать команды редактора программы.</p> <p>Организовывать ввод и вывод данных.</p> <p>Уметь решать задачи по теме "Условные операторы".</p> <p>Уметь реализовывать циклические алгоритмы, составлять программы с циклом.</p>
2.	Тип данных bool и NoneType	2	<p>Определять логический тип данных и NoneType.</p> <p>Задавать логические операторы.</p> <p>Представлять и использовать Булевы значения как числа.</p> <p>Использовать в программах функции bool(), type(), isinstance()</p>
3.	Вложенные списки	8	<p>Знать основы работы со списками.</p> <p>Использовать методы списков при решении задач.</p> <p>Уметь выполнять вывод элементов списка.</p> <p>Уметь использовать и различать строковые методы split() и join()</p> <p>Уметь решать задачи со списочными выражениями.</p> <p>Уметь использовать конструкцию вложенных списков при решении задач.</p>

4.	Кортежи	6	<p>Знать основы работы с кортежами. Использовать кортежи при решении задач. Уметь выполнять вывод кортежа. Знать методы кортежей. Уметь выполнять:</p> <ul style="list-style-type: none"> – Перебор кортежей – Сравнение кортежей – Сортировка кортежей – Преобразование кортежа в список и строку – Упаковка кортежей – Распаковка кортежей – Присваивание кортежей <p>Уметь использовать конструкцию вложенных кортежей при решении задач.</p>
5.	Множества	10	<p>Уметь определять множества в математике. Выполнять операции над множествами:</p> <ul style="list-style-type: none"> – Перебор – Сравнение – Сложение <p>Знать методы множеств и уметь использовать их в решении задач.</p>
6.	Словари	6	<p>операции создания, удаления, работы со словарями и их методы Знать основы работы со словарями. Использовать методы словарей при решении задач. Уметь выполнять вывод словаря. Уметь использовать и различать методы словарей. Уметь использовать конструкцию вложенных словарей при решении задач.</p>
7.	Модули decimal, fraction и другие	8	<p>Уметь работать с псевдослучайными числами. Знать и использовать методы модуля random. Использовать метод Монте-Карло для статических экспериментов и испытаний. Использовать дополнительные числовые типы данных модуля decimal:</p> <ul style="list-style-type: none"> – Decimal – десятичное число, для выполнения точных расчетов; – Fraction – число, представляющее собой обыкновенную дробь, с заданным числителем и знаменателем; – Complex – комплексное число. <p>Использовать числовой тип данных Fraction, для работы с обыкновенными дробями, с заданными числителем и знаменателем. Знать особенности модуля turtle и основные команды работы с исполнителем Черепашка. Создавать геометрические фигуры и анимированные объекты с помощью исполнителя Черепашка.</p>

8.	Функции	10	<p>Знать способы описания функции. Уметь использовать при решении задач принципы структурного программирования. Знать понятие формальных и фактических параметров подпрограмм, уметь использовать различные способы передачи параметров подпрограмм. Уметь создавать и использовать функции простые и рекурсивные и использовать их при решении задач. Уметь использовать механизм параметров для передачи значений в программах. Использовать для функций аргументы разных типов.</p>
9.	Работа с файлами	6	<p>Владеть основными понятиями, связанными с файлами, методы обращения к файлу. Уметь открывать и закрывать текстовый файл Обрабатывать данные из текстового файла. Выводить результат в файл.</p>
1.	Работа над итоговым проектом	6	<p>Применять предметные знания и способы действий. Уметь самостоятельно планировать и управлять своей познавательной деятельностью во времени. Уметь использовать ресурсные возможности для достижения целей. Осуществлять выбор конструктивных стратегий в трудных ситуациях. Уметь ясно изложить и оформить выполненную работу, представить её результаты, аргументировано ответить на вопросы.</p>

Тематическое планирование 9 класс - 68 часов.

№	Название раздела	Количес тво часов	Основные виды деятельности обучающихся
1.	Повторение основных конструкций Python	4	<p>Знать структуру программы на Python, · режимы работы с Python. Использовать команды редактора программы. Организовывать ввод и вывод данных. Уметь решать задачи по теме "Условные операторы". Уметь реализовывать циклические алгоритмы, составлять программы с циклом. Знать методы строк, списков, кортежей, множеств. Уметь организовывать в программе процедуру и функцию. Обрабатывать текстовые файлы.</p>
2.	Работа с датой и временем	6	<p>Знать и использовать типы данных: date, time, datetime, timedelta. В программе применять возможности модулей time, calendar.</p>

3.	Работа с файлами	8	Уметь использовать для чтения файлов потоковый ввод stdin и вывод stdout. Работать с файлами csv. Работать с файлами json. Работать с файлами zip. Работать с файлами pickle.
4.	Устройство памяти и особенности типов данных	5	Знать принципы устройства памяти в Python. Знать принципы работы переменных в Python. Различать изменяемые и неизменяемые типы данных. Использовать приемы поверхностного и глубокого копирования объектов. Использовать механизмы очистки памяти.
5.	Дополнительные типы коллекций	10	Знать и использовать типы данных: list, tuple, set, dict, namedtuple, defaultdict, OrderedDict, Counter, ChainMap.
6.	Обработка исключений	5	Различать три категории ошибок: <ul style="list-style-type: none"> – синтаксические – возникают из-за синтаксических погрешностей кода – логические – проявляются вследствие логических неточностей в алгоритме – ошибки времени выполнения, исключения – вызваны некорректными действиями пользователя или системы Уметь обрабатывать исключения.
7.	Рекурсия	4	Уметь организовывать рекурсивную обработку данных.
8.	Функции	8	Знать способы описания функции. Знать понятие формальных и фактических параметров подпрограмм, уметь использовать различные способы передачи параметров подпрограмм. Уметь создавать и использовать функции функции высшего порядка при решении задач. Использовать функции как объекты. Уметь использовать вложенные функции и замыкания. Применять модуль functools
9.	Итераторы и генераторы	10	Определять и обрабатывать итерируемые объекты. Использовать коллекции и последовательности. Создавать итераторы, генераторы. Отличать последовательность от итератора.
10.	Регулярные выражения	8	Использовать принцип соответствия символов. Использовать диапазон набора символов. Использовать принцип соответствия с чем угодно Кроме. Использовать принцип соответствия одному из нескольких регулярных выражений.

Критерии оценки предметных планируемых результатов обучающихся по курсу «Программирование»

Промежуточный и итоговый контроль проводится на онлайн платформе Stepik в форме защиты индивидуальных и групповых заданий по пройденному материалу. Отметочная система контроля отсутствует.

Для допуска к четвертному зачету необходимо выполнить 70% предложенных заданий. Один раз в четверть учащиеся сдают зачет по изученным темам.

Исходя из норм зачетной системы выставляется зачет:

«зачет» ставится при выполнении 50% от объема предложенных заданий;

«незачет» ставится, если допущены существенные ошибки, показавшие, что учащийся не владеет обязательными умениями по данной теме в полной мере (незнание основного программного материала).

Один раз в год обучающиеся представляют групповые и индивидуальные проекты. Проводится обсуждение результатов в коллективе с опорой на чек-лист, исправление ошибок коррекция и закрепление полученных знаний.