МБОУ "Степаненская СОШ"
Подписан цифровой подписью: МБОУ "Степаненская СОШ"
DN: O="МБОУ "Степаненская СОШ""", CN="МБОУ ""Степаненская СОШ""", E=steschool@yandex.ru
Основание: Я являюсь автором этого документа
Расположение:
Foxit PDF Reader Версия: 12.1.3

Муниципальное бюджетное общеобразовательное учреждение
«Степаненская средняя общеобразовательная школа»
Кезского района Удмуртской Республики

Рассмотрена на заседании методического совета школы
Протокол №___ от «___»_____20__ г.
Принята на заседании педагогического совета
Протокол № ___ от «___»_____20__ г.

Утверждено
Приказ № _____ от «___»_____ 20__г.
Директор школы: _____/ Пантелеева И.С./

Рабочая программа
учебного курса «Основы программирования» для учащихся 7 класса
на 2024-2025 учебный год
Составитель Бузмаков П.А., учитель МБОУ «Степаненская СОШ»

д.Степаненки

2024 г.

ПОЯСНИТЕЛЬНАЯ ЗАПИСКА

Примерная рабочая программа курса внеурочной деятельности «Основы программирования» (далее — курс) для 7 классов составлена на основе требований Федерального государственного образовательного стандарта основного общего образования к результатам освоения основной программы основного общего образования (Приказ Министерства просвещения Российской Федерации от 31.05.2021 № 287 «Об утверждении федерального государственного образовательного стандарта основного общего образования»), с учётом Примерной программы воспитания (протокол Федерального учебно-методического объединения по общему образованию № 3/22 от 23.06.2022) и Примерной основной образовательной программы основного общего образования (протокол Федерального учебно-методического объединения по общему образованию № 1/22 от 18.03.2022).

Примерная рабочая программа курса даёт представления о цели, задачах, общей стратегии обучения, воспитания и развития обучающихся средствами курса внеурочной деятельности, устанавливает содержание курса, предусматривает его структурирование по разделам и темам; предлагает распределение учебных часов по разделам и темам курса и последовательность их изучения с учётом межпредметных и внутрипредметных связей, логики учебного процесса, возрастных особенностей обучающихся, включает описание форм организации занятий и учебно-методического обеспечения образовательного процесса.

Примерная рабочая программа курса определяет количественные и качественные характеристики учебного материала для каждого года изучения, в том числе планируемые результаты освоения обучающимися программы курса внеурочной деятельности на уровне основного общего образования. Программа служит основой для составления поурочного тематического планирования курса внеурочной деятельности учителем.

ОБЩАЯ ХАРАКТЕРИСТИКА КУРСА ВНЕУРОЧНОЙ ДЕЯТЕЛЬНОСТИ «ОСНОВЫ ПРОГРАММИРОВАНИЯ НА PYTHON»

Программа курса внеурочной деятельности «Основы программирования» отражает:

- сущность информатики как научной дисциплины, изучающей закономерности протекания и возможности автоматизации информационных процессов в различных системах;
- основные области применения информатики, прежде всего информационные технологии, управление и социальную сферу;
- междисциплинарный характер информатики и информационной деятельности.

Информатика характеризуется всё возрастающим числом междисциплинарных связей, причём как на уровне понятийного аппарата, так и на уровне инструментария. Современная школьная информатика оказывает существенное влияние на формирование мировоззрения школьника, его жизненную позицию, закладывает основы понимания принципов функционирования и использования информационных технологий как необходимого инструмента практически любой деятельности и одного из наиболее значимых технологических достижений современной цивилизации. Многие предметные знания и способы деятельности, освоенные обучающимися при изучении информатики, находят применение как в рамках образовательного процесса при изучении других предметных областей, так и в иных жизненных ситуациях, становятся значимыми для формирования качеств личности, т. е. ориентированы на формирование мета- предметных и личностных результатов обучения.

Курс внеурочной деятельности отражает и расширяет содержание четырёх тематических разделов информатики на уровне основного общего образования:

- цифровая грамотность;
- теоретические основы информатики;
- алгоритмы и программирование;
- информационные технологии.

Целями изучения курса внеурочной деятельности «Основы программирования» являются:

- формирование основ мировоззрения, соответствующего со- временному уровню развития науки информатики, достижениям научно-технического прогресса и общественной практики, за счёт развития представлений об информации как о важнейшем стратегическом ресурсе развития личности, государства, общества; понимание роли информационных процессов, информационных ресурсов и информационных технологий в условиях цифровой трансформации многих сфер жизни современного общества;

- обеспечение условий, способствующих развитию алгоритмического мышления как необходимого условия профессиональной деятельности в современном информационном обществе, предполагающего способность обучающегося разбивать сложные задачи на более простые подзадачи; сравнивать новые задачи с задачами, решёнными ранее; определять шаги для достижения результата и т. д.;
- формирование цифровых навыков, в том числе ключевых компетенций цифровой экономики, таких как базовое программирование на Python, основы работы с данными, коммуникация в современных цифровых средах, информационная безопасность; воспитание ответственного и избирательного отношения к информации;
- формирование необходимых для успешной жизни в меняющемся мире универсальных учебных действий (универсальных компетентностей) на основе средств и методов информатики и информационных технологий, в том числе овладение умениями работать с различными видами информации, самостоятельно планировать и осуществлять индивидуальную и коллективную информационную деятельность, представлять и оценивать её результаты; формирование и развитие компетенций обучающихся в области использования информационно-коммуникационных технологий, в том числе знаний, умений и навыков работы с информацией, программирования, коммуникации в современных цифровых средах в условиях обеспечения информационной безопасности обучающегося; воспитание ответственного и избирательного отношения к ин-
- формации с учётом правовых и этических аспектов её распространения, стремления к продолжению образования в области информационных технологий и созидательной деятельности с применением средств информационных технологий.

Основные задачи курса внеурочной деятельности «Основы программирования на Python» — сформировать у обучающихся:

- понимание принципов устройства и функционирования объектов цифрового окружения, представления об истории и тенденциях развития информатики периода цифровой трансформации современного общества;
- владение базовыми нормами информационной этики и права, основами информационной безопасности;
- знания, умения и навыки грамотной постановки задач, возникающих в практической деятельности, их решения с помощью информационных технологий; умения и навыки формализованного описания поставленных задач;
- базовые знания об информационном моделировании, в том числе о математическом моделировании;
- знание основных алгоритмических структур и умение применять его для построения алгоритмов решения задач по их математическим моделям;
- умения и навыки составления простых программ по построенному алгоритму на Python;
- умения и навыки эффективного использования основных типов прикладных программ (приложений) общего назначения и информационных систем для решения с их помощью практических задач;
- умение грамотно интерпретировать результаты решения практических задач с помощью информационных технологий, применять полученные результаты в практической деятельности.

МЕСТО КУРСА ВНЕУРОЧНОЙ ДЕЯТЕЛЬНОСТИ «ОСНОВЫ ПРОГРАММИРОВАНИЯ НА PYTHON» В УЧЕБНОМ ПЛАНЕ

Программа курса предназначена для организации внеурочной деятельности за счёт направления «Дополнительное изучение учебных предметов». Программа курса внеурочной деятельности рассчитана на 34 учебных часа, по 1 ч в неделю в 7 классах.

Срок реализации программы внеурочной деятельности — один год.

Для класса предусмотрено резервное учебное время, которое может быть использовано участниками образовательного процесса в целях формирования вариативной составляющей содержания конкретной рабочей программы.

ПЛАНИРУЕМЫЕ РЕЗУЛЬТАТЫ ОСВОЕНИЯ ПРОГРАММЫ

ЛИЧНОСТНЫЕ РЕЗУЛЬТАТЫ

Патриотическое воспитание:

- ценностное отношение к отечественному культурному, историческому и научному наследию;
- понимание значения информатики как науки в жизни современного общества;
- заинтересованность в научных знаниях о цифровой трансформации 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 сети Интернет 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;
- стремление оценивать своё поведение и поступки своих товарищей с позиции нравственных и правовых норм с учётом осознания последствий поступков.

Формирование культуры здоровья:

- установка на здоровый образ жизни, в том числе и за счёт освоения и соблюдения требований безопасной эксплуатации средств ИКТ;
- соблюдение временных норм работы с компьютером.

Трудовое воспитание:

- интерес к практическому изучению профессий в сферах профессиональной деятельности, связанных с информатикой, программированием и информационными технологиями, основанных 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.

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 коррективы 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

К концу обучения обучающийся научится:

- соблюдать требования безопасности при работе на компьютере;
- объяснять, что такое информация, информационный процесс;

- перечислять виды информации;
- кодировать и декодировать сообщения по заданным правилам;
- переводить данные из одной единицы измерения информации в другую;
- характеризовать устройство компьютера;
- приводить примеры устройств для хранения и передачи ин формации;
- разбираться в структуре файловой системы;
- строить путь к файлу;
- объяснять, что такое алгоритм, язык программирования, программа;
- использовать переменные различных типов при написании программ на Python;
- использовать оператор присваивания при написании программ на Python;
- искать ошибки в программном коде на Python и исправлять их;
- дописывать программный код на Python;
- писать программный код на Python;
- использовать ветвления и циклы при написании программ на Python;
- анализировать блок-схемы и программы на Python;
- объяснять, что такое логическое выражение; 6 вычислять значение логического выражения; 6 записывать логическое выражение на Python; 6 понимать структуру адресов веб-ресурсов;
- форматировать и редактировать текстовую информацию в Google Документах;
- создавать презентации в Google Презентациях.

СОДЕРЖАНИЕ

Информация и информационные процессы (разделы «Цифровая грамотность» и «Теоретические основы информатики»)

Техника безопасности и правила работы на компьютере. Информация и информационные процессы. Виды информации. Хранение информации. Устройства для работы

с информацией. Единицы измерения информации Файловая система. Путь к файлу. Операции с файлами.

Основы языка программирования Python (раздел «Алгоритмы и программирование»)

Современные языки программирования. Алгоритм. Язык программирования. Программа. Среда разработки IDE. Интерфейс Sculpt. Виды алгоритмов: линейный, разветвляющийся. Переменные. Правила образования имён переменных. Типы данных: целое число, строка. Функция. Виды функций. Функция: print(), input(), int(). Ветвление в Python. Оператор if-else. Вложенное ветвление. Множественное ветвление. Оператор if-elif-else.

Циклы в языке программирования Python (раздел «Алгоритмы и программирование»)

Логическое выражение. Простые и сложные логические выражения. Результат вычисления логического выражения. Условие. Операции сравнении в Python. Логические операторы в Python: and, or и not. Операторы целочисленного деления и деления с остатком на Python. Проект «Максимум и минимум».

Информационная безопасность. Информационная безопасность. Основные типы угроз в Интернете. Правила поведения в Интернете. Кибербуллинг. Защита приватных данных.

КРИТЕРИИ ОЦЕНКИ ЗНАНИЙ

Критерии оценивания программы

Структура и оформление программы (17 баллов)

- крайне рекомендуется вообще не использовать в программе глобальные переменные;
- все глобальные переменные (если они вообще необходимы) должны иметь имена достаточно длинные, чтобы по имени можно было сразу понять назначение переменной (например, `countDots` или `lineA`, а не `a`, `b`, `c`);
- в списке глобальных переменных не должны присутствовать переменные, имя которых состоит из одного или двух символов;
- все счетчики циклов и временные переменные должны быть описаны как локальные (под временными переменными имеются ввиду те переменные, которые описывают не глобальное состояние, и/или выполняющие функцию хранения в течение ограниченного времени (по отношению ко всей программе));
- каждая подпрограмма должна иметь такое достаточно длинное имя, чтобы по нему можно было сразу понять, какое действие она (подпрограмма) делает (например, `printAxis` или `solveTask`, а не `myfunc1`);
- главная программа и каждая подпрограмма должна занимать не более 25 строк кода (если необходимо написать более длинную подпрограмму, используйте ее разбиение на несколько отдельных подпрограмм);
- выделение в отдельный модуль (файл) каких-то подпрограмм поощряется дополнительными баллами;
- каждый параметр каждой подпрограммы должен иметь достаточно длинное и понятное имя, по которому можно сразу понять, что это за параметр;
- вся программа должна быть оформлена с использованием одного, единого Code Style. Рекомендуется к использованию Code Style, выданный учителем и находящийся в общем доступе. При использовании собственного Code Style необходимо сформулировать его правила, согласовать его непротиворечивость с учителем, и приложить к проекту в качестве приложения отдельным документом. Пример рекомендуемого Code Style для языка программирования Паскаль см. в приложении;
- каждый смысловой фрагмент кода программы должен иметь комментарий, поясняющий смысл совершаемого действия. Исключение — оформление смыслового фрагмента в виде отдельной подпрограммы, из имени которой понятен смысл совершаемого ею действия.

Критерий	**Оценка**
Используются без необходимости глобальные переменные. В спорных случаях оправданность использования глобальных переменных должна быть обоснована в отчете. Рекомендуется, однако, не использовать глобальные переменные	1 балл
Все глобальные переменные (если они вообще необходимы) имеют имена достаточно длинные, чтобы по имени можно было сразу понять назначение переменной (например, `countdots` или `linea`, а не `a`, `b`, `c`)	2 балла
В списке глобальных переменных не присутствуют переменные, имя которых состоит из одного или двух символов	1 балл
Все счетчики циклов и временные переменные описаны как локальные	1 балл
Каждая подпрограмма имеет такое достаточно длинное имя, чтобы по нему можно было сразу понять, какое действие она (подпрограмма) делает (например, `printaxis` или `solvetask`, а не `myfunc1`)	2 балла
Главная программа и каждая подпрограмма занимают не более 25 строк кода	3 балла
Каждый параметр каждой подпрограммы* имеет достаточно длинное и понятное имя, по которому можно сразу понять, что это за параметр. * За исключением случаев, когда по названию подпрограммы смысл параметра очевиден.	2 балла
Вся программа оформлена с использованием одного, единого Code Style	2 балл
Используется Code Style, выданный учителем и находящийся в общем доступе	2 балла
Используется собственный, непротиворечивый Code Style	0,5 балла

При использовании собственного CodeStyle сформулированы его правила, они согласованы на непротиворечивость с учителем	0,5 балла
При использовании собственного CodeStyle его правила приложены к проекту в качестве приложения отдельным документом	1 балл
Каждый смысловой фрагмент кода программы имеет комментарий, поясняющий смысл совершаемого действия. Исключение — оформление смыслового фрагмента в виде отдельной подпрограммы, из имени которой понятен смысл совершаемого ею действия.	2 балла

Критерии оценивания реализации программы (для типового проекта) (21 балл)

Критерий	**Оценка**
В программе предусмотрено меню, предлагающее выбрать способ задания исходных данных	1 балл
Исходные данные можно задавать, вводя данные с клавиатуры	1 балл
Исходные данные можно задать из заранее подготовленного файла (то есть, имеется чтение данных из файла, и этот файл существует, заполнен и приложен при сдаче проекта)	1 балл
Исходные данные можно задать из одного из нескольких заранее подготовленных файлов (то есть, интерфейс программы предлагает возможность выбора чтения данных из нескольких различных файлов, и эти файлы существуют, заполнены и приложены при сдаче проекта)	1 балл
При задании данных из файла можно выбрать имя файла (то есть, интерфейс программы предлагает возможность указать произвольное имя файла для чтения входных данных)	1 балл
Исходные данные можно задать наглядно и визуально, при помощи мыши	2 балл
Задание данных при помощи мыши удобно и понятно	2 балл
Задание данных при помощи мыши правильно сохраняет данные	1 балл
Для хранения данных о составных объектах (например, о координатах точки на плоскости или о параметрах фигуры) используются записи/структуры/классы	1 балл
Эти записи/структуры/классы созданы для каждого вида объектов программы	1 балл
Корректность рассмотрения всех возможных случаев. Оценка ставится в зависимости от качества используемого алгоритма. При нерассмотренных случаях количество баллов снижается	8 баллов
Эффективность рассмотрения всех возможных случаев. Оценка ставится в зависимости от качества используемого алгоритма. При очевидной (для оценивающего) избыточности рассматриваемых случаев оценка снижается. При использовании нестандартного эффективного решения возможно увеличение оценки на 4 балла	8 баллов
Выводится на экран результат	1 балл
Результат выводится на экран на координатной плоскости и визуально выделен	2 балл
Выводятся также все необходимые характеристики визуализированного результата	1 балл
Программой удобно и понятно пользоваться	1 балл

Критерии оценивания реализации программы (для творческого проекта) (27 баллов)

Для некоторых творческих проектов, ввиду большого диапазона их видов, возможны ситуации, когда применение некоторых критериев оценивания становится невозможным. В этом случае комиссия может принять решение исключить какие-то требования критериев оценивания из общего списка оценивания конкретного проекта.

В общем случае, творческие проекты оцениваются комиссией/жюри по следующим категориям:

	0 баллов	**1 балл**	**2 балла**	**3 балла**
Красивость, визуальное	ужасно	среднее	хорошо	исключительно
Оформление				

Законченность	Не запускается даже у разработчика	Работает для некоторых частных случаев	Основной функционал работает	Что-то невероятное
Собственная разработка/ алгоритм	Алгоритм является неизмененной копией туториала/решения с форума и тд.	Копипаст с пониманием происходящего	Копипаст с элементами личного творчества	Полностью оригинальный код/ реализован новый алгоритм.
Исследование	Полностью отсутствует элемент исследования	Хоть какое-нибудь исследование	Что-то среднее	Исчерпывающее исследование
Практическая применимость	Не имеет никакой	Имеет потенциальное применение в редких случаях	Имеет ограниченное применение	Имеет реальное применение (или внешний заказ, например, от учителя)
Использование сторонних технологий, требующих доп. изучения (пример: сетевое взаимодействие, специальная библиотека, 3Dбиблиотека, Unity, …)	Используется, но человек не понимает этого (зачем и как ты это делаешь)	Не используется	Используется ограниченно, но с пониманием того, зачем и как	Используется полноценно, в большом объеме
Выполнена ли поставленная задача	Не выполнена, или незначительно	Задача выполнена значительно, но не вся	Поставленная задача выполнена (или доказано, что это невозможно)	Поставленная задача выполнена и перевыполнена
Ответы на вопросы	На вопросы не отвечает, ничего не понимает	Отвечает на часть вопросов, на часть вопросов не отвечает или "плавает"	Хорошо отвечает на вопросы	Прекрасно отвечает на вопросы, заранее подготовил картинки/схемы
Общее ощущение, включая доклад и презентацию	Ужасное	Ниже среднего	Среднее	Полный восторг

Критерии оценивания и требования к оформлению презентации

Максимальное количество баллов по пунктам

Титульный слайд (название, подзаголовок, автор)	
Постановка задачи	
Визуализация постановки задачи	
Входные и выходные данные	
Математическая модель	
Визуализация структуры данных	

Визуализация метода решения	
Структура программы	
Пример работы программы	
Визуализация результата	
Возникшие затруднения	
Финальный слайд	
Эргономика презентации	
Всего максимум	**6**

Основные правила, ко всей презентации

В презентации **нигде** не должно быть ни одного слова "презентация"!

Презентация должна помогать Вашему докладу, визуально представлять то, что Вы рассказываете.

В частности, на каждом слайде должна быть картинка. Либо она должна демонстрировать то, про что слайд рассказывать, либо (как в случае с титульным и финальным слайдами, например) содержать изображение решаемой задачи. Например, схему или логотип.

Титульный слайд (4 балла)

Должен содержать:

- Название (1 балл)
- Подзаголовок (1 балл)
- Автор (1 балл)
- Руководитель, учитель информатики (если он не является руководителем) (1 балл)

Название — это не "Проект по информатике" и не "Презентация проекта по информатике", а красивое название, отражающее содержание Вашей работы. Например, "Нахождение длины медианы треугольника".

Подзаголовок — это пояснение, про что именно презентация. Например, "Годовой проект по информатике". Еще раз — не "Презентация годового проекта по информатике"! Вы не презентацию представляете, а проект представляете, используя в помощь презентацию.

Автор — это, собственно, Вы. Например, "Иванов Иван, 10-8 класс". Руководитель — тот человек, который курировал Ваш проект.

Учитель информатики — указать отдельно, если учитель информатики не является руководителем проекта.

Критерий	**Оценка**
Пункт отсутствует совсем	0 баллов
Присутствует название проекта	0,5 балла
Название проекта сформулировано понятно и отражает смысл проекта	0,5 балла
Присутствует подзаголовок, который поясняет, про что эта презентация	1 балл
Присутствует Фамилия Имя, класс автора	1 балл
Присутствует Фамилия Имя Отчество руководителя проекта	0,5 балла
Присутствует Фамилия Имя Отчество учителя информатики	0,5 балла

Постановка задачи (3 балла), визуализация постановки задачи (5 баллов)

Вторым слайдом после титульного должен быть слайд, демонстрирующий решаемую задачу. Возможно, это будет несколько слайдов.

Типичная ошибка — создание слайда, в заголовке которого написано "Визуализация постановки задачи". Слайд должен называться "Постановка задачи". А вот содержимое этого слайда должно максимально наглядно демонстрировать, что нужно было сделать. Это и называется "визуализация".

Еще раз. Самое главное, что на этом сладе (2-х слайдах) должно быть — изображение (-я), позволяющее (-щие) понять, какую задачу Вы решали. Если при этом хочется/кажется нужным добавить текст, излагающий постановку задачи — добавьте. Главное — визуализация. При хорошей картинке и текст не нужен.

Критерий	Оценка
Пункт отсутствует совсем	0 баллов
На каждом слайде присутствует заголовок "Постановка задачи"	1 балл
Присутствует картинка, поясняющая, что должна была делать программа	1 балл
Картинка/последовательность/совокупность картинок полностью и понятно отображает то, что должна делать программа	3 балла
Присутствует рисунок, показывающий планируемый на этапе постановки задачи внешний вид программы. Этот рисунок не может быть скриншотом готовой программы	2 балла
Присутствует (при необходимости) текст или дополнительные картинки/схемы, поясняющие постановку задачи	1 балл

Входные и выходные данные (3 балла)

Нужно описать, что дается на вход. Желательно (опять же) обойтись без текста или почти без текста и показать картинками, что дается на входе, а что — на выходе. При необходимости, используйте несколько слайдов. При этом нужно сообщить, какой тип данных используется для каких входных параметров, сколько их, к чему они относятся. Без картинки — не более 1,5 баллов. И то, если все-все перечисленное указано.

Критерий	Оценка
Пункт отсутствует совсем	0 баллов
На каждом слайде присутствует заголовок "Входные данные", "Выходные данные" или "Входные и входные данные"	1 балл
На каждом слайде присутствует картинка, поясняющая и демонстрирующая, какие данные подаются на вход программы и какие данные получаются на выходе	1 балл
Присутствует описание или визуализация количества и типа входных и выходных данных	по 0,5 балла

Математическая модель (4 балла)

Приведите все формулы, которые используются для решения задачи. К каждой формуле приведите расшифровку, что какое обозначение означает. Крайне рекомендуется при этом демонстрировать картинкой каждую формулу, с изображением на ней тех величин, которые используются в формуле. Без картинок к каждой формуле — не более 1,5 баллов. При необходимости, используйте несколько слайдов.

Формулы должны быть оформлены по правилам математики. В частности, переменные должны быть набраны курсивом, индексы должны быть нижними индексами, произведение должно быть обозначено центральной точкой или опущено, знак корня должен располагаться над всем подкоренным выражением, возведение в квадрат/степень должно быть верхним индексом, минус должен быть минусом, а не дефисом. Лучше всего — использовать для вставки формулы специальный модуль для набора формул. Например, MS Equation, MathType или любое другое средство, кроме простого текста.

Критерий	Оценка
Пункт отсутствует совсем. Если для задачи не требуется математическая модель, баллы за этот раздел не учитываются при расчете итоговой оценки	0 баллов
Присутствует заголовок "Математическая модель"	0,5 балла
Присутствуют все формулы, которые используются при решении задачи	0,5 балла
Все формулы оформлены красиво, по правилам математики	1 балл
Возле каждой формулы имеется картинка, демонстрирующая используемые в формуле величины и их взаиморасположение	1 балл

Возле каждой формулы приведена расшифровка всех величин, используемых в формуле (обозначение и смысл)	1 балл

Структура данных (3 балла)

Нарисуйте, какие данные Вы храните в какой структуре. Массивы — таблицами, отдельные ячейки — отдельными прямоугольниками. Естественно, подпишите массивы и ячейки-переменные.

Рекомендуется нарисовать картинку с подписанными исходными данными (координаты, вектора, величины). От каждого обозначения картинки — стрелка к ячейке, в которой это значение этой величины хранится. Если из этих данных потом выбираются отдельные данные в другую структуру данных — тоже изобразите это стрелками.

Критерий	**Оценка**
Пункт отсутствует совсем	0 баллов
Присутствует заголовок "Структура данных"	0,5 балла
Присутствует понятная картинка с обозначениями исходных данных	1 балл
Присутствует картинка с обозначениями всех хранимых данных	0,5 балла
Присутствует понятная визуализация всей структуры в комплексе (ничего не потеряно, все наглядно и понятно)	1 балл

Метод решения (5 баллов)

Изобразите, как решалась Ваша задача. Как из входных данных произошли какие-то отборы/преобразования/вычисления/поиск, потом еще что-то, в результате чего Вы пришли к ответу. Наибольшие 5 баллов — только за подробную и понятную схему, где все это нарисовано, от картинки со входными данными, через стрелки и обозначения нужных обработок, к конечному результату.

Критерий	**Оценка**
Пункт отсутствует совсем	0 баллов
Присутствует заголовок "Метод решения"	0,5 балла
Отсутствуют стрелки, показывающие последовательность решения задачи	max 2 балла за раздел
Отсутствует рисунок исходных данных	max 1 балл за раздел
Отсутствует схема с последовательным отображением всех этапов	max 2 балла за раздел
Присутствует рисунок исходных данных	1 балл
Присутствует последовательность рисунков, наглядно и понятно демонстрирующих преобразования/вычисления	1,5 балла
Присутствует рисунок конечного результата	1 балл
Присутствуют стрелки, наглядно и правильно показывающие последовательность решения задачи	1 балл

Структура программы (4 балла)

Изобразите в виде схемы, каковы подпрограммы, из которых состоит Ваша программа, и (стрелками) какова схема их вызовов/взаимодействия. Название каждой подпрограммы поместите в отдельный прямоугольник. Между прямоугольниками расположите стрелки (или линии, если переход происходит сверху вниз). Обозначены не все использованные подпрограммы — не более 1 балла.

Критерий	**Оценка**
Пункт отсутствует совсем	0 баллов
Присутствует заголовок "Структура программы"	0,5 балла
На схеме присутствуют все подпрограммы, реализованные в программе	1,5 балла
Каждая подпрограмма, реализованная в программе, изображена на схеме в отдельном	0,5 балла

прямоугольнике	
На схеме отображены линиями/стрелками все вызовы, зависимости и взаимодействия (разным цветом), которые используются в задаче	1 балл
На всех линиях, которые требуются для описания вызовов/взаимодействий и которые направлены не вниз, присутствует стрелка, показывающая направление вызова/взаимодействия	0,5 балла

Пример работы программы (2 балла)

Критерий	**Оценка**
Пункт отсутствует совсем	0 баллов
Присутствует заголовок "Пример работы программы"	0,5 балла
Присутствуют скриншоты рабочей программы	0,5 балла
Присутствует скриншот с исходными данными	0,5 балла
Присутствует скриншот с отображением результата работы программы	0,5 балла

Визуализация результата (3 балла)

Этот пункт может быть отображен на слайде "Пример работы программы" или на отдельном слайде "Отображение результата"

Критерий	**Оценка**
Пункт отсутствует совсем	0 баллов
Присутствует визуально представленный результат	1 балл
Это визуальное представление результата красиво и понятно оформлено, выделено для наглядности	1 балл
Это визуальное представление результата содержит отображение всех найденных величин	1 балл

Возникшие затруднения (3 балла)

Редкий случай, когда нужен текст. Для каждого затруднения указать: описание собственно затруднения, метод решения, картинку.

Критерий	**Оценка**
Пункт отсутствует совсем	0 баллов
Присутствует заголовок "Возникшие затруднения"	0,5 балла
Присутствует описание хотя бы одного возникшего затруднения	0,5 балла
Присутствует описание всех возникших затруднений	0,5 балла
Для каждого затруднения указано: описание самого затруднения	1 балл
Для каждого затруднения указано: метод решения, картинка	1,5 балла
За утаивание затруднения (если учитель знает, что учащийся затруднялся выполнить какой-то этап, спрашивал об этом учителя, но при этом не привел его здесь)	max 1,5 балла за раздел

Финальный слайд (2 балла)

Написать что-нибудь вроде "Спасибо за внимание!". Украсить слайд картинкой. Указать информацию об авторе (Фамилия, имя, класс, e-mail).

Критерий	**Оценка**
Пункт отсутствует совсем	0 баллов
Присутствует заголовок "Спасибо за внимание" или что-то подобное, позволяющее однозначно понимать, что это финальный слайд	0,5 балла
Присутствует картинка (например, визуализация решенной задачи или хотя бы фотография автора)	0,5 балла

Присутствует информация об авторе (Фамилия, имя, класс, e-mail)	1 балл

Эргономика (5 баллов)

Это балл за общее ощущение от презентации, за приятность ее восприятия. В частности, за невызывающий, неконтрастный фон или схему оформления. Единый стиль слайдов для всей презентации. Использование текста — только по необходимости. Заголовок на каждом слайде.

Критерий	**Оценка**
За общее ощущение от презентации, max	5 баллов
За каждое из следующего: •контрастный фон или схема оформления; •разный стиль для разных слайдов; •разный шрифт (гарнитура) для разных слайдов; •необоснованно большое количество текста; •отсутствие заголовка хотя бы на одном слайде; •наличие раздражающей анимации хотя бы на одном слайде; •наличие раздражающего звукового оформления хотя бы на одном слайде; •наличие в презентации хотя бы одного из слов "Презентация" или "Визуализация".	–1 балл

ПЛАН МЕРОПРИЯТИЙ В РАМКАХ РЕАЛИЗАЦИИ ПРОГРАММЫ ВОСПИТАНИЯ

№ п/п	Мероприятие	Ответственный	Сроки проведения	Примечание
1	Проведение школьного этапа ВсОШ	Бузмаков П.А	сентябрь-октябрь	
2	Дни открытых дверей	администрация	феврвль	
3	Неделя информатики	Бузмаков П.А.	ноябрь	
4	Урок цифры	Бузмаков П.А.	сентябрь - май	
5	Подготовка к олимпиадам	Бузмаков П.А.	сентябрь - май	

ТЕМАТИЧЕСКОЕ ПЛАНИРОВАНИЕ

№ п/п	Наименование разделов и тем программы	Количество часов			Электронные (цифровые) образовательные ресурсы
		Всего	Контрольные работы	Практические работы	
Раздел 1. Информация и информационные процессы					
1.1	Информация и информационные процессы	4			https://znanio.ru/media/informatika-7-klass
1.2	Файлы и папки	2			https://znanio.ru/media/informatika-7-klass
Итого по разделу		6			
Раздел 2. Основы языка программирования Python					
2.1	Знакомство с языком программирования Python	2			https://znanio.ru/media/informatika-7-klass
2.2	Типы данных. Переменные	3			https://znanio.ru/media/informatika-7-klass-73a-tipy-dannyh-kratkosrochnyj-plan-2545703
2.3	Ввод и вывод данных	3			https://znanio.ru/media/informatika-7-klass
2.4	Ветвление	2			https://znanio.ru/media/informatika-7-klass
2.5	Проект «Чат-бот»	2		1	https://znanio.ru/media/informatika-7-klass
Итого по разделу		12			
Раздел 3. Циклы в языке программирования Python					
3.1	Логические выражения и операторы	3			https://znanio.ru/media/informatika-7-klass
3.2	Циклы	3			https://znanio.ru/media/informatika-7-klass-73a-tipy-dannyh-kratkosrochnyj-plan-2545703
3.3	Проект «Максимум и минимум»	3		1	https://znanio.ru/media/informatika-7-klass-73a-tipy-dannyh-kratkosrochnyj-plan-2545703
Итого по разделу		9			
Раздел 4. Информационные технологии					
4.1	Работа в Интернете	2	0		https://znanio.ru/media/informatika-7-klass
4.2	Обработка различных видов информации	2			https://znanio.ru/media/informatika-7-klass
4.3	Проект «Презентация ElevatorPitch»	3		1	https://znanio.ru/media/informatika-7-klass
Итого по разделу		7			
ОБЩЕЕ КОЛИЧЕСТВО ЧАСОВ ПО ПРОГРАММЕ		34	0	3	

ПОУРОЧНОЕ ПЛАНИРОВАНИЕ

№ п.п	Тема урока	Количество часов			Воспитательный компонент содержания программы
		Всего	Контрольные работы	Практические работы	
1	Техника безопасности и правила работы на компьютере. Информация и информационные процессы. Виды информации.	1			побуждение школьников соблюдать на уроке общепринятые нормы поведения, правила общения со старшими (учителями) и сверстниками (школьниками), принципы учебной дисциплины и самоорганизациипривлечение внимания школьников к ценностному аспекту изучаемых на уроках явлений, организация их работы с получаемой на уроке социально значимой информацией – инициирование ее обсуждения, высказывания учащимися своего мнения по ее поводу, выработки своего к ней отношения; применение на уроке интерактивных форм работы учащихся: интеллектуальных игр, стимулирующих познавательную мотивацию школьников; дидактического театра, где полученные на уроке знания обыгрываются в театральных постановках; дискуссий, которые дают учащимся возможность приобрести опыт ведения конструктивного диалога; групповой работы или работы в парах, которые учат школьников командной работе и взаимодействию с другими детьми, уроки - квесты.
2	Хранение информации. Устройства для работы с информацией.	1			
3	Единицы измерения информации	1			
4	Кодирование информации. Код. Процессы кодирования и декодирования	1			
5	Файловая система. Путь к файлу. Операции с файлами	1			
6	Одноуровневая и многоуровневая файловые структуры	1			
7	Современные языки программирования. Алгоритм. Язык программирования Python	1			
8	Среда разработки IDE. Интерфейс Sculpt. Виды алгоритмов: линейный, разветвляющийся	1			
9	Типы данных: целое число, строка	1			
10	Переменные.	1			
11	Правила образования имён переменных	1			
12	Функция.	1			

13	Виды функций.	1			
14	Функция: print(), input(), int()	1			
15	Ветвление в Python. Оператор if-else.	1			
16	Множественное ветвление. Оператор if-elif-else	1			
17	Цель проекта. Задачи проекта. Чат-бот.	1			
18	Создание проекта	1		**1**	
19	Логическое выражение. Простые и сложные логические выражения.	1			
20	Результат вычисления логического выражения	1			
21	Условие. Операции сравнения в Python. Логические операторы	1			
22	Понятие и определение циклы	1			
23	Цикл с предусловием.	1			
24	Цикл с параметром	1			
25	Статистика. Примеры статистических моделей.	1			
26	Формула вычисления среднего	1			
27	Функции для вычисления максимального и минимального значения	1			
28	Средства коммуникации. Современные средства общения. Всемирная паутина (WWW).	1			
29	Назначение браузера. Создание почтового ящика. Облачное хранилище. Правила безопасности в Интернете	1			
30	Текстовая информация в реальной жизни.	1			

31	Обработка текстовой информации.	1			
32	Свойства и правила хорошей презентации	1			
33	Создание проекта	1			
34	Особенности презентации типа «ElevatorPitch»	1			
ОБЩЕЕ КОЛИЧЕСТВО ЧАСОВ ПО ПРОГРАММЕ		34	1	0	