УПРАВЛЕНИЕ ОБРАЗОВАНИЯ, МОЛОДЕЖИ И СПОРТА
АДМИНИСТРАЦИИ БАХЧИСАРАЙСКОГО РАЙОНА
РЕСПУБЛИКИ КРЫМ

МУНИЦИПАЛЬНОЕ БЮДЖЕТНОЕ ОБРАЗОВАТЕЛЬНОЕ УЧРЕЖДЕНИЕ
ДОПОЛНИТЕЛЬНОГО ОБРАЗОВАНИЯ
«ЦЕНТР ДОПОЛНИТЕЛЬНОГО ОБРАЗОВАНИЯ ДЕТЕЙ»
БАХЧИСАРАЙСКОГО РАЙОНА РЕСПУБЛИКИ КРЫМ

ОДОБРЕНО
педагогическим советом
протокол № _3_
от «_31_»__03___ 2026 год

УТВЕРЖДАЮ
Директор МБОУ ДО «ЦДОД»
_____ П.С.Усатенко
«___» 2026 год
Приказ № 81

ДОПОЛНИТЕЛЬНАЯ ОБЩЕОБРАЗОВАТЕЛЬНАЯ ОБЩЕРАЗВИВАЮЩАЯ ПРОГРАММА ТВОРЧЕСКОГО ОБЪЕДИНЕНИЯ «ЦИФРОВОЙ МИР» «Программирование на Python»

Направленность: техническая
Возраст обучающихся: 12-17 лет
Срок реализации программы:1 год (144 часа)
Уровень: стартовый
Разработчик:
Мосин Сергей Николаевич,
педагог дополнительного образования
творческого объединения «Цифровой мир»

г. Бахчисарай,
2026 г.

Оглавление

РАЗДЕЛ № 1. ПОЯСНИТЕЛЬНАЯ ЗАПИСКА 3
Направленность программы. 3
Нормативно-правовая база 3
Цель программы 5
Задачи 5
Адресат программы 5
Актуальность программы 6
Педагогическая целесообразность 6
Формы обучения 7
Особенности организации 7
Режим занятий 8
Уровень программы 8
Объем и сроки освоения программы. 8
Формы аттестации/контроля 8
РАЗДЕЛ № 2. УЧЕБНЫЙ ПЛАН 10
РАЗДЕЛ № 3. КАЛЕНДАРНЫЙ УЧЕБНЫЙ ГРАФИК 15
РАЗДЕЛ № 4. СОДЕРЖАНИЕ ПРОГРАММЫ. 16
РАЗДЕЛ № 5. ВОСПИТАНИЕ. 23
РАЗДЕЛ № 6 ПЛАНИРУЕМЫЕ РЕЗУЛЬТАТЫ 29
РАЗДЕЛ № 7. РЕСУРСНОЕ ОБЕСПЕЧЕНИЕ. 32
Кадровое обеспечение. 32
Материально-техническое обеспечение: 32
Методическое обеспечение. 32
Информационно-телекоммуникационное обеспечение 34
РАЗДЕЛ № 8. СПИСОК ЛИТЕРАТУРЫ. 35
РАЗДЕЛ № 9. ПРИЛОЖЕНИЯ. 38
Оценочные материалы 38
Методические материалы 47
КАЛЕНДАРНО-ТЕМАТИЧЕСКОЕ ПЛАНИРОВАНИЕ 58
ЛИСТ КОРРЕКТИРОВКИ 64

РАЗДЕЛ №1

ПОЯСНИТЕЛЬНАЯ ЗАПИСКА

Дополнительная общеобразовательная общеразвивающая программа «Программирование на Python» создает условия для формирования у подростков алгоритмического и логического мышления, воспитывает уважение к интеллектуальному труду и способствует развитию проектных компетенций. Программа позволяет не только подготовить обучающегося к дальнейшему освоению IT-дисциплин, но и сформировать основы цифровой грамотности, научить структурировать информацию и создавать собственные программные продукты.

Направленность программы. Дополнительная общеобразовательная общеразвивающая программа «Программирование на Python» имеет техническую направленность. Она способствует развитию инженерного, пространственного обучающихся через освоение современных алгоритмов

Нормативно-правовой основой разработки и реализации дополнительной общеобразовательной общеразвивающей программы технической направленности. Программа содержит комплекс документов федерального, регионального и локального уровней, обеспечивающих правовое, организационное и методическое сопровождение образовательного процесса:

— Федеральный закон от 29.12.2012 г. № 273-ФЗ «Об образовании в Российской Федерации» (в действующей редакции);

— Федеральный закон Российской Федерации от 24.07.1998 г. № 124-ФЗ «Об основных гарантиях прав ребенка в Российской Федерации» (в действующей редакции);

— Федеральный закон Российской Федерации от 13.07.2020 г. № 189-ФЗ «О государственном (муниципальном) социальном заказе на оказание государственных (муниципальных) услуг в социальной сфере» (в действующей редакции);

— Указ Президента Российской Федерации от 07.05.2024 г. № 309 «О национальных целях развития России до 2030 года и на перспективу 2036 года»;

— Распоряжение Правительства Российской Федерации от 31.03.2022 г. № 678-р «Об утверждении Концепции развития дополнительного образования детей до 2030 года» (в действующей редакции);

— Постановление Главного государственного санитарного врача Российской Федерации от 28.09.2020 г. № 28 «Об утверждении санитарных правил СП 2.4.3648-20 «Санитарно-эпидемиологические требования к организациям воспитания и обучения, отдыха и оздоровления детей и молодежи»;

— Постановление Главного государственного санитарного врача Российской Федерации от 28.01.2021 г. № 2 «Об утверждении санитарных

правил и норм СанПиН 1.2.3685-21 «Гигиенические нормативы и требования к обеспечению безопасности и (или) безвредности для человека факторов среды обитания» (в действующей редакции);

— Приказ Министерства просвещения Российской Федерации от 27.07.2022 г. № 629 «Об утверждении Порядка организации и осуществления образовательной деятельности по дополнительным общеобразовательным программам» (в действующей редакции);

— Приказ Минпросвещения России от 03.09.2019 г. № 467 «Об утверждении Целевой модели развития региональных систем развития дополнительного образования детей» (в действующей редакции);

— Приказ Минобрнауки России № 882 и Минпросвещения России № 391 от 05.08.2020 г. «Об организации и осуществлении образовательной деятельности при сетевой форме реализации образовательных программ» (в действующей редакции);

— Приказ Министерства труда и социальной защиты Российской Федерации от 22.09.2021 г. № 652н «Об утверждении профессионального стандарта «Педагог дополнительного образования детей и взрослых»;

— Закон Республики Крым от 06.07.2015 г. № 131-ЗРК/2015 «Об образовании в Республике Крым» (в действующей редакции);

— Распоряжение Совета министров Республики Крым от 11.08.2022 г. № 1179-р «О реализации Концепции дополнительного образования детей до 2030 года в Республике Крым» (в ред. от 25.08.2025 № 1338-р);

— Приказ Министерства образования, науки и молодежи Республики Крым от 03.09.2021 г. № 1394 «Об утверждении моделей обеспечения доступности дополнительного образования для детей Республики Крым»;

— Приказ Министерства образования, науки и молодежи Республики Крым от 09.12.2021 г. № 1948 «О методических рекомендациях «Проектирование дополнительных общеобразовательных общеразвивающих программ»;

— Приказ Министерства образования, науки и молодежи Республики Крым от 17.02.2026 г. № 281 «Об утверждении Требований к условиям и порядку оказания государственной услуги в социальной сфере «Реализации дополнительных общеразвивающихся программ» в Республике Крым в 2026 году»;

— Методические рекомендации для педагогических работников образовательных организаций общего образования, образовательных организаций среднего профессионального образования, образовательных организаций дополнительного образования по использованию российского программного обеспечения при взаимодействии с обучающимися и их родителями (законными представителями), письмо Минпросвещения России от 31.07.2023 № 04-423 «Об исполнении протокола»;

— Письмо Министерства просвещения Российской Федерации от 23.01.2026 № АБ-254/06 «Методические рекомендации по разработке

дополнительных общеразвивающих программ, в том числе в части интеграции с учебными предметами «Труд (технология)», «Музыка», «Изобразительное искусство», «Физическая культура»;

— Устав муниципального бюджетного образовательного учреждения дополнительного образования «Центр дополнительного образования детей» Бахчисарайского района Республики Крым, утвержденный постановлением администрации Бахчисарайского района Республики Крым от 22.08.2018г. № 438;

— Положение о разработке и утверждении дополнительной общеобразовательной общеразвивающей программы Муниципального бюджетного образовательного учреждения дополнительного образования «Центр дополнительного образования детей» Бахчисарайского района Республики Крым, утвержденное приказом от 31.03.2026г. № 81.

Цель программы - формирование базовых алгоритмических и технических компетенций обучающихся 12-17 лет через освоение языка программирования Python, развитие логического мышления и создание условий для первичной профессиональной ориентации в сфере IT.

Задачи:

Образовательные:

- ознакомить с базовыми конструкциями Python (переменные, условия, циклы, функции, списки);
- научить составлять и отлаживать алгоритмы решения практических задач.

Развивающие:

- развивать алгоритмическое, логическое и пространственное мышление;
- формировать навыки поиска и исправления ошибок (дебаггинг);
- стимулировать креативность через создание мини-игр и графических проектов.

Воспитательные:

- воспитывать культуру безопасной работы за ПК, цифровую гигиену, уважение к авторскому праву и умение работать в команде (парное программирование).

Адресат программы. Программа предназначена для обучающихся в возрасте 12–17 лет, не требующая предварительной подготовки. Приём осуществляется в свободном порядке по заявлению родителей. Программа реализуется на стартовом уровне и доступна для всех желающих. Наполняемость группы — 15 - 20 человек. Общая трудоёмкость программы составляет 144 часа в год (36 учебных недель).

Характеристика контингента обучающихся.

Возрастная категория (12-14 лет).

Обучающиеся этого возраста переходят от образного к абстрактному и пространственному мышлению, учатся анализировать алгоритмы, запоминать многоэтапные инструкции и работать с цифровыми интерфейсами. Для них

важны чёткие задачи, наглядные примеры, визуальный результат и понимание пользы каждого этапа.

Программа развивает базовые навыки программированию на “python”. Особое внимание уделяется цифровой гигиене, организации файлов, самостоятельности в работе и умению спокойно справляться с техническими ошибками.

Как выстраивать деятельность:

- строить обучение на проектах: каждая тема — это создание своего продукта на основе полученных знаний. Так виден результат, есть мотивация и понятна связь с различными знаниями и алгоритмами при решение конкретных этапов поставленных задач.;
- чередовать технические задания и творческие проекты, давая всё больше свободы для собственных идей. Это помогает учитывать интересы и темп каждого ребёнка;
- создавать дружелюбную цифровую среду, где дети вместе работают, обсуждают ошибки и видят свой прогресс. Это помогает не бояться ошибок и верить в свои силы.

Возрастная категория 14-17 лет характеризуется возрастной категории которая находятся на этапе активного формирования абстрактно-логического, системного и проектного мышления. У них развивается способность к самостоятельному целеполаганию, критическому анализу, многоэтапному планированию и работе с профессиональным программным обеспечением. Для успешного обучения им важны задачи с практической значимостью, возможность выбора темы и формата проекта, обратная связь по результатам работы и чёткая связь обучения с будущими образовательными и профессиональными траекториями.

Как выстраивать деятельность:

- обучение строится на проектах при решении задач на основе пройденного материала. Это помогает видеть практическую пользу и поддерживает интерес;
- предлагать на выбор разные реализации при решении задач пройденного материала и предыдущих этапов обучения, создавать профессиональную среду: командные проекты, взаимная оценка работ, встречи со специалистами, участие в конкурсах и выставках. Это развивает коммуникацию, стрессоустойчивость и готовит к реальной работе.

Актуальность программы отвечает на запрос общества на раннюю профориентацию в высокотехнологичных сферах, формируя у обучающихся востребованные на рынке труда компетенции.

Педагогическая целесообразность заключается в построении обучения на основе современных образовательных принципов: доступности (переход от простого к сложному), деятельностного подхода (мгновенное закрепление знаний на практике), системности и проектной ориентированности.

Применение кейс-метода и метода проектов обеспечивает высокую учебную мотивацию, позволяет каждому обучающемуся двигаться в индивидуальном темпе и способствует формированию портфолио из реальных, практически значимых работ. Такой подход не только способствует глубокому усвоению материала, но и развивает самостоятельность, ответственность и навыки командной работы.

Основные педагогические принципы реализации программы:

- *доступность* (материал подаётся от простых геометрических примитивов и базовых операций к сложным сборкам и параметрическому проектированию, что позволяет любому подростку, независимо от начального уровня, уверенно освоить базовые навыки при программировании на “python”);
- *деятельность* (всё, что обучающиеся узнают на занятиях, они сразу же закрепляют на практике — программные продукты на поставленные задачи, работая как самостоятельно, так и в группах);
- *системность и систематичность* (программа выстроена логично и последовательно, что помогает подросткам шаг за шагом осваивать алгоритмы программирования, накапливать технический опыт и плавно переходить от учебных задач к самостоятельным проектам);
- *креативность* (каждый изученный модуль завершается созданием авторской программы, где обучающийся может проявить полученные знания при создании программного продукта. Творческий поиск и желание пробовать новое поддерживаются на всех этапах обучения).

Формы обучения. Программа реализуется в очном формате: педагог напрямую работает с детьми в специально оборудованном кабинете. При необходимости для обучения могут использоваться электронные и дистанционные технологии. Основной формой занятий является учебное занятие.

Возможно сетевое взаимодействие со школами Бахчисарайского района Республики Крым: партнёрские организации предоставляют помещение и оборудование, педагог программы проводит занятия. Все формы реализации соответствуют требованиям СанПиН и обеспечивают высокое качество образования.

Особенности организации образовательного процесса. Организация образовательного процесса происходит в соответствии с учебным планом. Группы формируются как одновозрастные, так и разновозрастные (в зависимости от начальных знаний и уровня подготовки). Состав группы постоянный. Программа предусматривает 2 вида занятий: теоретические занятия и практические занятия. В обучении применяется групповая форма с индивидуальным подходом, включающая обучение в малых группах.

Набор в группы проводится посредством подачи заявки в АИС «Навигатор ДО РК» с последующим предоставлением заявления родителем (законным представителем) и согласия на обработку персональных данных в письменном виде.

Режим занятий разработан с учётом возрастных особенностей обучающихся. Реализация программы планируется:

— Продолжительность одного занятия:2 академических часа;

— Периодичность занятий: 2 раза в неделю;

— Количество занятий: 4 занятия по 45 мин.;

— Общая трудоёмкость программы:144 часа в год;

— Перерыв между занятиями: 10-15 минут, с обязательным проветриванием помещения;

— Наполняемость группы: 15-20 человек, в соответствии с региональными нормативами для художественных объединений.

Уровень программы: стартовый, что предполагает минимальный уровень сложности освоения материала, доступности для данной возрастной группы обучающихся.

Объем и сроки освоения программы. Общее количество учебных часов, необходимых для освоения программы, составляет 144 часа. Срок реализации программы – 1 год. Продолжительность обучения по программе - 36 недель, период обучения - с сентября по май.

Формы аттестации/контроля. Система отслеживания и оценивания результатов обучения детей проходит через их участие в:

- беседе;
- тестировании;
- самостоятельных работах;
- комплексных работах;
- участия в конкурсных программах;
- педагогическом наблюдении.

Входной контроль – проводится в сентябре. Его цель — определить стартовый уровень логического мышления, базовые навыки работы с ПК, выявить индивидуальные особенности и мотивацию к обучению. Диагностика осуществляется с помощью тестирования на алгоритмическую логику, беседы и наблюдения педагога.

Текущий контроль – регулярная проверка знаний и умений после изучения каждой темы. Его задача — оперативное выявление затруднений в понимании синтаксиса или логики алгоритмов и оказание адресной помощи. Проводится в форме наблюдения, выполнения практических заданий, мини-тестов и взаимопроверки кода.

Промежуточный контроль (промежуточная аттестация) – проводится в декабре с целью выявления уровня освоения теоретических знаний и практических умений за I полугодие. Осуществляется в форме защиты мини-проекта (например, «Моя первая программа»), рефлексивной беседы и заполнения цифрового портфолио.

Итоговый контроль (итоговая аттестация) – проводится в мае в конце обучения с целью определения уровня сформированности компетенций. Осуществляется в форме публичной защиты итогового программного проекта и итогового тестирования.

Формы отслеживания и фиксации образовательных результатов: опрос, тестирование, практические программные работы, участие в муниципальных и всероссийских олимпиадах/конкурсах, дипломы, цифровое портфолио, диагностическая карта группы.

Формы предъявления и демонстрации образовательных результатов: готовые программные продукты, презентация проектов.

РАЗДЕЛ № 2

УЧЕБНЫЙ ПЛАН

№ п/п	Название темы	Количество часов			Форма аттестации (контроля)
		Всего часов	Теория	Практика	
1.	**Раздел 1. Техника безопасности. Общая характеристика языка Python – 4 часа**	**6**	**5**	**1**	**Входная диагностика, инструктаж**
1.1.	Тема 1. Основные этапы решения задач на компьютере. Алгоритм, его свойства. Инструктаж по ТБ»	3	2	1	Входная диагностика (беседа, диагностическое задание), инструктаж
1.2.	Тема 2. Организация рабочего места. Файловая дисциплина. Знакомство с интегрированной средой IDLE. Основные файлы среды, назначение основных горячих клавиш, структура программы.	3	2	1	Наблюдение, мини-опрос
2.	**Раздел 2. «Базовые операции в Python».**	**16**	**7**	**9**	**Текущее наблюдение, практические задания**
2.1.	Тема 3. Запуск программы. Создание простейших программ. Функции ввода	4	2	2	Практическое упражнение
2.2.	Тип переменной, имя переменной, объявление переменных, оператор присваивания.	4	2	2	Практическое упражнение
2.3.	Целочисленные типы данных. Деление.	4	2	2	Практическое задание, текущее наблюдение
2.4.	Разбор задач школьного этапа всероссийской олимпиады школьников по информатике по теме: «Целочисленная арифметика»	4	1	3	Практическое задание, текущее наблюдение

3.	**Раздел 3. «Условный оператор IF»**	**16**	**6**	**10**	**Тестирование, практикум**
3.1.	Вещественные типы данных в Python. Условный оператор if (полная и неполная формы).	4	2	2	Практическое упражнение
3.2.	Условный оператор if (полная и неполная формы). Вложенность условных операторов.	2	1	1	Практическое упражнение
3.3.	Разбор задач школьного этапа всероссийской олимпиады школьников по информатике по теме: «Условный оператор»	4	2	2	Практикум, карта наблюдений
3.4.	Решение задач на использование логических операций. Тестирование по теме: «Простейшие программы. Ветвления»	2	0	2	Практическое упражнение
3.5.	Оператор выбора if-elif-else.	2	1	1	Практическое задание, текущее наблюдение
3.6.	Решение задач на оператор выбора if-elif-else.	2	0	2	Практикум, взаимопроверка
4.	**Раздел 4. «Операторы циклов for, while, dowhile»**	**22**	**10**	**12**	**Практическое задание, взаимопроверка**
4.1..	Цикл, циклический алгоритм. Синтаксис операторов цикла. Отличия между циклами for, while, dowhile.	4	2	2	Практическое задание, текущее наблюдение
4.2.	Циклы с переменной	4	2	2	Практическое задание, текущее наблюдение
4.3.	Циклы с постусловием.	4	2	2	Практическое задание, текущее наблюдение
4.4.	Вложенные циклы.	4	2	2	Практическое задание, текущее наблюдение

4.5.	Разбор задач школьного этапа всероссийской олимпиады школьников по информатике по теме: «Циклические алгоритмы»»	2	2	0	Творческое задание
4.6.	Решение задач. Тестирование по теме: «Циклические алгоритмы»	4	0	4	Тестирование, практическая работа
5.	**Раздел 5. «Подведение итогов за I полугодие»**	**6**	**4**	**2**	Защита мини-проекта, рефлексия
5.1.	Анализ прогресса. Рефлексия: "Чему я научился?"»	2	2	0	Защита мини-проекта, групповая дискуссия, оценка по критериям
5.2.	Соревнование "Моя первая программа"	2	0	2	Конкурсная работа, оценивание жюри/педагога
5.3.	Планирование работы во II полугодии. Выбор темы итогового проекта»	2	2	0	Беседа, утверждение темы проекта
6.	**Раздел 6. «Символьные строки»**	**26**	**14**	**12**	**Практикум, взаимопроверка**
6.1.	Что такое символьная строка	4	3	1	Практическое упражнение, текущее наблюдение
6.2.	Сравнение строк. Сцепление строк	4	2	2	Практическое упражнение, текущее наблюдение
6.3.	Обращение к символам. Перебор всех символов	2	1	1	Практическое упражнение, текущее наблюдение
6.4.	Подстрока. Удаление и вставка	2	1	1	Практическое упражнение, текущее наблюдение
6.5.	Поиск в символьных строках	2	1	1	Практическое упражнение, текущее наблюдение
6.6.	Замена. Преобразования «строка — число»	4	2	2	Практическая работа, самопроверка
6.7.	Символьные строки в функциях	2	1	1	Практикум, код-ревью

6.8.	Срезы	4	1	3	Практикум тестирование навыков
6.9.	Разбор задач муниципального этапа всероссийской олимпиады школьников по информатике по теме: «Символьные строки»	2	2	0	Решение олимпиадных задач, анализ ошибок
7.	**Раздел 7. «Массивы»**	**38**	**18**	**20**	**Мини-тест, практическое задание**
7.1.	Массивы. Понятие массива. Обращение к элементу массива.	2	1	1	Мини-тест, работа с массивами
7.2.	Ввод и вывод элементов массива	4	2	2	Практикум, текущее наблюдение
7.3.	Заполнение массива случайными числами.	2	1	1	Практическое задание, проверка кода
7.4.	Простейшие алгоритмы обработки одномерных массивов. Сумма и количество	4	1	3	Практическая работа, расчет результатов
7.5.	Простейшие алгоритмы обработки одномерных массивов. Поиск min (max) элемента.	4	2	2	Практическая работа, тестирование алгоритма
7.6.	Простейшие алгоритмы обработки одномерных массивов. Выписывание элементов в новый массив.	4	2	2	Практическая работа, сравнение результатов
7.7.	Решение задач по теме: «Алгоритмы обработки массивов»	6	1	5	Практическая работа, сравнение результатов
7.8	Разбор задач муниципального этапа всероссийской олимпиады школьников по информатике по теме: «Массивы»	4	4	0	Анализ решений, обсуждение оптимальности
7.9.	Матрицы. Размещение матрицы в памяти компьютера. Заполнение матрицы	4	2	2	Практическое задание, визуализация данных

7.10.	Матрицы. Обработка матриц	4	2	2	Практическая работа, итоговый тест по разделу
8.	**Раздел 8. Словари**	**10**	**4**	**6**	**Практикум**
8.1..	Словари	6	4	2	Практическое задание, создание базы данных
8.2.	Перебор элементов словаря	4	0	4	Практическая работа, поиск данных
9.	**Раздел 9. Программирование на Python**	**2**	**1**	**1**	**Итоговое тестирование, защита элемента ООП**
9.1.	Основы Объектно-ориентированное программирование Итоговое тестирование	2	1	1	Тестирование, демонстрация класса/объекта
10.	**Раздел 10. «Подведение итогов за II полугодие»**	**2**	**0**	**2**	**Защита итогового проекта**
10.1	Тема1. «Защита итоговых проектов»	2	0	2	Публичная защита, презентация, оценка по критериям
	ИТОГО	**144**	**69**	**75**	-

РАЗДЕЛ № 3

КАЛЕНДАРНЫЙ УЧЕБНЫЙ ГРАФИК

Годовой календарный учебный график Программы «Программирование на Python» составлен с учетом годового календарного графика Муниципального бюджетного образовательного учреждения дополнительного образования «Центр дополнительного образования детей» Бахчисарайского района Республики Крым и учитывает в полном объеме возрастные, психофизические особенности обучающихся, отвечает требованиям охраны жизни и здоровья и нормам СанПиНа.

Учебные занятия проводятся с понедельника по субботу согласно расписанию, утвержденному директором Муниципального бюджетного образовательного учреждения дополнительного образования «Центр дополнительного образования детей» Бахчисарайского района Республики Крым, включая каникулярное время.

Год обучения	**Дата начала обучения**	**Дата окончания обучения**	**Всего учебных недель**	**Всего учебных дней**	**Всего учебных часов**	**Всего занятий в неделю**
1 год	1 сентября	31 мая	36	72	144	2 по 2 часа

Каникулы
01 июня – 31 августа

РАЗДЕЛ № 4

СОДЕРЖАНИЕ ПРОГРАММЫ

Раздел 1. Техника безопасности. Общая характеристика языка Python

Цель раздела: ориентация обучающихся в сфере цифрового проектирования, формирование культуры безопасной работы и организация цифрового рабочего пространства.

Задачи:

1. Сформировать представление о сферах применения языка Python.
2. Освоить базовые правила техники безопасности, цифровой гигиены и эргономики при работе за ПК.
3. Провести входную диагностику для выявления стартового уровня навыков и индивидуальных интересов.

Тема 1. «Основные этапы решения задач на компьютере. Алгоритм, его свойстваИнструктаж по ТБ»

Теория: Беседа о направлениях и перспективах использования языка Python при решении различных по структуре и направленности задач в различных сферах деятельности. Инструктаж по охране труда в компьютерном классе, правила поведения, основы цифровой безопасности.

Практика: Организация рабочего места, настройка программного обеспечения,

Тема 2. «Организация рабочего места. Файловая дисциплина. Знакомство с интегрированной средой IDLE. Основные файлы среды, назначение основных горячих клавиш, структура программы.»

Теория: Способы установки программного обеспечения на ПК , изучить Основные файлы среды, назначение основных горячих клавиш, структура программы.

Практика: Установка и первичная настройка Python, знакомство с интегрированной средой IDLE, проба использования горячих клавиш.

Формы контроля: Входная диагностика (беседа, диагностическое задание), инструктаж. Наблюдение, мини-опрос.

Раздел 2. «Базовые операции в Python. Типы данных».

Цель раздела: освоение рабочей среды Python, формирование уверенности при работе с IDLE и SHE LL.

Задачи:

1. Научить уверенно ориентироваться в окнах, панелях и режимах программы.
2. Отработать навыки перемещения между окнами программы, Освоить работу в режимах IDLE и SHELL.

Тема 1. «Запуск программы. Создание простейших программ. Функции ввода»

Теория: Практика: Запуск программы. Использование IDLE и SHELL Создание простейших программ. Функции ввода

Тема 2. «Тип переменной, имя переменной, объявление переменных, оператор присваивания»

Теория: Тип переменной, имя переменной, объявление переменных, оператор присваивания.

Практика: Определение типов переменных

Тема 3. «Целочисленные типы данных. Деление»

Теория: Структура данных. Операции деления в Python

Практика: Ввод в программу различных типов данных

Тема 4. «Разбор задач школьного этапа всероссийской олимпиады школьников по информатике по теме: «Целочисленная арифметика».

Практика: Решение задач по теме: «Целочисленная арифметика» с использованием среды SHE LL.

Формы контроля: Практическое упражнение. Практическое задание, текущее наблюдение.

Раздел 3. «Условный оператор IF»

Цель программы: погружение в логику ветвлений и принятия решений в программе..

Задачи:

1. Научить использовать ветвления.
2. Освоить основы пользования оператора IF
3. Понять принцип работы ветвлений .

Тема 1. «Вещественные типы данных в Python. Условный оператор if (полная и неполная формы)»

Теория: Понятие полная и неполная формы,

Практика: усвоение навыков использования полной и неполнойя формы

Тема 2. «Инструменты моделирования: *and, or, not*»

Теория: Назначение инструментов, правила применения, контроль плотности сетки.

Практика: Создание текстового квеста с ветвлением сюжета.

Тема 3. «Разбор задач школьного этапа всероссийской олимпиады школьников по информатике по теме: «Условный оператор»

Теория: Решение задач школьного этапа всероссийской олимпиады школьников по информатике по теме: «Условный оператор»

Тема 4. «Решение задач на использование логических операций. Тестирование по теме: «Простейшие программы. Ветвления»

Практика: Решение задач школьного этапа всероссийской олимпиады школьников по информатике по теме: «Условный оператор»

Тема 5. «Оператор выбора if-elif-else»

Теория: Конструкция множественного выбора. Особенности реализации в Python использование оператора выбора if-elif-else.

Практика: Создание программы-калькулятора или меню выбора с использованием множественного ветвления.

Тема 6. «Решение задач на оператор выбора if-elif-else»

Теория: Анализ типовых ошибок при множественном ветвлении.

Практика: Задачи на использование оператора выбора if-elif-else.

Формы контроля: Практическое упражнение. Практикум, карта наблюдений. Текущее наблюдение, взаимопроверка.

Раздел 4. «Операторы циклов for, while, dowhile»

Цель раздела: изучение принципов циклических алгоритмов и автоматизации повторяющихся действий.

Задачи:

1. Понять принципы работы циклов, их особенности и способы реализации.
2. Освоить синтаксис циклов;
3. Научиться избегать бесконечных циклов и использовать операторы.

Тема 1. «Цикл, циклический алгоритм. Синтаксис операторов цикла.

Теория: Синтаксис операторов цикла. Отличия между циклами for, while, dowhile..

Практика: решение задачи с использованием различных операторов цикла.

Тема 2. «Циклы с переменной».

Теория: Использование переменных в циклах.

Практика: Практическое задание «Решение одной и той же задачи с использованием переменной и разных типов циклов

Тема 3. «Циклы с постусловием»

Теория: Циклы с постусловием. Работа в программах.

Практика: Практическая работа «Построение программы с использованием цикла с постусловием

Тема 4. «Вложенные циклы»

Теория: Принцип работы циклов внутри циклов, управление внешним и внутренним счетчиками.

Практика: Вложенные циклы. Работа в программах

Тема 5 «Разбор задач школьного этапа всероссийской олимпиады школьников по информатике по теме: «Циклические алгоритмы»»

Теория: Анализ условий задач, требующих перебора вариантов или итеративных вычислений.

Практика: Написание оптимальных решений олимпиадных задач.

Тема 6 «Решение задач. по теме: «Циклические алгоритмы»

Теория: Повторение пройденного материала.

Практика: Самостоятельное выполнение тестовых заданий и практических задач на циклы.

Форма контроля: Практическое задание, текущее наблюдение. Тестирование, практическая работа.

Раздел 5. «Подведение итогов за I полугодие»

Цель раздела: систематизация знаний, демонстрация промежуточных результатов, планирование дальнейшего развития.

Задачи:

1. Провести анализ динамики освоения навыков.
2. Организовать соревнование алгоритмов для демонстрации прогресса.
3. Сформировать индивидуальные цели на второе полугодие.

Тема 1. «Анализ прогресса. Рефлексия: "Чему я научился?"»

Теория: Критерии оценки алгоритмов, принципы самоанализа, работа над ошибками.

Практика: Индивидуальная беседа с педагогом.

Тема 2. «Соревнование "Моя первая программа"»

Теория: Постановка задачи, критерии оценки.

Практика: создание программы, групповая дискуссия, оценка по критериям.

Тема 3. «Планирование работы во II полугодии. Выбор темы итогового проекта»

Теория: Этапы проектной деятельности, определение целевой аудитории, формулировка ТЗ.

Практика: Индивидуальная консультация, составление алгоритма, выбор направления проекта.

Формы контроля: Защита мини-проекта, групповая дискуссия, оценка по критериям. Конкурсная работа, оценивание жюри/педагога. Беседа, утверждение темы проекта

Раздел 6. «Символьные строки»

Цель раздела: переход к освоению решению задач использующих символьную информацию.

Задачи:

1. Понять разницу между цифровой и символьной информацией
2. Отличие ввода вывода, обработки между цифровой и символьной информацией.
3. Научить переходу между цифровыми и символьными переменными.

Тема 1. «Что такое символьная строка»

Теория: Что такое символьная строка. Типы переменных., примеры.

Практика: Практическое упражнение: манипуляции со строчными переменными с использованием SHELL .

Тема 2. «Сравнение строк. Сцепление строк»

Теория: Сравнение строк. Сцепление строк.

Практика: Практическое упражнение: манипуляции со строчными переменными.

Тема 3. «Обращение к символам. Перебор всех символов»

Теория: структура символьной переменной

Практика: Поиск внутри строки с использованием SHELL.

Тема 4. «Подстрока. Удаление и вставка»

Теория: способы удаления и вставки в символьных строках.

Практика: Практическая работа «Подстрока. Удаление и вставка в указанной строке.

Тема 5. «Поиск в символьных строках»

Теория: способы поиска в символьных строках.

Практика: Практическая работа «Поиск в символьных строках».

Тема 6. «Замена. Преобразования «строка — число»»

Теория: способы поиска и замены . Преобразования «строка — число» и обратное действие в символьных строках.

Практика: Практическая работа «Преобразования «строка — число» и обратное действие в символьных строках».

Тема 7. «Символьные строки в функциях»

Теория: понятие функции.

Практика: Практическая работа «функции с использованием строчных переменных».

Тема 8. «Срезы»

Теория: как работают срезы .

Практика: Практическая работа «Использование срезов для поиска внутри строки».

Тема 9. «Разбор задач муниципального этапа всероссийской олимпиады школьников по информатике по теме: «Символьные строки»»

Теория: Разбор задач муниципального этапа всероссийской олимпиады школьников по информатике по теме: «Символьные строки».

Практика: Решение задач муниципального этапа всероссийской олимпиады школьников по информатике по теме: «Символьные строки».

Формы контроля: Практическое упражнение, текущее наблюдение. Практикум. Тестирование навыков. Решение олимпиадных задач, анализ ошибок.

Раздел 7. «Массивы»

Цель раздела: освоение понятие м**ассивы в Python**.

Задачи:

1. Понять принципы работы массивов.
2. Изучить правила работы с массивами.
3. Навыки построения алгоритмов с массивами .

Тема 1. «Массивы. Понятие массива. Обращение к элементу массива)»

Теория: Массивы. Понятие массива. Обращение к элементу массива.

Практика: Мини-тест, работа с массивами.

Тема 2. «Ввод и вывод элементов массива»

Теория: Ввод и вывод элементов массива.

Практика: Практикум. Заполнения данных в массиве

Тема 7.3. «Заполнение массива случайными числами»

Теория: Генератор случайных чисел Заполнение массива случайными числами.

Практика: решение простых задач с использованием генератора случайных чисел, заполнение массива.

Тема 4. «Простейшие алгоритмы обработки одномерных массивов. Сумма и количество»

Теория: Изучить работу операторов суммы и количества в массивах

Практика: Подсчет суммы и количества в заданном массиве.

Тема 5. «Простейшие алгоритмы обработки одномерных массивов. Поиск min (max) элемента»

Теория: Изучить работу операторов Поиска min (max) элемента в массивах

Практика: Поиск min (max) элемента в заданном массиве.

Тема 6. «Простейшие алгоритмы обработки одномерных массивов. Выписывание элементов в новый массив»

Теория: Изучить работу формирования нового массива из элементов других массивов.

Практика: Заполнить новый массив используя заданные данные.

Тема 7. «Решение задач по теме: «Алгоритмы обработки массивов»»

Теория: Повторить основные операторы для работы с массивами.

Практика: Решить тестовую задачу используя заданные данные.

Тема 8. «Разбор задач муниципального этапа всероссийской олимпиады школьников по информатике по теме: «Массивы»»

Теория: Разобрать основные шаги решения олимпиадных задач по теме.

Практика: Анализ и разбор решения одной из задач по теме .

Тема 9. «Матрицы. Размещение матрицы в памяти компьютера. Заполнение матрицы»

Теория: матрицы в python особенности.

Практика: Работа с готовой матрицей.

Тема 10. «Матрицы. Обработка матриц»

Теория: матрицы в python Обработка матриц.

Практика: Работа с готовой матрицей.

Формы контроля: Мини-тест, работа с массивами. Практикум, текущее наблюдение. Практическое задание, проверка кода. Расчет результатов. Тестирование алгоритма. Анализ решений, обсуждение оптимальности. Итоговый тест по разделу.

Раздел 8. «Словари»

Цель раздела: изучение базовых понятий и принципов размещения данных в словарях.

Задачи:

1. Освоить работу с словарями.

2. Научить создавать , заполнять ,редактировать ,удалять элементы словарей.
3. Сформировать навык работы с словарями.

Тема 1. «Словари»

Теория: Словари. Обращение к элементу Словаря.

Практика: Создание простой базы данных

Тема 2. «Перебор элементов словаря»

Теория: Ввод и вывод элементов словаря.

Практика: Практикум. Заполнения данных в словаре

Формы контроля: Практическое задание, создание базы данных. Практическая работа, поиск данных.

Раздел 9. Программирование на Python

Цель раздела: знакомство с парадигмой объектно-ориентированного программирования (ООП) и подготовка к итоговой аттестации.

Задачи:

1. Познакомить учащихся с основами ООП: классами, объектами, атрибутами и методами.
2. Показать преимущества ООП для структурирования крупных проектов.
3. Провести итоговое тестирование теоретических знаний.

Тема 1. «Объектно-ориентированное программирование»

Теория: Основы проектной деятельности: Учащиеся изучают этапы разработки проектов, начиная с постановки цели и заканчивая презентацией готового продукта. Рассматриваются методы планирования, распределения задач, документации.

Практика: Разработка ООП-программ: Учащиеся создают программы, использующие ООП, например, симуляции объектов реального мира, например: авто или приемная поликлиники.

Форма контроля: Тестирование, демонстрация класса/объекта.

Раздел 10. «Подведение итогов за II полугодие»

Цель раздела: итоговая оценка результатов освоения программы, создание ситуации успеха через публичную презентацию проектов.

Задачи:

1. Систематизировать знания, полученные за год.
2. Организовать публичную защиту итоговых проектов.
3. Заполнить цифровое портфолио, провести рефлексию.

Тема 1. «Защита итоговых проектов»

Теория: Правила публичного выступления, структура презентации проекта (проблема, решение, демонстрация кода, выводы).

Практика: Публичная защита проектов, презентация программного продукта, ответы на вопросы жюри/педагога, оценка по критериям, заполнение портфолио.

Формы контроля: Публичная защита, презентация, оценка по критериям.

РАЗДЕЛ № 5

ВОСПИТАНИЕ

Цель воспитательной работы направлена на формирование у обучающихся целостной, социально ответственной и технически грамотной личности, способной к созидательному применению цифровых технологий, безопасной проектной деятельности, командному взаимодействию и осознанному профессиональному самоопределению в сфере IT и программирования.

Задачи:

1. Духовно-нравственные:
   - Воспитывать этику цифрового творчества, уважение к авторскому праву и интеллектуальной собственности;
   - Формировать ответственность за результат, честность при использовании чужого кода (недопустимость плагиата) и открытость к конструктивной критике (code review);
   - Развивать эмпатию через создание социально значимых цифровых продуктов, облегчающих жизнь людям.
2. Эстетические:
   - Развивать художественно-образное восприятие цифрового пространства и пользовательских интерфейсов (UI/UX);
   - Формировать чувство гармонии, логики и «чистоты» в написании программного кода;
   - Воспитывать вкус к качественному цифровому дизайну и удобству использования программных продуктов.
3. Гражданско-патриотические:
   - Формировать уважение к отечественным инженерам, конструкторам и IT-специалистам;
   - Воспитывать чувство сопричастности к технологическому суверенитету и развитию страны и региона (Республики Крым);
   - Приобщать к государственным и региональным символам через создание тематических цифровых проектов и приложений.
4. Социально-коммуникативные:

- Развивать навыки командной проектной работы (парное программирование), распределения ролей, ведения диалога и публичной презентации;
- Формировать умение аргументированно объяснять свой алгоритм и защищать проектные решения;
- Воспитывать культуру поведения в цифровом пространстве (сетевой этикет) и на очных конференциях.

5. Личностные:

- Развивать уверенность в своих логических и цифровых способностях, умение спокойно справляться с ошибками (багами) как с частью учебного процесса;
- Формировать навыки саморегуляции, тайм-менеджмента и рефлексии;
- Воспитывать ответственность за процесс и результат своей интеллектуальной деятельности.

Формы и методы воспитательной работы:

- **Словесные** (беседа-диалог по итогам проектов, разбор этических IT-кейсов, рефлексивные круги, встречи с представителями IT-профессий);
- **Наглядные** (демонстрация прототипов и готовых программ, видеолекции о российских технологических достижениях);
- **Практические** (проектная лаборатория, парное программирование, создание волонтёрских цифровых продуктов для соцучреждений);
- **Игровые** (IT-квесты, ролевая игра «Заказчик – Разработчик», симуляции проектного офиса);
- **Рефлексивные** (доброжелательный code review, обсуждение работ, анализ ошибок без оценивания личности);
- **Социальные** (участие в олимпиадах, конкурсах и фестивалях технического творчества).

Условие реализации воспитательного компонента программы обеспечивается комплексом взаимосвязанных условий это:

— Психолого-педагогические условия:
  - Комфортная эмоциональная среда;
  - Безоценочная поддержка на старте;
  - Индивидуальный учёт темпа и интересов;
  - Безопасное пространство для экспериментов и ошибок.

— Организационные условия:
  - Чёткая проектная структура занятий;
  - Чередование индивидуальной и групповой работы;
  - Регулярная обратная связь;
  - Взаимодействие с родителями.

— Материально-технические условия:
  - Оборудованный компьютерный класс;
  - Оборудование;
  - Материалы;
  - Информационные ресурсы.

— Кадровые условия:
  - Квалификация педагога в сфере IT и педагогики;
  - Профессиональное развитие и самообразование;
  - Личностные качества педагога (наставничество, эмпатия).

Анализ результатов воспитательной работы по программе «Программирование на Python» осуществляется в **неотметочной, качественной форме** с акцентом на фиксацию индивидуальной динамики каждого обучающегося и создание ситуации успеха.

Система анализа воспитательной работы заключается в:

1. Текущем контроле:
   - Педагогическое наблюдение за взаимодействием в парах, анализ этики общения при code review, оценка самостоятельности при поиске ошибок;
   - Профессиональное развитие и самообразование;
   - Личностные качества педагога (наставничество, эмпатия).
2. Промежуточный контроль:
   - Анализ мини-проектов и качества кода;
   - Анкетирование родителей;
   - Защита мини-проектов с обсуждением.
3. Итоговый контроль:
   - Итоговая презентация (защита проекта);
   - Демонстрация цифрового портфолио;
   - Итоговое анкетирование;
   - Сводная аналитическая таблица.

Инструментами анализа являются: карта педагогических наблюдений, визуальная шкала самооценки для обучающихся, адаптированная анкета для родителей, аналитический лист по модулю и цифровое портфолио.

Календарный план воспитательной работы

№ п/п	Содержание работы	Дата проведения	Место проведения	Примечание
ВОСПИТАТЕЛЬНЫЕ И ТЕМАТИЧЕСКИЕ МЕРОПРИЯТИЯ, АКЦИИ				
1.	День открытых дверей «Цифровая лаборатория: мы рады вам»	сентябрь	МБОУ ДО «ЦДОД»	
2.	Викторина ко Дню Государственного герба и Флага Республики Крым	сентябрь	МБОУ ДО «ЦДОД»	
3.	Акция «Цифровая грамотность для старшего поколения» (обучение основам ПК)	октябрь	МБОУ ДО «ЦДОД»	
4.	Тематическое мероприятие «Папа может» (совместный IT-проект с родителями)	октябрь	МБОУ ДО «ЦДОД»	
5	Участие в акции ко Дню народного единства «Единство народа – несокрушимая сила»	ноябрь	МБОУ ДО «ЦДОД»	
6.	Выставка цифровых проектов и открыток «Самой милой и любимой…» (ко Дню матери)	ноябрь	МБОУ ДО «ЦДОД»	
7.	Викторина ко Дню Конституции Российской	декабрь	МБОУ ДО «ЦДОД»	

	Федерации «Закон и цифровые технологии»			
8.	Квест «Инженерный и IT-прорыв: российские изобретения»	январь	МБОУ ДО «ЦДОД»	
9.	Тематическое мероприятие ко Дню Республики Крым «Любимый сердцу Крым в цифровом пространстве»	январь	МБОУ ДО «ЦДОД»	
10.	Выставка работ ко Дню защитника Отечества «На страже Родины моей» (симуляции, игры)	февраль	МБОУ ДО «ЦДОД»	
11.	Акция-поздравление ко Международному женскому дню «Тебе единственной» (дизайн и код)	март	МБОУ ДО «ЦДОД»	
12.	День воссоединения Крыма с Россией (создание интерактивного цифрового проекта)	март	МБОУ ДО «ЦДОД»	
13.	Участие в мероприятиях ко Дню Победы «Победный Май» (цифровой архив, интерактивные открытки)	май	МБОУ ДО «ЦДОД»	
14	День открытых дверей «Цифровая лаборатория: мы рады вам»	сентябрь		
15	Викторина ко Дню Государственного герба и Флага Республики Крым	сентябрь		
МУНИЦИПАЛЬНЫЕ ЭТАПЫ КОНКУРСНЫХ ПРОГРАММ				
1.	Конкурс детского творчества «Дорога глазами детей» (программирование симуляции или дизайна среды)	сентябрь-октябрь	МБОУ ДО «ЦДОД»	
2.	Конкурс «Космические фантазии» (создание текстовой или графической IT-модели)	октябрь-ноябрь	МБОУ ДО «ЦДОД»	
3.	Конкурс «Базовые национальные ценности» (цифровой storytelling)	ноябрь-декабрь	МБОУ ДО «ЦДОД»	

4.	Республиканский конкурс-фестиваль «Крым в сердце моем»	февраль-март	МБОУ ДО «ЦДОД»	
5.	Конкурс «Ради жизни на земле!» (эко-IT проект, приложение или игра)	май	МБОУ ДО «ЦДОД»	
РАБОТА ПО ВЫЯВЛЕНИЮ И ПОДДЕРЖКЕ ОДАРЕННЫХ ДЕТЕЙ				
1.	Проведение открытых занятий и мастер-классов	в течение года		
2.	Индивидуальные консультации для подготовки к региональным/всероссийским техническим конкурсам	в течение года		
3.	Рекомендации и сопровождение участия в олимпиадах по технологии/информатике	в течение года		
РАБОТА С РОДИТЕЛЯМИ				
1.	Родительские собрания	в течение года		
2.	Анкетирование	в течение года		
3.	Консультации для родителей	в течение года		
4.	Работа в сообществах и группах в социальных сетях	в течение года		
ВОСПИТАТЕЛЬНЫЕ БЕСЕДЫ				
1.	Правила ТБ и эргономики при работе с ПК	сентябрь		
2.	Цифровая этика, кибербезопасность и авторское право	октябрь		
3.	Профилактика переутомления, цифрового выгорания и правила цифрового детокса	ноябрь		
4.	Экологичность IT: ответственное потребление и утилизация электронной техники	декабрь		
5.	Правила дорожно-транспортной безопасности	февраль		

	пешехода, велосипедиста (зимний период)			
6.	Правила пожарной безопасности в быту и учреждении образования. Электробезопасность	март		
7.	Действия в экстремальных ситуациях: во дворе, на улицах, дома, в общественных местах	апрель		
8.	Правила безопасности при общении с животными и в сети Интернет	май		

РАЗДЕЛ № 6

ПЛАНИРУЕМЫЕ РЕЗУЛЬТАТЫ

В результате освоения дополнительной общеобразовательной общеразвивающей программы технической направленности «Программирование на Python» обучающиеся достигнут следующих планируемых результатов:

Предметные:

Обучающиеся будут **знать:**

- названия и назначение основных типов данных в Python (целые числа, вещественные числа, строки, списки/массивы, словари);
- правила техники безопасности, цифровой гигиены и эргономики при работе за компьютером;
- синтаксис и логику работы базовых конструкций языка: операторов присваивания, условного оператора if (полная и неполная формы), операторов цикла (for, while);
- основные приемы работы со строками: конкатенация, срезы, поиск, замена и преобразование типов данных;
- принципы создания, заполнения и обработки одномерных и двумерных массивов (списков и матриц), включая поиск минимального/максимального элемента и суммы;
- базовые понятия объектно-ориентированного программирования (ООП): класс, объект, атрибуты и методы;
- назначение и способы подключения стандартных библиотек Python для расширения функциональности программ.

Обучающиеся будут **уметь:**

- организовывать свое цифровое рабочее место, соблюдать файловую дисциплину и использовать горячие клавиши в среде разработки (IDLE);
- писать, запускать и отлаживать простейшие программы, используя функции ввода и вывода данных;
- применять условные операторы и циклы для реализации ветвящихся и циклических алгоритмов;
- выполнять операции над строками и массивами: обращаться к элементам по индексу, использовать срезы, перебирать элементы в цикле;
- читать и анализировать чужой код, находить синтаксические и логические ошибки и исправлять их;
- оформлять код в соответствии с базовыми стандартами (PEP 8), добавлять комментарии для пояснения логики работы программы.

Обучающиеся **смогут:**

- самостоятельно реализовывать полный цикл создания небольшой программы: от анализа задачи и составления алгоритма до написания кода и тестирования;

- применять полученные навыки для решения задач школьного этапа всероссийской олимпиады школьников по информатике;
- адаптировать базовые алгоритмы (поиск, сортировка, фильтрация) под конкретные практические задачи;
- создавать и презентовать собственный итоговый программный проект, демонстрирующий освоенные компетенции.

Метапредметные:

Обучающиеся будут **знать:**

- алгоритм решения алгоритмических задач: анализ условия, выбор структур данных, составление блок-схемы или псевдокода, написание и отладка программы;
- критерии оценки качества программного кода (читаемость, эффективность, отсутствие избыточных операций);
- правила работы в паре и группе при совместной разработке проекта: распределение ролей, взаимопомощь;
- способы самопроверки программы и методы поиска информации в официальной документации и справочных ресурса.

Обучающиеся будут **уметь:**

- планировать последовательность действий при выполнении программного задания, разбивая сложную задачу на подзадачи;
- работать по инструкции, образцу и алгоритму, предложенному педагогом, а также самостоятельно искать решения при возникновении нестандартных ошибок;
- осуществлять самоконтроль: сравнивать результат работы программы с ожидаемым, находить и исправлять логические ошибки;
- аргументированно объяснять свой выбор структур данных и алгоритмов при защите проекта;
- взаимодействовать со сверстниками в процессе командной работы: договариваться, распределять задачи, оказывать техническую помощь.

Обучающиеся **смогут:**

- самостоятельно ставить учебно-исследовательскую или проектную задачу и выбирать оптимальные способы её программной реализации;
- переносить освоенные универсальные учебные действия (алгоритмическое мышление, планирование, контроль, оценка) в другие образовательные ситуации (математика, физика, технология);
- осуществлять рефлексивную оценку своей деятельности: анализировать успешность написанного кода, формулировать цели для дальнейшего изучения новых библиотек и технологий;
- использовать цифровые инструменты (облачные хранилища, системы контроля версий, электронное портфолио) для фиксации и презентации результатов своей работы.

Личностные:

Обучающиеся будут **знать:**

- ценность интеллектуального труда и уважение к авторскому праву, этике цифрового творчества;
- значение отечественных разработок в сфере IT и важность технологического суверенитета страны;
- важность дисциплины, усидчивости, внимательности к деталям и ответственности при написании программного кода.

Обучающиеся будут **уметь:**

- проявлять интерес и эмоциональный отклик на успешное решение сложных алгоритмических задач;
- выражать своё отношение к цифровым технологиям, осознавать их роль в современном мире;
- соблюдать правила безопасного поведения в цифровой среде и культуру труда в компьютерном классе;
- проявлять инициативу, творческую самостоятельность и настойчивость в поиске причин ошибок (дебаггинге) и достижении работающего результата;
- уважительно относиться к работам (коду) сверстников, давать доброжелательную и конструктивную обратную связь.

Обучающиеся **смогут:**

- испытывать чувство удовлетворения от создания собственного работающего программного продукта;
- быть уверенными в своих силах, не бояться экспериментировать с кодом и воспринимать программную ошибку не как неудачу, а как часть учебного процесса;
- осознавать прямую связь между личными усилиями, качеством написанного кода и достигнутым результатом;
- проявлять гражданско-патриотические чувства через участие в проектах, посвященных культуре, истории и развитию родного края (с использованием IT-инструментов);
- формировать позитивную самооценку через ситуацию успеха, публичную защиту проектов и признание достижений в профессиональном сообществе.

РАЗДЕЛ № 7

РЕСУРСНОЕ ОБЕСПЕЧЕНИЕ

Кадровое обеспечение.

Педагог, работающий по дополнительной общеобразовательной общеразвивающей программе технической направленности «Программирование на Python», имеет высшее техническое или педагогическое образование в сфере информационных технологий, обладает профессиональными знаниями, знает специфику дополнительного образования, имеет практические навыки в сфере организации интерактивной проектной деятельности детей и подростков. Также владеет современными навыками работы с компьютерной техникой и программным обеспечением (включая языки программирования и среды разработки), базовыми навыками работы со средствами телекоммуникаций (системами навигации в сети Интернет, навыками поиска информации, электронной почтой и облачными сервисами).

Материально-техническое обеспечение:

— Специализированный компьютерный класс площадью не менее 50 м² с естественным и искусственным освещением, системой приточно-вытяжной вентиляции, безопасностью.

— Оборудование:

- рабочие станции (ПК: процессор от Intel Core i5 / AMD Ryzen 5, ОЗУ 16 ГБ, видеокарта от 4 ГБ, SSD 512 ГБ);
- столы;
- стулья,
- мультимедийный проектор,
- стеллажи для хранения материалов и прототипов,
- огнетушитель, журнал инструктажей по ТБ,
- стеллажи для материалов,
- аптечка,
- огнетушитель.

— Программное обеспечение и расходные материалы (для личного пользования обучающихся приобретается родителями (законными представителями):
калибровочные щупы,

- салфетки и расходные материалы для ухода за оборудованием.

Методическое обеспечение.

Методы обучения:

— словесный (беседа, объяснение, инструктаж, дискуссия),

— наглядный (демонстрация интерфейса ПО, скринкасты, видеоуроки, образцы готовых моделей и прототипов),

— практический (выполнение упражнений, проектная и исследовательская работа),

— объяснительно-иллюстративный (поэтапный показ приёмов моделирования),

— репродуктивный (повторение образцов, копирование элементов),

— проблемный (анализ реальных инженерно-дизайнерских задач),

— метод проектов (создание цифровых и физических прототипов от эскиза до презентации).

Формы организации образовательного процесса:

— индивидуальная (персональная траектория, работа над ошибками),

— парная (взаимопроверка моделей, совместное решение задач),

— групповая (командные мини-проекты, мозговые штурмы),

— коллективная (хакатоны, выставки, защита итоговых работ).

Педагогические технологии:

— технология проектного обучения;

— технология дифференцированного и разноуровневого обучения;

— технология сотрудничества и взаимного обучения;

— ИКТ-технологии и цифровое портфолио;

— здоровьесберегающие технологии (динамические паузы, гимнастика для глаз и опорно-двигательного аппарата при работе за ПК, контроль времени непрерывной экранной нагрузки).

Алгоритм учебного занятия:

— Подготовка кабинета к проведению занятия: проветривание помещения, проверка работоспособности ПО и оборудования, расстановка рабочих мест, подготовка наглядных материалов и инструкций.

— Организационный момент: приветствие обучающихся, проверка готовности, настрой на продуктивную работу, актуализация опорных знаний.

— Введение в тему: объявление цели и задач занятия, мотивация (демонстрация примеров из индустрии, связь с реальными профессиями).

— Теоретическая часть: объяснение нового материала, пошаговая демонстрация приёмов, разбор типичных ошибок и способов их устранения.

— Физкультминутка: упражнения для снятия зрительного напряжения, разминка кистей рук и шейно-плечевого отдела.

— Практическая часть: самостоятельное выполнение заданий, моделирование, сохранение и резервное копирование кода, загрузка в облачное хранилище / репозиторий /рендеру, индивидуальные и групповые консультации педагога, промежуточный контроль.

— Завершение занятия: рефлексия (самооценка, обсуждение трудностей и достижений), сохранение и резервное копирование файлов, уборка рабочих мест, анонс следующей темы.

Информационно-телекоммуникационное обеспечение:

— Электронный журнал «Навигатор» для учёта посещаемости и тем занятий;

— Презентации к занятиям

— Видеоматериалы:;

— Облачное хранилище для обмена файлами, совместной работы и хранения цифрового портфолио обучающихся;

— Компьютер педагога с доступом к интернету,

— Мультимедийное оборудование,

— Wi-Fi покрытие в учебном помещении;

— Программное обеспечение: Яндекс.Браузер;

— Государственные информационные ресурсы:

— Официальный сайт Министерства просвещения Российской Федерации https://edu.gov.ru/

— Официальный сайт Министерства образования, науки и молодежи Республики Крым https://monm.rk.gov.ru/ru/index

— Официальный сайт МБОУ ДО «ЦДОД» https://cdod-bakhchisaray.krymschool.ru

— 2. Информационно-коммуникационные педагогические платформы:

— «Сферум» https://sferum.ru/?p=start

— Навигатор дополнительного образования Республики Крым https://p82.навигатор.дети/

— Образовательные порталы:

— Российское образование http://www.edu.ru

— Инфоурок https://infourok.ru/

— Официальные сайты разработчиков ПО:

РАЗДЕЛ № 8

СПИСОК ЛИТЕРАТУРЫ

1. Литература для педагога:

1.1. Основная литература:

- Поляков, К. Ю., Еремин, Е. А. Информатика. Углубленный уровень: учебник для 10–11 классов [Текст] / К. Ю. Поляков, Е. А. Еремин. — Москва : БИНОМ. Лаборатория знаний, 2023. — 384 с.
- Федоров, Д. Ю. Программирование на языке высокого уровня Python : учебное пособие [Текст] / Д. Ю. Федоров. — 3-е изд., перераб. и доп. — Москва : Юрайт, 2024. — 180 с.
- Лутц, М. Изучаем Python. Том 1 [Текст] / М. Лутц ; пер. с англ. — 5-е изд. — Санкт-Петербург : Символ-Плюс, 2023. — 832 с.

1.2. Дополнительная литература:

- Свигарт, Э. Python для детей. Программируем игры и приложения [Текст] / Э. Свигарт. — Санкт-Петербург : Питер, 2024. — 352 с.
- Шень, А. Программирование: теоремы и задачи [Текст] / А. Шень. — 3-е изд. — Москва : МЦНМО, 2023. — 296 с.
- Гутько, А. И. Алгоритмы и структуры данных на Python [Текст] / А. И. Гутько. — Москва : ДМК Пресс, 2024. — 256 с.

1.3. Электронные ресурсы для педагога:

- Официальная документация по языку программирования Python [Электронный ресурс]. — URL: https://docs.python.org/3/
- Российская электронная школа: образовательная платформа [Электронный ресурс]. — URL: https://resh.edu.ru
- Портал «Инфоурок»: методические разработки и курсы [Электронный ресурс]. — URL: https://infourok.ru/
- Министерство просвещения Российской Федерации: официальный сайт [Электронный ресурс]. — URL: https://edu.gov.ru

2. Литература для обучающихся:

2.1. Основная литература:

- Поляков, К. Ю., Еремин, Е. А. Информатика. Углубленный уровень. 10 класс [Текст] / К. Ю. Поляков, Е. А. Еремин. — Москва : БИНОМ. Лаборатория знаний, 2023. — 320 с.
- Свигарт, Э. Изучаем Python. Программируем игры, визуализируем данные, анализируем информацию [Текст] / Э. Свигарт. — Санкт-Петербург : Питер, 2024. — 416 с.

2.2. Дополнительная литература:

- Брайсон, Д. Изучаем Python. Введение в программирование [Текст] / Д. Брайсон. — Санкт-Петербург : Питер, 2023. — 304 с.
- Зед А. Шоу, Л. Учимся программировать на Python [Текст] / Л. Зед А. Шоу. — Москва : Эксмо, 2023. — 288 с.

- Марков, С. В. От Scratch к Python: программируем просто и понятно [Текст] / С. В. Марков. — Москва : Бином, 2023. — 144 с.

2.3. Электронные ресурсы для обучающихся:

- PythonTutor: интерактивная визуализация выполнения кода [Электронный ресурс]. — URL: http://pythontutor.ru/
- Stepik: интерактивные курсы «Поколение Python» [Электронный ресурс]. — URL: https://stepik.org/course/58852
- Metanit: руководство по изучению Python [Электронный ресурс]. — URL: https://metanit.com/python/
- Открытый курс по программированию на Open Book Project [Электронный ресурс]. — URL: http://openbookproject.net/

3. Литература для родителей:

3.1. Основная литература:

- Гиппенрейтер, Ю. Б. Общаться с подростком. Как? [Текст] / Ю. Б. Гиппенрейтер. — 16-е изд. — Москва : АСТ, 2023. — 256 с.
- Савенков, А. И. Одарённый ребёнок в массовой школе : книга для родителей [Текст] / А. И. Савенков. — Москва : Линка-Пресс, 2024. — 192 с.

3.2. Дополнительная литература:

- Венгер, Л. А. Воспитание и развитие ребёнка : книга для родителей и педагогов [Текст] / Л. А. Венгер, В. С. Мухина. — 3-е изд. — Москва : Юрайт, 2024. — 224 с.
- Дубровская, Н. В. Цифровая грамотность ребёнка : рекомендации для родителей [Текст] / Н. В. Дубровская. — 2-е изд. — Москва : Национальное образование, 2023. — 128 с.

3.3. Электронные ресурсы для родителей:

- Я — родитель: портал о воспитании детей [Электронный ресурс]. — URL: https://www.ya-roditel.ru
- Навигатор дополнительного образования Республики Крым [Электронный ресурс]. — URL: https://p82.навигатор.дети/

3. Наглядный материал и иллюстративные ресурсы:

4.1. Альбомы и справочные материалы:

- Синтаксис Python: справочная таблица (шпаргалка) [Изоиздание]. — Москва : Просвещение, 2023. — 1 табл. (60×90 см) : цв.
- Блок-схемы алгоритмов: наглядное пособие для школьников [Изоиздание]. — Москва : БИНОМ, 2023. — 8 плакатов (42×60 см) : цв. ил.

4.2. Дидактические и раздаточные материалы:

- Карточки «Типы данных и переменные в Python» [Текст].
- Карточки «Операторы ветвления и циклы» [Текст].
- Чек-лист «Проверка кода на типичные ошибки» [Текст].
- Рабочие листы «От алгоритма к программе» [Текст].

4.3. Электронные наглядные ресурсы:

- Видеоуроки «Основы программирования на Python» [Электронный ресурс] / образовательный канал «Инфоурок». — URL: https://infourok.ru/video
- Интерактивный визуализатор алгоритмов [Электронный ресурс] / Python Tutor. — URL: http://pythontutor.ru/

РАЗДЕЛ № 9

Приложение 1

Оценочные материалы

ВХОДНАЯ ДИАГНОСТИКА

Диагностическое задание «Мой первый алгоритм»

Назначение: Оценка стартовых представлений об алгоритмическом мышлении, базовой компьютерной грамотности и выявление мотивации к обучению.
Время выполнения: 20–25 минут.
Материалы: Лист с заданиями, компьютер (по возможности), ручка.
Условия: Спокойная обстановка, отсутствие жесткого ограничения времени, возможность задать уточняющий вопрос педагогу.
Критерии оценки диагностического задания

Параметр	Индикаторы	Оценка (баллы)
Алгоритмическое мышление	1. Понимает последовательность действий для решения простой задачи. 2. Может выделить условие и результат. 3. Предлагает логичный, пусть и простой, план действий.	3 – самостоятельно 2 – с небольшой подсказкой 1 – требуется помощь
Компьютерная грамотность	1. Знает основные устройства ПК и их назначение. 2. Имеет первичный опыт работы с текстовым редактором или простыми программами. 3. Понимает базовые правила цифровой гигиены.	3 – самостоятельно 2 – с небольшой подсказкой 1 – требуется помощь
Мотивация и интерес	1. Четко формулирует, что хочет создавать (игры, сайты, боты и т.д.). 2. Проявляет любопытство и готовность пробовать. 3. Адекватно оценивает свои силы.	3 – высокая мотивация 2 – средняя мотивация 1 – низкая мотивация

Лист анализа диагностического рисунка (Шаблон)
ФИО ребёнка: _____ Возраст: _____
Дата: _____

Критерии	Оценка (3/2/1)	Комментарий
Алгоритмическое мышление		
Компьютерная грамотность		
Мотивация и интерес		

Рефлексия ребёнка (со слов ребёнка):
«Что я хочу научиться создавать на Python:» _____
«Что кажется самым сложным в программировании:» _____
«Что кажется самым сложным в программировании:» _____
«Мой опыт работы с компьютером:» _____

Педагог: _____ Подпись: _____

Карта первичного наблюдения (Для педагога)

Программа «Программирование на Python»
Группа: _____ Дата: _____

ФИО ребёнка	Работа с клавиатурой / мышью	Следование инструкции	Реакция на ошибку	Темп работы	Эмоциональный фон	Комментарий
Иванов А.	Уверенно	Да	Спокойно ищет решение	Средний	Заинтересован	Быстро схватывает логику
Петрова И.	Не уверено	Требуется повтор	Расстраивается, бросает	Медленный	Тревожный	Нужна поддержка и простые задачи
Сидоров А.	…	…	…	…	…	…
…	…	…	…	…	…	…

Условные обозначения:
✓✓ – самостоятельно
○ – с поддержкой
✕ – требуется помощь

Анкета «Интересы и ожидания» (для родителей и ребёнка)

ФИО ребёнка: _____ Дата: _____
ФИО родителя: _____ Контакт: _____

1. **Какие направления в IT тебе наиболее интересны?** (можно отметить несколько))
- ☐ Создание компьютерных игр
- ☐ Разработка сайтов или приложений
- ☐ Создание чат-ботов или автоматизация
- ☐ Анализ данных и искусственный интеллект
- ☐ Другое: _____

2. Был ли у тебя опыт изучения программирования ранее?

☐ Да, изучал(а) Python / Scratch / другие языки
☐ Были только школьные уроки информатики
☐ Нет, я начинаю с нуля

3. Чего ты ожидаешь от программы «Программирование на Python»?

☐ Научиться писать собственный код
☐ Развить логическое и алгоритмическое мышление
☐ Подготовиться к олимпиадам по информатике
☐ Найти друзей с похожими интересами

4**. Есть ли у ребёнка особенности, о которых важно знать педагогу?**
(аллергии, ограничения по здоровью, предпочтения в общении)

_____

Подпись родителя: _____

ТЕКУЩИЙ КОНТРОЛЬ

Назначение: Проверка освоения тем: «Базовые операции», «Условный оператор IF», «Операторы циклов».
Время выполнения: 20–25 минут.
Форма: Письменное тестирование + мини-практическое задание.

Шкала интерпретации результатов:

Баллы	**Критерии**
12–15 баллов	Уверенно ориентируется в синтаксисе, типах данных и логике ветвления/циклов. Способен найти ошибку в коде и написать работающий скрипт
8–11 баллов	Материал усвоен в основном объёме. Допускаются синтаксические неточности, но логика алгоритма верна. Готов к изучению сложных структур данных.
4–7 баллов	Усвоены отдельные фрагменты. Требуется систематическая поддержка, повторение синтаксиса и отработка практических навыков в парном формате.
0–3 балла	Необходима разработка индивидуальной образовательной траектории, усиление наглядно-практической поддержки.

Промежуточное тестирование

1. **Какой символ используется для написания комментариев в Python?**

☐ а) //
☐ б) #
☐ в) /*
☐ г) <!--

2. **Какой тип данных будет у переменной x = 5.5?**

☐ а) int (целое)
☐ б) str (строка)
☐ в) float (вещественное)
☐ г) bool (логическое)

3. **Что выведет на экран команда print(2 ** 3)?**

☐ а) 6
☐ б) 8
☐ в) 5
☐ г) Ошибку

4. **Выберите правильный синтаксис условного оператора:**

☐ а) if x > 5 then print("Да")
☐ б) if (x > 5) { print("Да") }
☐ в) if x > 5: print("Да")

5. **Какой цикл в Python используется для перебора элементов последовательности?**

☐ а) while
☐ б) for
☐ в) do-while
☐ г) loop

6. **Установи соответствие:**

☐ А) = ---- 1) Оператор сравнения (равно ли)
☐ Б) == ---- 2) Оператор присваивания
☐ В) != ---- 3) Оператор "не равно"

7. **Найди ошибку в коде:**
   name = input("Введите имя: ") print("Привет, " + name)
   (Задание с подвохом: ошибки нет, или укажи, что нужно преобразование типов, если бы было сложение с числом. Здесь проверяет внимательность).

8. **Напиши короткую программу (2-3 строки), которая запрашивает у пользователя число и выводит его квадрат.**

   _____
   _____

9. **Назови одно правило цифровой гигиены или техники безопасности при работе за ПК.**

   _____

Ключ для проверки и критерии оценивания:

- Задания 1–5: по 1 баллу за верный ответ (5 баллов).
- Задание 6: 2 балла за полное соответствие (А-2, Б-1, В-3).
- Задание 7: 2 балла (1 балл, если указал, что код верный; 1 балл за обоснование).

- Задание 8: 3 балла (1 балл за input, 1 балл за преобразование в int/float, 1 балл за print с вычислением).
- Задание 9: 2 балла за любой корректный ответ (например, "делать перерывы каждые 20 минут", "не открывать подозрительные файлы").

Максимальное количество баллов: 15.

ИТОГОВАЯ АТТЕСТАЦИЯ

Итоговый проект «Моя программа на Python»

Назначение: Комплексная оценка освоения предметных компетенций программы (включая строки, массивы, словари, основы ООП).
Процедура:

1. **Тема:** Свободная (например, "Текстовая игра", "Калькулятор", "Анализатор оценок", "База данных книг") или модульная (по выбору из предложенных педагогом).
2. **Материалы:** Компьютер с установленной средой разработки (IDLE, PyCharm или онлайн-компилятор).
3. **Время**: 2 занятия (80 минут) на доработку и оформление + защита проекта (3-5 минут на человека).
4. **Поддержка:** Педагог консультирует по архитектуре и отладке, но не пишет код за обучающегося.

Критерии оценки итогового проекта

Критерий	**(Самостоятельно) 3 балла**	**(С поддержкой) 2 балла**	**(Требуется помощь) 1 балл**
Работоспособность кода	Программа запускается без ошибок и полностью решает поставленную задачу.	Программа запускается, но имеет незначительные сбои или работает не во всех сценариях.	Программа не запускается из-за синтаксических или логических ошибок, задача не решена.
Алгоритмическая логика	Использованы оптимальные структуры (циклы, условия, функции/методы). Код эффективен.	Логика верна, но решение избыточно или неоптимально (например, много повторяющегося кода).	Логика нарушена, алгоритм не соответствует задаче, структуры данных используются неверно.
Читаемость и стиль	Код хорошо структурирован, переменные имеют осмысленные имена,	Имена переменных частично непонятны, комментариев мало	Код представляет собой "спагетти", имена переменных

	присутствуют комментарии.	или они неинформативны.	типа `a`, `b`, `x`, комментариев нет.
Творческая самостоятельность	Проявлена инициатива: добавлены уникальные функции, оригинальный сюжет или интерфейс.	Выполнено по образцу или шаблону с минимальными изменениями.	Строгое копирование кода из интернета или учебника без понимания.
Качество защиты проекта	Уверенная презентация, чёткое объяснение алгоритма, грамотные ответы на вопросы.	Презентация сбивчивая, затрудняется объяснить отдельные части своего кода.	Не может объяснить, как работает собственная программа, ответы на вопросы отсутствуют.

Интерпретация результатов:

- 15–18 баллов — высокий уровень освоения программы (Сертификат с отличием / Диплом "Юный программист").
- 10–14 баллов — базовый уровень, программа освоена (Сертификат об успешном окончании).
- 6–9 баллов — начальный уровень, рекомендовано продолжение обучения с акцентом на практику.
- 0–5 баллов — требуется индивидуальная траектория поддержки.

Протокол итоговой аттестации

Группа: _____ Педагог: _____ Дата: _____

№ п/п	ФИО обучающегося	Баллы по критериям (макс. 18)	Уровень освоения	Форма поощрения
1	…	16	Высокий	Диплом "Юный программист"
2	…	14	Базовый	Сертификат участника
3	…	9	Начальный	Сертификат + рекомендации
…	…	…	…	…

Итоговые показатели группы:
Высокий: ___ чел. (***%),***
Базовый: ___ чел. (%),
Начальный: ___ чел. (___%).
Средний балл: ___ / 18.
Рекомендации на следующий учебный год:

_____
_____
_____

Подпись педагога: _____ / _____

Диагностическая карта группы
«Оценка результатов по освоению дополнительной общеразвивающей программы «Программирование на Python"»

ФИО педагога: _____
Группа : _____

№ п/п	**ФИО обучающегося**	**Входная диагностика**	**Промежуточная аттестация**	**Текущий контроль**					**Итоговая аттестация** (Защита проекта + тест)	**ИТОГО**
				Навыки кодинга (синтаксис)	Алгоритмическое мышление	Работа в проекте / команде	Теоретическая подготовка	Практическая подготовка, участие в конкурсах		
1										
2										
3…										

Диагностическая карта заполняется по результатам диагностик:

1. **Входная диагностика:** анкетирование + логическая задача.
2. **Промежуточная аттестация:** письменный тест (разделы 1-4).
3. **Текущий контроль:** педагогическое наблюдение на практических занятиях, проверка домашних заданий, код-ревью.
4. **Итоговая аттестация:** защита итогового программного продукта и итоговое тестирование.

УЧАСТИЕ В МЕРОПРИЯТИЯХ

Название мероприятия: Итоговая защита проектов «Цифровой мир: от идеи до кода»

Дата проведения: 25 мая 2026 года

Место проведения: МБОУ ДО «Центр дополнительного образования детей» Бахчисарайского района Республики Крым (компьютерный класс / актовый зал).

Участники: Обучающиеся программы «Программирование на Python» (12–17 лет), родители (законные представители), педагоги, приглашенные IT-специалисты, гости.

Цель мероприятия: Создание ситуации успеха для каждого обучающегося через публичную презентацию итоговых программных продуктов, демонстрация индивидуального прогресса в освоении алгоритмического мышления и навыков кодирования, формирование опыта публичной защиты технических проектов и развития цифровой грамотности.

Описание участия:

Обучающийся Петров Алексей, 14 лет (пример), принял активное участие в итоговой защите проектов по программе «Программирование на Python».

Формы участия:

1. **Демонстрация программных продуктов** — представил 3 завершенных проекта, разработанных в течение учебного года:
   - «Текстовый квест "Путешествие по Крыму"» (использование условных операторов if/else, циклов while, работа со строками).
   - «Анализатор успеваемости» (работа со списками/массивами, циклы for, функции ввода-вывода данных).
   - «Симулятор электронной очереди» (применение базовых принципов объектно-ориентированного программирования: классы, объекты, методы).
2. **Публичная защита (презентация) проекта** — подготовил и провел защиту своего лучшего проекта «Текстовый квест "Путешеств_by Крыму"» (3–4 минуты):
   - рассказал о постановке задачи, выборе языка Python и структуре программы;
   - вживую продемонстрировал работу приложения и объяснил ключевые участки кода (например, использование словарей для хранения локаций и инвентаря игрока);
   - поделился трудностями (например, отладка бесконечных цикZNов) и способами их решения;
   - уверенно ответил на вопросы гостей и педагогов о логике работы алгоритма.
3. **Оформление проектной документации** — самостоятельно подготовил файл описания проекта (README), составил блок-схему основного алгоритма и оформил раздел цифрового портфолио.

4. **Участие в торжественной части** — получил сертификат об успешном освоении программы «Программирование на Python» за 2025–2026 учебный год.

Достигнутые результаты:

1. Продемонстрировал уверенное владение базовыми конструкциями Python (переменные, циклы, условия, функции, базовые структуры данных).
2. Показал умение декомпозировать сложную задачу и выстраивать логический алгоритм её решения.
3. Проявил творческую самостоятельность и инженерное мышление при выборе тематики проекта и его программной реализации.
4. Приобрел ценный опыт публичного выступления, аргументации технических решений и защиты своей интеллектуальной собственности.
5. Научился оформлять программный код в соответствии с правилами читаемости и документировать свои работы.
6. Получил положительную обратную связь от педагога, родителей и приглашенных специалистов.

Отзыв педагога:

«*Алексей проявил себя как целеустремленный, вдумчивый и технически грамотный обучающийся. За год он прошел путь от написания простейших линейных программ до создания полноценного интерактивного приложения с разветвленным сюжетом. Его код отличается логичностью, наличием поясняющих комментариев и соблюдением базовых правил оформления. Особенно удалось продуманное использование словарей и функций для организации игрового процесса. Алексей уверенно защитил свой проект, грамотно ответил на вопросы о структуре данных и показал высокий уровень мотивации к дальнейшему изучению IT-технологий. Желаю новых профессиональных высот и побед в олимпиадах по информатике!*»

Педагог: _____

Дата: ___ мая 20___ года

Приложение 2

МЕТОДИЧЕСКИЕ МАТЕРИАЛЫ

План-конспект занятия:

Тема: *«Вещественные типы данных в Python. Условный оператор if (полная и неполная формы)»*

Цель: Развитие алгоритмического мышления и формирование первичных навыков написания ветвящихся программ на языке Python с использованием условного оператора if.

Задачи:

1. Предметные:
   - познакомить с понятием вещественных типов данных (float) и их отличиями от целочисленных;
   - изучить синтаксис полной и неполной формы условного оператора if;
   - научиться применять оператор if для решения простых алгоритмических задач.
2. Метапредметные:
   - развивать навыки логического мышления, умения анализировать условие задачи и строить алгоритм;
   - формировать навыки работы в интегрированной среде разработки (IDLE/PyCharm) и поиска ошибок в коде.
3. Личностные:
   - воспитывать внимательность к деталям (синтаксису языка), аккуратность при написании кода (соблюдение отступов), а также уверенность в своих силах при решении нестандартных задач и готовность оказывать взаимопомощь.

Формат: комбинированный (изучение нового материала + практическое выполнение).

Форма: урок-практикум с элементами проблемного диалога и игровой мотивации.

Продолжительность: 45 минут (1 академический час).

Необходимые материалы:

1. Компьютеры с установленной средой разработки (IDLE, PyCharm или онлайн-компилятор).
2. Мультимедийный проектор и экран для демонстрации кода.
3. Раздаточный материал: карточки с алгоритмическими задачами разного уровня сложности или ссылки на интерактивные тренажеры.
4. Памятка (чек-лист) «Правила написания кода на Python» (акцент на двоеточиях и отступах).

Подготовка пространства:

1. Проверка работоспособности ПК, программного обеспечения и проектора.

2. Рассадка учащихся (индивидуально или парами для организации взаимопроверки).
3. Настройка проектора для вывода экрана педагога.
4. На доске или слайде визуальная схема алгоритма ветвления: «Условие → Да (действие 1) / Нет (действие 2)».

Ход занятия

1. Организационный этап (2 мин)

Действия педагога: Приветствие обучающихся, проверка готовности к занятию (включенные ПК, открытая среда разработки), проверка отсутствующих. Краткое напоминание о правилах техники безопасности и цифровой гигиене (правильная осанка, перерывы для глаз).

Действия обучающихся: Откликаются, проверяют рабочее место, настраиваются на продуктивную деятельность.

Методика: словесное воздействие, визуальная проверка готовности.

2. Мотивация (5 мин)

Действия педагога: Задает проблемный вопрос: «Как компьютер принимает решения?». Приводит пример из жизни: *"Если на улице идет дождь, я беру зонт, иначе иду в кепке"*. Предлагает перевести это правило на язык алгоритмов. Демонстрирует на проекторе работу простейшей программы-«предсказателя» или «калькулятора скидки», которая меняет поведение в зависимости от введенных данных.

Действия обучающихся: Отвечают на вопросы, пытаются сформулировать условие и варианты развития событий, связывают жизненный опыт с логикой работы программы.

Методика: проблемный диалог, использование наглядности, активизация жизненного опыта.

3. Изучение нового материала (10 мин)

Действия педагога:

1. Кратко объясняет разницу между целыми (int) и вещественными (float) числами (пример: 3 и 3.0, деление / всегда дает float).
2. Демонстрирует на экране синтаксис неполной формы if:
   if условие:
   действие
   Акцентирует внимание на обязательных элементах: двоеточие в конце строки и отступ (4 пробела) для тела условия.
3. Демонстрирует полную форму if-else.
4. Пишет в реальном времени код программы "Допуск на аттракцион" (запрос роста, проверка условия).

Действия обучающихся: Внимательно наблюдают за экраном, фиксируют ключевые правила в тетради или в виде комментариев в своем файле кода, задают уточняющие вопросы.

Методика: объяснительно-иллюстративный метод, пошаговая демонстрация (скринкаст в реальном времени).

4. Практическая реализация (20 мин)

Действия педагога**:** Дает установку на выполнение практических заданий. Осуществляет обход рабочих мест, оказывает адресную консультативную поддержку (особенно при возникновении ошибок IndentationError или SyntaxError). Поощряет взаимопомощь в парах. Ведет карту педагогических наблюдений.

Действия обучающихся:

- *Задание 1 (Базовое):* Написать программу, которая запрашивает температуру воды и выводит "Можно купаться", если она > 20 градусов, и "Холодно" в противном случае.
- *Задание 2 (Продвинутое):* Написать программу-калькулятор скидки: если сумма покупки > 1000, применить скидку 10%, иначе скидки нет. Вывести итоговую сумму с комментарием.
- Сохраняют файлы в своей рабочей папке с правильным именем (файловая дисциплина).

Методика: практико-ориентированный подход, дифференцированная поддержка, технология взаимного обучения.

5. Рефлексия (6 мин)

Действия педагога: Предлагает обсудить типичные ошибки, встреченные во время практики. Задает вопросы: "Что было самым сложным?", "Где в реальной жизни или в играх мы используем такие условия?". Предлагает провести самооценку с помощью визуальной шкалы (например, поднять карточку: зеленый – "все понял и сделал", желтый – "есть вопросы", красный – "нужна помощь"). Поощряет каждого участника, отмечая успехи.

Действия обучающихся: Кратко рассказывают о своих решениях, анализируют свой опыт, слушают отзыв педагога и сверстников, проводят самооценку.

Методика: диалоговая рефлексия, визуальная шкала оценки, создание ситуации успеха.

6. Завершение занятия и уборка (2 мин)

Действия педагога: Кратко резюмирует: "Сегодня мы научили компьютер принимать решения с помощью оператора if и узнали о вещественных числах. Отличная работа!". Напоминает порядок завершения работы: сохранить файл, корректно закрыть среду разработки, выключить ПК, привести рабочее место в порядок. Сообщает тему следующего занятия («Вложенные условные операторы»).

Действия обучающихся: Сохраняют файлы, закрывают программы, выключают компьютеры, убирают рабочее место.

Методика: словесное руководство, структурированная инструкция, формирование культуры цифрового труда.

СЦЕНАРИЙ ИТОГОВОГО МЕРОПРИЯТИЯ

Выставка-презентация итоговых проектов «Цифровой мир: мой первый код»

Цель мероприятия: создать ситуацию успеха для каждого обучающегося через публичную презентацию результатов года (защиту итогового проекта), зафиксировать индивидуальный прогресс в освоении алгоритмического мышления и сформировать опыт краткой публичной презентации программного продукта.

Задачи мероприятия:

1. Образовательные: демонстрация динамики освоения современных алгоритмов и структур данных (переменные, циклы, массивы, словари, основы ООП), показ применения теоретических знаний при решении практических задач, фиксация освоения программы через выдачу сертификатов.
2. Воспитательные: воспитывать уверенность в своих силах, гордость за результаты интеллектуального труда, ответственность за публичную презентацию; формировать уважение к интеллектуальной собственности и этике цифрового творчества.
3. Развивающие: развивать навыки публичной речи, алгоритмической грамотности и рефлексии: умение лаконично презентовать свой программный продукт, аргументировать выбор структур данных и инструментов, анализировать код сверстников.
4. Социализирующие: вовлечь родителей в образовательный процесс через совместное мероприятие; создать условия для признания достижений обучающихся в социуме; сформировать позитив_imidж программы и организации.

Участники мероприятия: обучающиеся программы «Программирование на Python», родители (законные представители), педагог программы, администрация организации, приглашённые гости (педагоги других программ, партнёры).

Подготовка к мероприятию

Подготовка к защите проектов начинается за 3 недели до мероприятия и включает следующие этапы:

1. **За 3 недели до мероприятия:** Педагог определяет дату, время и место проведения. Обучающиеся начинают отбор и доработку своих итоговых проектов (игры, утилиты, базы данных). Подготавливаются титульные слайды или распечатки архитектуры программы (блок-схемы, фрагменты кода).
2. **За 2 недели до мероприятия:** Рассылка приглашений родителям. Подготовка пространства: определение зон для демонстрации работающих программ на экранах мониторов/ноутбуков. Заказ и

подготовка сертификатов об освоении программы. Подготовка фонового музыкального сопровождения.

3. **За 1 неделю до мероприятия:** Окончательное тестирование (отладка) программ каждым обучающимся. Подготовка файлов для демонстрации (локально или через облачное хранилище). Репетиция краткой презентации проектов с обучающимися (1–2 минуты на проект). Проверка технического оборудования (проектор, работоспособность ПК, наличие интерпретатора Python/среды IDLE).
4. **За 1 день до мероприятия:** Монтаж зоны демонстрации: расстановка компьютеров, проверка запуска программ, размещение распечатанных блок-схем или QR-кодов со ссылками на код. Генеральная репетиция с обучающими: маршрут движения, порядок выступлений. Проверка готовности сертификатов.

Организация пространства:

1. **Зона регистрации:** Стол для встречи гостей. Книга отзывов. Программа мероприятия и список авторов проектов. Фотозона с названием выставки «Цифровой мир: мой первый код».
2. **Зона демонстрации проектов:** Рабочие станции (ПК/ноутбуки), на которых запущены итоговые программы обучающихся. Рядом с каждым ПК размещена карточка с названием проекта, именем автора и кратким описанием функционала.
3. **Зона презентации:** Пространство для сбора гостей и выступающих. Место для выступления педагога и демонстрации проектов на большом экране (через проектор).
4. **Зона награждения:** Стол для размещения сертификатов. Пространство для торжественного вручения документов и группового фотографирования.

СЦЕНАРИЙ МЕРОПРИЯТИЯ

Часть 1. Встреча гостей и ознакомление с проектами (15 минут)

За 15 минут до официального начала гости собираются в зоне регистрации. Педагог встречает гостей, регистрирует их в книге посетителей, раздаёт программу мероприятия. Предлагает гостям в свободном режиме подойти к компьютерам в «Зоне демонстрации проектов» и самостоятельно протестировать программы (например, запустить текстовую игру, ввести данные в программу-калькулятор или базу данных). Обучающиеся занимают места рядом со своими компьютерами. Они готовы кратко рассказать заинтересованным гостям о логике работы программы, ответить на вопросы о том, какие циклы или функции были использованы. Родители осматривают проекты, знакомятся с работами своего ребёнка, задают вопросы педагогу о прогрессе ребёнка в освоении алгоритмического мышления.

Часть 2. Торжественное открытие выставки (10 минут)

По истечении 15 минут свободного осмотра педагог приглашает всех в зону презентации.

Приветственное слово педагога: «Дорогие ребята, уважаемые родители и гости! Сегодня мы открываем итоговую презентацию «Цифровой мир: мой первый код», которая посвящена завершению первого года обучения по программе «Программирование на Python». За этот год наши юные разработчики проделали огромный путь. Они начали с изучения алгоритмов и среды IDLE, освоили базовые операции, научились использовать условные операторы и циклы, разобрались в работе со строками, массивами и словарями, и даже прикоснулись к основам объектно-ориентированного программирования. Сегодня на защите представлено более [количество] уникальных программных проектов. Каждый проект — это результат логического мышления, упорного труда и творческого поиска. Я приглашаю вас увидеть, как абстрактный код превращается в работающие, полезные и интересные продукты. Отдельные слова благодарности я хочу сказать родителям за поддержку, терпение и веру в своих детей. Без вашего участия этот успех был бы невозможен. А теперь я предоставляю слово нашим юным программистам!»

Часть 3. Презентации и защита проектов обучающимися (20 минут)

Педагог предоставляет слово обучающимся для краткой презентации (защиты) своих итоговых проектов.

Формат презентаций: От каждой группы выступают 3–5 обучающихся (по желанию или по очереди). Время выступления одного ребёнка: 1–2 минуты. Ребёнок выходит к проектору или своему компьютеру, кратко рассказывает о задаче, алгоритме решения и использованных инструментах Python.

Примерный текст презентации ребёнка: «Здравствуйте! Меня зовут (имя). Я представляю свой итоговый проект «(название, например: Текстовый квест "Путешествие по Крыму" или "Умный калькулятор")». Я написал эту программу на языке Python. Для реализации я использовал (циклы for/while, условный оператор if, списки/словари). Мне хотелось создать программу, которая (решает задачу / развлекает / помогает в учёбе). Самым сложным было (найти ошибку в коде / правильно организовать вложенные циклы / спроектировать логику словаря), но я справился. Демонстрация работы программы (ребёнок запускает код и показывает результат). Спасибо за внимание!»

Педагог представляет каждого выступающего, задаёт уточняющие вопросы по коду (например: «А почему ты выбрал здесь цикл while, а не for?»), поддерживает и подбадривает выступающих. Зал аплодирует после каждого выступления.

Часть 4. Вручение сертификатов (10 минут)

Педагог приглашает всех обучающихся для торжественного вручения сертификатов.

Церемония вручения: Педагог объявляет: «А теперь настал самый торжественный момент — вручение сертификатов об успешном освоении

программы "Программирование на Python"!» Обучающиеся по одному подходят к столу. Педагог вручает сертификат, называя имя и фамилию ребёнка, и произносит краткие персональные слова поощрения, опираясь на результаты КТП:

- «(Имя), ты отлично освоил работу с массивами и твоя программа работает без ошибок!»
- «(Имя), твои алгоритмы всегда хорошо структурированы, ты настоящий мастер логики!»
- «(Имя), ты смело экспериментировал с функциями и словарями, это высший пилотаж!»

Ребёнок получает сертификат, делает реверанс или кивок, возвращается на место. Включается торжественная музыка. После вручения всем: «Дорогие ребята! Вы все большие молодцы! Каждый из вас прошел путь от "Hello, World!" до создания собственных приложений. Эти сертификаты — подтверждение вашего интеллектуального труда. Я горжусь каждым из вас!»

Часть 5. Завершение мероприятия и рефлексия (5 минут)

Заключительное слово педагога: «Наша презентация подходит к концу. Я хочу поблагодарить всех за внимание, за интерес к цифровому творчеству наших детей. Дорогие ребята, помните: программирование — это не только то, что происходит в кабинете. Это способ мышления, который поможет вам в любой профессии будущего. Продолжайте писать код, решайте олимпиадные задачи, не бойтесь ошибок (багов) — они делают нас сильнее! Я жду вас в следующем учебном году на новых занятиях, где мы будем изучать продвинутые библиотеки и создавать ещё более масштабные проекты! А теперь я приглашаю всех на общее фотографирование и неформальное общение у компьютеров!»

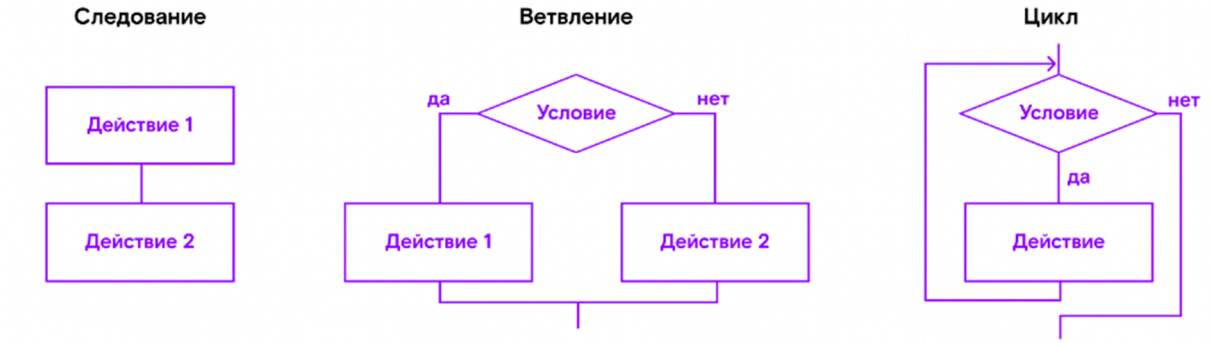
Действия после официального завершения: Общее групповое фотографирование всех участников. Свободное общение гостей с детьми и педагогом. Заполнение книги отзывов. Индивидуальные консультации родителей с педагогом о перспективах обучения ребёнка на следующем уровне.

Иллюстрированный материал по тематике занятий

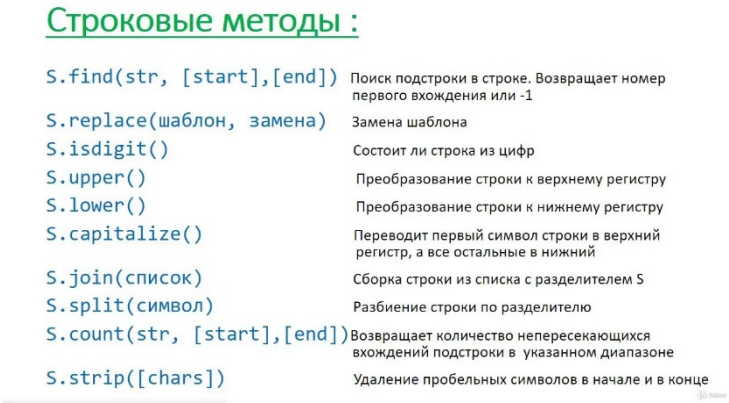
1. Базовые операции в Python:



2. Логика ветвлений



3. Работа с текстом: символьные строки и их методы



4. Словари: хранение и обработка пар «ключ-значение»



Методы словаря

➢ *[... 'clear', 'copy', 'fromkeys', 'get', 'items', 'keys', 'pop', 'popitem', 'setdefault', 'update', 'values']*

clear()	Очищает словарь.
copy()	Возвращает копию словаря.
fromkeys(seql, value])	Создает словарь с ключами из seq и значением value (по умолчанию ***None***).
get(key[, default])	Возвращает значение ключа, а такого нет, не генерирует исключение, а возвращает ***default*** (по умолчанию ***None***).
items()	Возвращает пары (ключ, значение).
keys()	Возвращает ключи в словаре.
pop(key[, default])	Удаляет ключ и возвращает значение. Если ключа нет, то возвращает ***default*** (по умолчанию бросает исключение).
popitem()	Удаляет и возвращает пару (ключ, значение). Если словарь пуст, бросает исключение KeyError. (примечание: словари неупорядочены)
setdefault(key[, default])	Возвращает значение ключа, но если его нет, не бросает исключение, а создает ключ с значением ***default*** (по умолчанию ***None***).
update([other])	Обновляет словарь, добавляя пары (ключ, значение) из other. Существующие ключи перезаписываются. Возвращает ***None*** (не новый словарь!).
values()	Возвращает значения в словаре.

Словарь специальных терминов:

Алгоритм — чёткая и понятная последовательность действий, приводящая к решению поставленной задачи.

Баг (Ошибка) — ошибка в программе, из-за которой она работает неправильно или завершается сбоем.

Библиотека (Модуль) — набор готового кода и функций, который можно подключить к своей программе для расширения её возможностей (например, math, random).

Ветвление — алгоритмическая структура, в которой выполнение того или иного действия зависит от истинности или ложности условия.

IDLE — стандартная интегрированная среда разработки для языка Python, включающая редактор кода и интерактивную оболочку (Shell).

Интерпретатор — программа, которая построчно читает код на Python и сразу же выполняет его.

Комментарий — текст в коде, который игнорируется интерпретатором. Нужен для пояснений программиста (в Python начинается с символа #).

Массив (Список / List) — упорядоченная коллекция элементов, доступ к которым осуществляется по индексу. В Python списки могут содержать элементы разных типов.

Отладка (Debugging) — процесс поиска, анализа и исправления ошибок в программе.

Переменная — именованная область памяти, в которой хранятся данные. Имя переменной должно быть понятным и отражать её содержание.

Синтаксис — набор строгих правил написания кода на языке программирования (расстановка скобок, отступов, двоеточий).

Словарь (Dictionary) – структура данных в Python, хранящая пары «ключ: значение». Позволяет быстро находить данные по уникальному ключу.

Тип данных — характеристика данных, определяющая, какие значения может принимать переменная и какие операции с ней можно выполнять (например, int – целое число, str – строка, bool – логический тип).

Функция – именованный блок кода, который выполняет определённую задачу и может быть вызван многократно.

Цикл — алгоритмическая структура, позволяющая многократно выполнять один и тот же блок кода, пока выполняется заданное условие.

ООП (Объектно-ориентированное программирование) — парадигма программирования, в которой программа строится как совокупность взаимодействующих объектов, обладающих свойствами (атрибутами) и поведением (методами).

Дидактические материалы:

1. Карточка «Правила юного программиста»

Формат: А5, цветная, ламинация.

Содержание:

Я должен:

1. Сохранять свою работу каждые 10–15 минут (Ctrl+S).
2. Давать переменным и файлам понятные имена (на латинице).
3. Оставлять комментарии в слож участках кода.
4. Соблюдать правила техники безопасности при работе за ПК.
5. Просить помощи, если не могу найти ошибку более 10 минут.

Я не должен:

6. Копировать чужой код без понимания его работы.
7. Оставлять открытыми личные данные и пароли.
8. Есть и пить непосредственно над клавиатурой.

2. Памятка «Организация цифрового рабочего места»

Формат: А6, цветная.
Содержание:
МОЁ РАБОЧЕЕ МЕСТО:

1. Шаг 1: Проверь осанку (спина прямая, экран на уровне глаз).
2. Шаг 2: Создай отдельную папку для проекта с понятным именем.
3. Шаг 3: Открой IDLE / редактор кода.
4. Шаг 4: Напиши код, соблюдая отступы (4 пробела).
5. Шаг 5: Запусти и протестируй программу.
6. Шаг 6: Сохрани изменения и закрой программу корректно. *Чек-лист:* □ Осанка правильная □ Папка создана □ Код сохранён □ Ошибок в консоли нет.

3. Шпаргалка «Основные типы данных в Python»

Формат: А4, цветная (для размещения в классе).
Содержание:

- int (Целое число): 5, -10, 2024
- float (Дробное число): 3.14, -0.5, 2.0
- str (Строка): "Привет", 'Python', "123" (всегда в кавычках)
- bool (Логический тип): True (Истина), False (Ложь)
- type(x) – функция, которая покажет тип переменной x.

4. Рабочий лист «Логика условного оператора IF»

Формат: А4, ч/б (для распечатки).
Содержание:
ЗАДАНИЕ 1: Соедини линией условие и результат.

- if 10 > 5: ➔ Выведет "Да"
- if 3 == 4: ➔ Ничего не выведет (или выполнится блок else)

ЗАДАНИЕ 2: Нарисуй блок-схему для проверки: "Если на улице идёт дождь, взять зонт, иначе надеть кепку".

5. Технологическая карта «Работа со списками (массивами)»

Формат: А5, цветная.

Содержание:
КАК РАБОТАТЬ СО СПИСКАМИ:

- Создание: my_list = [1, 2, 3]
- Добавление: my_list.append(4)
- Обращение к элементу (помни, счёт с нуля!): my_list[0] (вернёт 1)
- Длина списка: len(my_list)
- Срез (часть списка): my_list[1:3]
- Попробуй сам: Напиши код, который создаёт список из 5 твоих любимых фильмов и выводит первый и последний.

6. Игра «Собери алгоритм» (Карточки)

Формат: Набор карточек А6 (20 шт.).
Содержание:

На одних карточках написаны этапы решения задачи на естественном языке (например: "1. Спросить возраст", "2. Если возраст >= 18, вывести 'Доступ разрешён'", "3. Иначе вывести 'Доступ запрещён'").
На других карточках – соответствующие строки кода на Python (age = int(input()), if age >= 18:, else:).
Правила: Обучающийся должен разложить карточки в правильном логическом порядке, сопоставив алгоритм и код.

8. Карта достижений «Мой путь в Python»

Формат: А4, цветная (заполняется обучающимся в конце каждого модуля).
Содержание:
МОИ ДОСТИЖЕНИЯ В ПРОГРАММЕ «ЦИФРОВОЙ МИР»
Навыки:
□ Умею запускать код в IDLE и сохранять файлы.
□ Понимаю разницу между типами данных (int, str, float).
□ Могу написать программу с использованием if-else.
□ Умею использовать циклы for и while.
□ Знаю, как создавать и изменять списки и словари.
□ Понимаю базовые принципы ООП (классы и объекты).
Участие:
□ Решил(а) олимпиадную задачу.
□Участвовал(а) в мини-соревновании по программированию.
□ Успешно защитил(а) итоговый проект.

Дата заполнения: _____ *Подпись:* _____

Приложение 3

КАЛЕНДАРНО-ТЕМАТИЧЕСКОЕ ПЛАНИРОВАНИЕ

программы «Программирование на Python»

Количество часов в год – 144; количество часов в неделю – 2

№ п/п	Содержание программного материала	Кол-во часов	Дата по расписанию		Форма аттестации/контроля	Примечание (корректировка)
			по плану	по факту		
	Раздел 1. «Техника безопасности. Общая характеристика языка Python» — 6 часов					
1.	Основные этапы решения задач на компьютере. Алгоритм, его свойства. Инструктаж по ТБ.	2			Входная диагностика (беседа), инструктаж	
2.	Организация рабочего места. Файловая дисциплина. Знакомство со средой IDLE. Горячие клавиши.	2			Наблюдение, мини-опрос	
3.	Структура программы на Python. Основные файлы среды. Пробный запуск первой программы.	2			Практическое упражнение	
	Раздел 2. «Базовые операции в Python» — 16 часов					
4.	Запуск программы. Создание простейших программ. Функции ввода (часть 1).	2			Практическое упражнение	
5.	Запуск программы. Создание простейших программ. Функции ввода (часть 2).	2			Практическое упражнение	
6.	Тип переменной, имя переменной, объявление переменных, оператор присваивания (часть 1).	2			Практическое упражнение	
7.	Тип переменной, имя переменной, объявление переменных, оператор присваивания (часть 2).	2			Практическое упражнение	
8.	Целочисленные типы данных. Деление. Операции в Python (часть 1).	2			Практическое задание, текущее наблюдение	
9.	Целочисленные типы данных. Деление. Операции в Python (часть 2).	2			Практическое задание, текущее наблюдение	
10.	Разбор задач школьного этапа ВсОШ по информатике: «Целочисленная арифметика» (часть 1).	2			Практическое задание, текущее наблюдение	

11.	Разбор задач школьного этапа ВсОШ по информатике: «Целочисленная арифметика» (часть 2).	2			Практикум, проверка кода	
	Раздел 3: «Условный оператор IF» - 16 часов					
12.	Вещественные типы данных в Python. Условный оператор if (полная и неполная формы) (часть 1).	2			Практическое упражнение	
13.	Вещественные типы данных в Python. Условный оператор if (полная и неполная формы) (часть 2).	2			Практическое упражнение	
14.	Инструменты моделирования: and, or, not. Вложенность условных операторов.	2			Практическое упражнение	
15.	Разбор задач школьного этапа ВсОШ по информатике: «Условный оператор» (часть 1).	2			Практикум, карта наблюдений	
16.	Разбор задач школьного этапа ВсОШ по информатике: «Условный оператор» (часть 2).	2			Практикум, карта наблюдений	
17.	Решение задач на использование логических операций.	2			Практическое упражнение	
18.	Тестирование по теме: «Простейшие программы. Ветвления».	2			Тестирование	
19.	Оператор выбора if-elif-else. Решение задач на оператор выбора.	2			Практикум, взаимопроверка	
	Раздел 4. «Операторы циклов for, while» - 22 часов					
20.	Цикл, циклический алгоритм. Синтаксис операторов цикла. Отличия между циклами for, while (часть 1).	2			Практическое задание, текущее наблюдение	
21.	Цикл, циклический алгоритм. Синтаксис операторов цикла. Отличия между циклами for, while (часть 2).	2			Практическое задание, текущее наблюдение	
22.	Циклы с переменной (часть 1).	2			Практическое задание, текущее наблюдение	
23.	Циклы с переменной (часть 2).	2			Практическое задание, текущее наблюдение	

24.	Циклы с постусловием (часть 1).	2			Практическое задание, текущее наблюдение	
25.	Циклы с постусловием (часть 2).	2			Практическое задание, текущее наблюдение	
26.	Вложенные циклы (часть 1).	2			Практическое задание, текущее наблюдение	
27.	Вложенные циклы (часть 2).	2			Практическое задание, текущее наблюдение	
28.	Разбор задач школьного этапа ВсОШ по информатике: «Циклические алгоритмы».	2			Творческое задание, анализ решений	
29.	Решение задач по теме «Циклические алгоритмы» (часть 1).	2			Практическая работа	
30.	Решение задач по теме «Циклические алгоритмы» (часть 2). Тестирование.	2			Тестирование, практическая работа	
	Раздел 5. «Подведение итогов за I полугодие» - 6 часа					
31.	Анализ прогресса. Рефлексия: "Чему я научился?".	2			Защита мини-проекта, групповая дискуссия	
32.	Соревнование "Моя первая программа".	2			Конкурсная работа, оценивание жюри/педагога	
33.	Планирование работы во II полугодии. Выбор темы итогового проекта.	2			Беседа, утверждение темы проекта	
	Раздел 6. «Символьные строки» - 26 часов					
34.	Что такое символьная строка. Типы переменных (часть 1).	2			Практическое упражнение, текущее наблюдение	
35.	Что такое символьная строка. Типы переменных (часть 2).	2			Практическое упражнение, текущее наблюдение	
36.	Сравнение строк. Сцепление строк (часть 1).	2			Практическое упражнение, текущее наблюдение	
37.	Сравнение строк. Сцепление строк (часть 2).	2			Практическое упражнение, текущее наблюдение	

38.	Обращение к символам. Перебор всех символов.	2			Практическое упражнение, текущее наблюдение	
39.	Подстрока. Удаление и вставка.	2			Практическая работа	
40.	Поиск в символьных строках.	2			Практическая работа	
41.	Замена. Преобразования «строка — число» (часть 1).	2			Практическая работа, самопроверка	
42.	Замена. Преобразования «строка — число» (часть 2).	2			Практическая работа, самопроверка	
43.	Символьные строки в функциях.	2			Практикум, код-ревью	
44.	Срезы в Python (часть 1).	2			Практикум, тестирование навыков	
45.	Срезы в Python (часть 2).	2			Практикум, тестирование навыков	
46.	Разбор задач муниципального этапа ВсОШ по информатике: «Символьные строки».	2			Решение олимпиадных задач, анализ ошибок	
	Раздел 7. «Массивы» - 38 часов					
47.	Массивы. Понятие массива. Обращение к элементу массива.	2			Мини-тест, работа с массивами	
48.	Ввод и вывод элементов массива (часть 1).	2			Практикум, текущее наблюдение	
49.	Ввод и вывод элементов массива (часть 2).	2			Практикум, текущее наблюдение	
50.	Заполнение массива случайными числами (часть 1).	2			Практическое задание, проверка кода	
51.	Заполнение массива случайными числами (часть 2).	2			Практическое задание, проверка кода	
52.	Простейшие алгоритмы обработки одномерных массивов. Сумма и количество (часть 1).	2			Практическая работа, расчет результатов	
53.	Простейшие алгоритмы обработки одномерных массивов. Сумма и количество (часть 2).	2			Практическая работа, расчет результатов	
54.	Простейшие алгоритмы обработки одномерных массивов. Поиск min (max) элемента (часть 1).	2			Практическая работа, тестирование алгоритма	

55.	Простейшие алгоритмы обработки одномерных массивов. Поиск min (max) элемента (часть 2).	2			Практическая работа, тестирование алгоритма	
56.	Простейшие алгоритмы обработки одномерных массивов. Выписывание элементов в новый массив (часть 1).	2			Практическая работа, сравнение результатов	
57.	Простейшие алгоритмы обработки одномерных массивов. Выписывание элементов в новый массив (часть 2).	2			Практическая работа, сравнение результатов	
58.	Решение задач по теме: «Алгоритмы обработки массивов» (часть 1).	2			Практическая работа, сравнение результатов	
59.	Решение задач по теме: «Алгоритмы обработки массивов» (часть 2).	2			Практическая работа, сравнение результатов	
60.	Решение задач по теме: «Алгоритмы обработки массивов» (часть 3).	2			Практическая работа, сравнение результатов	
61.	Разбор задач муниципального этапа ВсОШ по информатике: «Массивы» (часть 1).	2			Анализ решений, обсуждение оптимальности	
62.	Разбор задач муниципального этапа ВсОШ по информатике: «Массивы» (часть 2).	2			Анализ решений, обсуждение оптимальности	
63.	Матрицы. Размещение матрицы в памяти компьютера. Заполнение матрицы (часть 1).	2			Практическое задание, визуализация данных	
64.	Матрицы. Размещение матрицы в памяти компьютера. Заполнение матрицы (часть 2).	2			Практическое задание, визуализация данных	
65.	Матрицы. Обработка матриц.	2			Практическая работа	
	Раздел 8. «Словари» – 10 часов					
66.	Словари. Понятие, создание, обращение к элементу (часть 1).	2			Практическое задание, создание базы данных	
67.	Словари. Понятие, создание, обращение к элементу (часть 2).	2			Практическое задание, создание базы данных	

68.	Словари. Практикум по созданию и редактированию.	2			Практическое задание, создание базы данных	
69.	Перебор элементов словаря. Поиск данных (часть 1).	2			Практическая работа, поиск данных	
70.	Перебор элементов словаря. Поиск данных (часть 2).	2			Практическая работа, поиск данных	
	Раздел 9. «Программирование на Python» – 2 часа					
71.	Основы объектно-ориентированного программирования (ООП). Классы и объекты. Итоговое тестирование	2			Тестирование, Итоговое тестирование, защита элемента ООП	
	Раздел 10. «Подведение итогов за II полугодие» – 2 часа					
72.	Защита итоговых проектов. Публичная презентация, оценка по критериям.	2			Публичная защита, презентация, оценка по критериям	
ИТОГ		**144**				

Приложение 4

ЛИСТ КОРРЕКТИРОВКИ

Указываются внесенные в программу изменения, причина корректировки и дата.

№ п/п	Причина корректировка	Дата	Согласование с заместителем директора по УВР (подпись)

ДОКУМЕНТ ПОДПИСАН
ЭЛЕКТРОННОЙ ПОДПИСЬЮ

СВЕДЕНИЯ О СЕРТИФИКАТЕ ЭП

Сертификат 647030360437668574821219143876024766403350371046

Владелец Усатенко Павел Сергеевич

Действителен С 21.01.2026 по 21.01.2027