

Министерство образования и науки Хабаровского края
Краевое государственное автономное нетиповое
общеобразовательное учреждение
«Краевой центр образования»

РАССМОТРЕНО
на заседании
Педагогического совета
КГАНОУ «Краевой центр
образования»
Протокол №1
от «25» августа 2025 г.

УТВЕРЖДАЮ
Генеральный директор
КГАНОУ «Краевой центр
образования»

Черёмухин П.С.
«25» августа 2025 г.



ДОПОЛНИТЕЛЬНАЯ ОБЩЕОБРАЗОВАТЕЛЬНАЯ ОБЩЕРАЗВИВАЮЩАЯ
ПРОГРАММА ТЕХНИЧЕСКОЙ НАПРАВЛЕННОСТИ

«Программирования на языке Python.»

Уровень освоения: базовый
Возраст учащихся: 13 – 18 лет
Общий объем программы в часах: 144 часа

Составители программы:
Носач Т.В., ПДО
Валетова М.А., методист

г. Хабаровск
2025 г.

Раздел 1. Комплекс основных характеристик ДООП

1.1. Пояснительная записка

Дополнительная общеобразовательная общеразвивающая программа «Программирования на языке Python.» имеет техническую направленность (IT – технологии), базовый уровень.

Программа разработана с учетом следующих нормативно-правовых документов:

- Федеральный закон от 29 декабря 2012 года № 273-ФЗ «Об образовании в Российской Федерации»;

- Распоряжение Правительства Российской Федерации от 31 марта 2022 года № 678-р «Концепция развития дополнительного образования детей до 2030 года»;

- Приказ Минпросвещения Российской Федерации от 27 июля 2022 года № 629 «Об утверждении порядка организации и осуществления образовательной деятельности по дополнительным общеобразовательным программам»;

- Постановление Главного государственного санитарного врача Российской Федерации от 28 сентября 2020 года № 28 «Об утверждении санитарных правил СП 2.4.3648-20 «Санитарно-эпидемиологические требования к организации воспитания и обучения, отдыха и оздоровления детей и молодежи» в редакции от 01.03.2025 г.;

- Постановление Главного государственного санитарного врача Российской Федерации от 28 января 2021 года № 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»;

- Распоряжение Министерства образования и науки Хабаровского края от 26 сентября 2019 года № 1321 «Об утверждении методических рекомендаций «Правила персонифицированного финансирования дополнительного образования детей в городском округе, муниципальном районе Хабаровского края»;

- Методические рекомендациями по проектированию дополнительных общеразвивающих программ (включая разноуровневые программы) (Письмо Министерства образования и науки Российской Федерации от 18 ноября 2015 года № 09-3242);

- Методические рекомендациями по формированию механизмов обновления содержания, методов и технологий обучения в системе дополнительного образования детей, направленных на повышение качества дополнительного образования детей, в том числе включение компонентов, обеспечивающих формирование функциональной грамотности и компетентностей, связанных с эмоциональным, физическим, интеллектуальным, духовным развитием человека, значимых для вхождения Российской Федерации в число десяти ведущих стран мира по качеству общего образования, для реализации приоритетных направлений научно-

технологического и культурного развития страны (Письмо Минпросвещения Российской Федерации от 29 сентября 2023 года № АБ-3935/06);

– Приказ краевого государственного автономного образовательного учреждения дополнительного образования «Центр развития творчества детей (региональный модельный центр дополнительного образования детей Хабаровского края)» от 27 мая 2025 года №220 П «Об утверждении Положения о дополнительной общеобразовательной общеразвивающей программе, реализуемой в Хабаровском крае»);

– Устав краевого государственного автономного нетипового образовательного учреждения «Краевой центр образования».

Актуальность обусловлена социальным заказом. Развитие нейросетей, чат-ботов, обработка баз данных предъявляют всё большие требования к знаниям учащихся в области современных IT-технологий, что является неотъемлемой частью профессиональной жизни человека во всех сферах деятельности. Программа ориентирована на изучение языка программирования Python. Это современный язык программирования, основными достоинствами которого являются: кроссплатформенность, бесплатность, простой и понятный синтаксис, высокая читаемость кода программы, богатство возможностей. Область применения языка программирования Python самая различная: от разработки веб-приложений до машинного обучения. Программа строится на концепции подготовки учащихся к профессии программиста.

Педагогическая целесообразность Разработка и внедрение программы обусловлены необходимостью комплексного подхода в обучении учащихся современным языкам программирования. Знания, умения и практические навыки решения актуальных задач программирования, полученные на занятиях, готовят учащихся к самостоятельной проектно-исследовательской деятельности с применением современных информационных технологий, являются драйвером их ранней профессиональной навигации. Использование на занятии различных форм работы предоставляет более широкую возможность учащимся развить свои компетенции, проявить нестандартность мышления и креативность. Использование новейшего оборудования и программного обеспечения, сделает процесс обучения интереснее, ярче и нагляднее.

Новизна программы заключается в том, что она позволяет более глубоко погрузиться в мир программирования, даёт возможность изучать теоретические концепции на практике. Учащиеся приобретают глубокие знания в области технических наук, ценные практические умения и навыки, воспитывают трудолюбие, дисциплинированность, культуру труда, умение работать в коллективе.

Отличительная особенность программы «Программирования на языке Python» заключается в возможности получения обучающимися универсальных компетенций, необходимых при дальнейшем изучении не только информационных технологий, но и предметов гуманитарного и естественно-научного цикла.

Адресат программы: Программа ориентирована на обучающихся возрастной категории от 13 до 18 лет. Состав группы постоянный. Наполняемость в группах: до 15 человек.

Форма обучения: очная.

Объём реализации программы:

Период	Продолжительность занятия	Кол-во занятий в неделю	Кол-во часов в неделю	Кол-во недель	Кол-во часов в год
1 год	2 часа	2	4 ч	36	144 ч
Итого по программе:					144 ч

Режим организации занятий: Занятия в объединении рекомендуется проводить по 2 часа 2 раза в неделю. Продолжительности занятия - 45 минут. После 45 минут занятий организовывается перерыв длительностью 10 минут для проветривания помещения и отдыха обучающихся

1.2. Цель и задачи программы

Цель программы формирование у учащихся компетенций в области программирования на языке Python, способствующих их профессиональному самоопределению; становление и формирование личности учащегося.

Задачи программы:

- сформировать навыки алгоритмического и логического мышления, грамотной разработки эффективных алгоритмов и программ;
- познакомить с принципами и методами структурного программирования;
- сформировать навыки работы в системе программирования Python;
- научить работать с основными структурами данных и типовыми методами обработки этих структур;
- сформировать навыки чтения чужого кода;
- развитие терпения, самоконтроля;
- развитие логического мышления, самостоятельности, познавательного потенциала обучающегося.
- воспитывать упорство в достижении результата;
- стимулировать интерес к изучению профессии, связанной с программированием.

1.3. Учебный план

№ п/п	Название разделов, тем	Количество часов			Формы аттестации/контроля
		Всего	Теория	Практика	
1	Модуль 1. Введение в программирование	36	14	22	
1.1	Основы языка Python	2	1	1	Собеседование
1.2	Ввод и вывод данных	4	1	3	Собеседование, проверка решения задач

1.3	Работа с целыми числами	4	2	2	Собеседование, проверка решения задач
1.4	Условный оператор	8	4	4	Собеседование, проверка решения задач
1.5	Типы данных	6	2	4	Собеседование, проверка решения задач
1.6	Цикл for и while	10	4	6	Собеседование, проверка решения задач
1.7	Тестирование по модулю	2		2	Тест
2	Модуль 2. Строки и списки	36	12	24	
2.1	Проект 1. Игра «Камень, ножницы, бумага»	6	2	4	Собеседование, проверка решения задач
2.2	Строковый тип данных	10	4	6	Собеседование, проверка решения задач
2.3	Списки	10	4	6	Собеседование, проверка решения задач
2.4	Проект 2. «Бот»	8	2	6	Собеседование, проверка решения задач
2.5	Тестирование по модулю	2		2	Тест
3	Модуль 3. Функции и модули	36	11	25	
3.1	Работа с функциями	6	2	4	Собеседование, проверка решения задач
3.2	Работа с модулями	6	2	4	Собеседование, проверка решения задач
3.3	Проект «Автоответчик»	6	2	4	Собеседование, проверка решения задач
3.4	Графический модуль Turtle	8	2	6	Собеседование, проверка решения задач
3.5	Проект 4 «Мой город»	8	3	5	Собеседование, проверка решения задач
3.6	Тестирование	2		2	Тест
4.	Модуль 4. Игры и приложения	36	12	24	Собеседование, проверка решения задач
4.1	Разработка игр	10	4	6	Собеседование, проверка решения задач

4.2	Проект 5 «Шутер»	8	2	6	Собеседование, проверка решения задач
4.3	Публикация и распространение ПО	8	4	4	Собеседование, проверка решения задач
4.4	Введение в объектно-ориентированное программирование	8	2	6	Собеседование, проверка решения задач
4.5	Тестирование по модулю	2		2	Тест
5.	Итоговая аттестация		4		
5.1	Итоговая аттестация	2		2	Тест
	Итого:	144	49	95	

1.4. Содержание учебного плана

МОДУЛЬ 1. Введение в программирование

Тема 1. Основы языка Python (2 ак.ч.)

Теория: История языков программирования. История языка Python. Знакомство с Python и средами программирования. Среда разработки IDLE. Синтаксис языка Python.

Практика: Установка Python и PyCharm. Первая программа

Тема 2. Ввод и вывод данных (2 ак.ч.)

Теория: Вывод данных: команда print(). Ввод данных: команда input(). Необязательные параметры команды print(): sep, end. Переменные. Стандарт PEP 8. Комментарии. Типичные ошибки.

Практика: Решение задач по теме.

Тема 3. Работа с целыми числами (2 ак.ч.)

Теория: Определение переменной. Целочисленный тип данных. Работа с целыми числами. Операции над целыми числами. Построение арифметических выражений. Формат результата. Преобразование строки к целому числу.

Практика: Решение задач по теме.

Тема 4. Условный оператор (2 ак.ч.)

Теория: Условный оператор. Ознакомление с условным оператором if на языке программирования Python. Отступы. Операторы сравнения. Проверка истинности. Инструкция if-elif-else. Множественное ветвление. Программирование вложенных условий. Программирование составных условий.

Практика: Решение задач по теме

Тема 5. Типы данных (2 ак.ч.)

Теория: Числовые типы данных: int, float. Модуль math. Строковый тип данных. Генерирование случайных чисел. Модуль random.

Практика: Решение задач по теме

Тема 6. Цикл for и while (2 ак.ч.)

Теория: Виды операторов цикла. Цикл for. Функция range. Формат оператора цикла с предусловием while. Операторы break и continue. Вложенные циклы. Алгоритм Евклида.

Практика: Решение задач.

Тема 7. Тестирование по модулю (2 ак.ч.)

Практика: Выполнение теста

МОДУЛЬ 2. Строки и списки

Тема 1. Проект 1. Игра «Камень, ножницы, бумага»

Теория: Проектная деятельность. Структура проекта. Концепция будущего проекта. Особенности презентаций в рамках проектных конкурсов. Пример описания проекта. Критерии оценки проекта. Реализация проекта. Оформление и защита проекта

Тема 2. Строковый тип данных (2 ак.ч.)

Теория: Строки и операции над ними. Индексация. Длина строки и отрицательные индексы. Срезы. Методы строк. Применение цикла для обхода строки. Представление строк в памяти компьютера, ASCII и Unicode. Форматирование строк. Срезы строк. Сравнение строк. Операторы для всех типов последовательностей.

Практика: Решение задач по теме.

Тема 3. Списки (2 ак.ч.)

Теория: Списки и операции над ними. Списки: индексы и срезы. Пустые списки. Вывод списков. Функции и методы списков. Методы списков. Встроенные функции.

Практика: Решение задач по теме.

Тема 4. Проект 2. «Бот» (2 ак.ч.)

Теория: Структура проекта. Концепция будущего проекта. Критерии оценки проекта

Практика: Реализация проекта. Оформление и защита проекта

Тема 5. Тестирование по модулю (2 ак.ч.)

Практика: Тестирование (2 ак. ч.)

МОДУЛЬ 3. Функции и модули

Тема 1. Работа с функциями (2 ак.ч.)

Теория: Вспомогательный алгоритм. Понятие «функция». Преимущества использования функций. Структура функция. Объявление функции. Вызов функции. Локальные и глобальные переменные. Функции без параметров. Функции с параметрами. Функции с возвратом нескольких значений.

Практика: Решение задач по теме.

Тема 2. Работа с модулями (2 ак.ч.)

Теория: Стандартные библиотеки. Установка модулей. Импорт стандартных модулей. Создание моделей. Импорт собственных модулей в Python.

Практика: Решение задач по теме.

Тема 3. Проект 3 «Автоответчик» (2 ак.ч.)

Теория: Структура проекта. Концепция будущего проекта. Критерии оценки проекта

Практика: Реализация проекта. Оформление и защита проекта.

Тема 4. Графический модуль Turtle (2 ак.ч.)

Теория: Общие сведения и подходы работы с модулем. Рисование фигур.

Практика: Решение задач по теме.

Тема 5. Проект 4 «Мой город» (2 ак.ч.)

Теория: Структура проекта. Концепция будущего проекта. Критерии оценки проекта

Практика: Реализация проекта. Оформление и защита проекта.

Тема 6. Тестирование по модулю

Практика: Выполнение теста

МОДУЛЬ 4. Игры и приложения

Тема 1. Разработка игр

Теория: Особенности разработки компьютерных игр. Библиотека PyGame. Библиотека PyGame. Установка PyGame. Создание заготовки с фоном. Создание и размещение спрайтов. События клавиатуры. Музыкальное сопровождение.

Практика: Решение задач по теме.

Тема 2. Проект 5 «Шутер»

Теория: Структура проекта. Концепция будущего проекта. Критерии оценки проекта

Практика: Реализация проекта. Оформление и защита проекта.

Тема 3. Публикация и распространение ПО (2 ак.ч.)

Теория: IT-продукт. Тестирование IT-продукта. Создание исполняемого файла. Работа со шрифтами. Интерфейс приложения. Сборка проекта. Модули OS.

Практика: Решение задач по теме.

Тема 4. Введение в объектно-ориентированное программирование (2 ак.ч.)

Теория: Классы. Экземпляры классов. Динамическое изменение. Статические и классовые методы. Специальные методы. Объект как функция. Инкапсуляция. Наследование. Ассоциация. Полиморфизм. Порядок разрешения доступа к атрибутам.

Практика: Решение задач по теме.

Тема 5. Тестирование по модулю.

Практика: Решение задач по теме.

МОДУЛЬ 5. Итоговая аттестация

Тема 1. Итоговая аттестация

Практика: Выполнение теста

Планируемые результаты

Предметные:

- будут знать понятия «алгоритм», «программа», основные конструкции языка программирования Python (ветвления if, операторы цикла while, for, вспомогательные алгоритмы, функции, модули);
- будут знать структуру данных языка программирования Python;
- будут владеть основными приёмами составления программ на языке программирования Python

– смогут отлаживать программный код средствами среды разработки.

Метапредметные:

- развитие терпения, самоконтроля;
- развитие логического мышления, самостоятельности, познавательного потенциала обучающегося;
- смогут самостоятельно разрабатывать алгоритмы достижения целей (решения задач, выполнения проектов и т.д.).

Личностные:

- формирование навыков анализа и самоанализа;
- приобретение начальной профессиональной подготовки по данному направлению, что поспособствует повышению социальной адаптации учащихся после окончания школы.

Раздел 2. Комплекс организационно – педагогических условий

2.1. Календарный учебный график программы (Приложение 1)

Год обучения	Дата начала занятий	Дата окончания занятий	Кол-во учебных недель	Кол-во учебных дней	Кол-во учебных часов	Режим занятий
1 год обучения	01.09.2025 г.	30.05.2026 г.	36	72	144	2 раза в нед. по 2 часа

2.2. Условия реализации программы

Материально-техническое обеспечение:

Персональный компьютер — по 1 шт. на одного обучающегося:

- Не менее 4 ГБ оперативной памяти;
- Процессор с тактовой частотой не менее 1,2 ГГц;
- Диагональ мониторов не менее 12 дюймов;
- Наличие колонок или наушников;
- Свободные 50 ГБ на накопителях;
- компьютеры (ноутбуки) должны быть подключены к единой сети с доступом в Интернет не медленнее 10 Мбит/с;
- презентационное оборудование (проектор с экраном/телевизор с большим экраном) с возможностью подключения к компьютеру (ноутбуку) – 1 комплект;
- флипчарт с комплектом листов / маркерная доска, соответствующий набор письменных принадлежностей – 1 шт.

Программное обеспечение:

- Операционная система — любая, желательно Windows;
- Любой современный браузер (например, Яндекс.Браузер, Google Chrome, Mozilla Firefox, Safari);
- Python версии не ниже 3.6;
- среда разработки. PyCharm Community Edition версии не ниже 2020.1.

Информационное обеспечение:

- Документация Python - <https://docs.python.org/3/>
- "Поколение Python": курс для продвинутых - <https://stepik.org/course/68343/promo;>
- Python для продвинутых - https://ru.hexlet.io/courses/advanced_python;
- Информатика. Алгоритмы и структуры данных на Python. Лекции Хирьянова Т. - http://judge.mipt.ru/mipt_cs_on_python3.

Кадровое обеспечение: Программа реализуется педагогом дополнительного образования.

2.3. Формы аттестации

Для отслеживания результативности и динамики освоения ребенком образовательной программы педагогу необходимо понимать, какой уровень развития тех или иных навыков имеет учащийся при зачислении на обучение. Поэтому проводится входящая диагностика по диагностической карте промежуточной аттестации, чтобы определить уровень развития ребенка по определенным параметрам, заложенным картой. Также используются такие формы и методы контроля, как:

- наблюдение;
- устный опрос;
- устный анализ самостоятельных работ;
- тестирование, самостоятельные работы;
- итоговая аттестация заключается написанием программы для решения одной из предложенных задач.

Наблюдение – необходимый педагогу метод для осуществления промежуточной и текущей аттестации, применяется педагогом постоянно. *Устный опрос* – метод при котором педагог может оценить теоретически знания учащихся. Проводится в конце пройденной темы.

Устный анализ самостоятельных работ – дает возможность учащимся научиться логически мыслить и уметь высказать собственное суждение, поможет педагогу оценить логическое мышление учащихся. Проводится в конце пройденной темы.

Тестирование, самостоятельные работы – помогут учащимся проверить свои знания по пройденным темам и оценить приобретенные умения,

проводятся в конце каждого раздела. Участие в проектной деятельности – это условие, необходимое для становления самостоятельности учащихся, которые сформируют установку на самостоятельность, индивидуальные цели и возможности их реализации.

Итоговая аттестация проводится в конце учебного года. Задачей промежуточной аттестации является определение эффективности педагогического воздействия и полноты освоения образовательной программы.

Формы предъявления и демонстрации образовательных результатов: участие в предметных олимпиадах и конкурсах.

2.4. Оценочный материал

Для отслеживания и фиксации результатов предусмотрены следующие формы контроля: опрос, наблюдение, практическая работа. Опрос позволяют своевременно и быстро выявить сложности, возникающие у обучающихся, при освоении темы занятия. Практическая работа проверяет уровень владения практическими навыками. Наблюдение позволяет оценить групповую и индивидуальную работу обучающихся без непосредственного вмешательства педагога, здесь отслеживаются не только знания и практические навыки, но и личностные результаты, достигнутые обучающимися.

В течение всего срока обучения ведется учет результатов участия каждого обучающегося в конкурсах и соревнованиях различного уровня.

Пример итоговой аттестации (Приложение 2).

Диагностическая карта итоговой аттестации

№	ФИО обучающегося	Параметры				
		Самостоятельность в выполнении задач	Участие в конкурсах	Специальные знания	Алгоритмическое логическое мышление	ИТОГ

1 балл – низкий уровень; 2 балла – средний уровень; 3 балла – высокий уровень.

При подсчете баллов по каждому учащемуся можно определить уровень освоения им программы, также по каждой группе и по объединению в целом.

Определение уровня каждого учащегося выводится после подсчета баллов по всем параметрам:

- от 1 до 5 баллов – низкий уровень освоения программы учащимся;
- от 6 до 9 баллов – средний уровень освоения программы учащимся;
- от 10 до 12 баллов – высокий уровень освоения программы учащимся.

2.5. Методическое обеспечение программы

Методика преподавания включает разнообразные формы, методы и приемы обучения и воспитания. Обоснованность применения различных

методов обусловлена тем, что нет ни одного универсального метода для решения разнообразных творческих задач

Методы обучения

Методы обучения, применяемые в реализации программы, можно систематизировать на основе источника получения знания:

- словесные: рассказ, объяснение, беседа, дискуссия;
- наглядные: демонстрация дидактических материалов, видеофильмов; компьютерные игры.
- практические: работа с аудио- и видеоматериалами, тематические экскурсии, интернет-экскурсии, тренинги, участие в мероприятиях.

Вместе с традиционными методами на занятиях спешно используются активные методы обучения: мозговой штурм, моделирование, метод проектов, метод эвристических вопросов, игровые ситуации, анализ конкретных ситуаций (case-study) и др.

Выбор методов обучения зависит от дидактических целей, от характера содержания занятия, от уровня развития обучающихся.

Формы организации образовательного процесса

Занятия проводятся с использованием различных форм организации учебной деятельности (групповая, фронтальная, индивидуальная, индивидуальная дистанционная, групповая дистанционная). Разнообразные формы обучения и типы занятий создают условия для развития познавательной активности, повышения интереса детей к обучению.

Формы организации учебного занятия

Основной формой проведения учебных занятий является практическое занятие. Однако в ходе реализации программы, педагог вправе применять любую из доступных форм организации учебного занятия: беседа, встреча с интересными людьми, выставка, диспут, защита проектов, игра, конкурс, конференция, круглый стол, лекция, мастеркласс, «мозговой штурм», наблюдение, олимпиада, презентация, семинар, соревнование, чемпионат, экскурсия.

Типы занятий: изучение новой информации, занятия по формированию новых умений, обобщение и систематизация изученного, практическое применение знаний, умений, комбинированные занятия, контрольно-проверочные занятия.

Педагогические технологии, используемые в образовательном процессе

1. Проектная технология, учебно-исследовательская деятельность. На протяжении всего курса обучения учащиеся вовлечены в учебно-исследовательскую деятельность, которая позволяет им находить, обрабатывать, сравнивать и систематизировать информацию, полученную из встреч с интересными людьми, публикаций в сети Интернет. В ходе образовательного процесса учащиеся создают и защищают собственные проекты, учатся методам поиска информации, самопрезентации, которые необходимы им в дальнейшей жизни и профессиональной карьере, на практических занятиях учащиеся выполняют исследовательские проекты. Проектная деятельность позволяет учащимся принять активную гражданскую

позицию, сформировать потребность в участии в общественно полезной деятельности, необходимость быть нужным обществу. На занятиях создаются и реализуются учебные мини-проекты, в которых учащиеся решают учебные задачи на основе построения последовательности этапов от цели к конкретному результату. В процессе обучения осуществляется знакомство учащихся с информационно-коммуникационными технологиями, достижениями науки техники в области инженерной мысли. Современные педагогические технологии в сочетании с современными информационными технологиями могут существенно повысить эффективность образовательного процесса, решить стоящие перед педагогом задачи воспитания всесторонне развитой, творчески свободной личности.

2. Технология развития критического мышления помогает учащимся определять приоритеты, анализировать, оценивать, выявлять ошибки, повысить мотивацию. Осуществляется при совместной работе в группах, при взаимодействии во время выполнения заданий, при диалоге обучающихся между собой и с педагогом. Обязательным условием является сбор данных о динамике обучающегося и анализ его достижений и трудностей.

Алгоритм формирования критического мышления, предполагающий ответы на следующие вопросы:

1. Какова цель данной познавательной деятельности?
2. Что известно?
3. Что делать?
4. Достигнута ли поставленная цель?

Таким образом, критическое мышление - значит «искусство суждения, основанное на критериях». Результат - владение стратегиями критического мышления.

3. Технология имитационной игры – это моделирование реальной деятельности в специально созданных условиях, а её элементы включают в себя взаимосвязанные знаниевые и деятельностные компоненты обучения. Особенности:

- не моделируется труд конкретных работников;
- имитируются лишь некоторые хозяйственные, правовые, экономические, экологические, социально-психологические принципы, определяющие поведение людей и механизмы их действий (в экстремальных ситуациях);
- моделирование только среды, особенности среды знакомы играющим в основном понаслышке, что делает анализ информации более сложным и субъективным;
- общая цель всего игрового коллектива изначально не задана, и для ее достижения самими игроками может быть найден определенный механизм взаимодействия;
- отсутствуют альтернативы, участники должны действовать лишь в предложенных вариантах;
- не программируется конфликтная ситуация (как, например, в деловых играх), а представлены только различные личные (субъективные) интересы участников игры;

– описанные сценарии игр не включают технологии и механизмы специального обучения общению и коллективному принятию решений.

4.Технология проблемного обучения способствует развитию проблемного мышления учащихся и педагога.

Результаты:

- усвоение учащимися системы знаний и способов умственной деятельности; – развитие интеллектуальных умений и навыков учащихся;
- усвоение способов организации познавательной деятельности и формирования познавательной самостоятельности;
- развитие интеллектуальных возможностей, включающих творческие способности и прошлый опыт учащихся.

Проблемный вопрос - это входящий в состав проблемной задачи или отдельно взятый учебный вопрос (вопрос-проблема), требующий ответа на него посредством мышления. Вопрос же, требующий воспроизведения по памяти, не является проблемным. Вопросы, стимулирующие мышление, начинаются с таких вопросительных слов и словосочетаний, как «почему», «отчего», «как (чем) это объяснить», «как это понимать», «как доказать (обосновать)», «что из этого следует (какой вывод)» и т.п. А вопросительные слова «кто», «что», «когда», «где», «сколько», «какой» всегда требуют ответа на основе памяти.

Проблемная задача – учебная проблема с четкими условиями, задаваемыми преподавателем (лектором) или выявленными и сформулированными кем-либо из обучаемых (студентов), и в силу этого получившую ограниченное поле поиска (в отличие от объективно возникающей перед человеком жизненной проблемы) и ставшую доступной для решения всеми обучаемыми (студентами).

Проблемная ситуация – это ситуация познавательного затруднения, вовлекающая учащихся в самостоятельное познание элементов новой темы.

5.Интерактивные технологии направлены на развитие готовности к организации группового общения.

Результаты:

- готовность воспринимать многомерность информацию, работать в режиме полилога;
- способность выбирать и обосновывать выбор методов, форм и техник организации коммуникационного процесса;
- владение психологическими техниками и методами организации коммуникационного процесса.

6.Технология дискуссионного общения включает в себя взаимосвязанные компоненты:

- мотивационный (готовность, желание принять участие в дискуссии);
- познавательный (знание о предмете спора, проблемная ситуация);
- операционно-коммуникативный (умение вести спор, отстаивать свою точку зрения, владеть способами осуществления логических операций);
- эмоционально-оценочный (эмоциональные переживания, потребности, отношения, мотивы, оценки, личностный смысл).

Календарный план воспитательной работы

1.Модуль «Учебное занятие»			
№	Наименование мероприятия	Сроки, дата	
1.1	Проведение тематических уроков: «Урок цифры», «Урок НТИ», «Технологический диктант», «Цифровой диктант», «День науки», «Урок Победы», ««Урока безопасности и навыков безопасного поведения в Интернете, информационной безопасности, повышение правовой грамотности»	В течение учебного года согласно графика	
2.Модуль «Детское объединение»			
2.1	Формирование корпуса волонтеров	Сентябрь, далее – по мере необходимости	
2.2	Обеспечение работы корпуса волонтеров в течение года	Сентябрь-май	
2.3	Организация мероприятий в рамках дня открытых дверей каникулярных школ (мастер-классы, квесты, викторины, лагерные сборы)	Сентябрь-октябрь Июнь	
3. Модуль «Взаимодействие с родителями»			
№	Тема мероприятия	Форма организации	Сроки
3.1	Консультирование семей учащихся попавших в трудную жизненную ситуацию, а также по актуальным вопросам обучения, воспитания и личностного развития детей	Индивидуальные консультации по запросу	В течении года
3.2	Проведение мониторинга эффективности реализации проекта, с целью выявления и анализа степени удовлетворенности родителей.	Анкетирование и диагностика участников проекта	Апрель-Май
3.3	Проведение родительских собраний с включением в повестку следующих вопросов: особенности ранней профориентации и предпрофессиональной подготовки учащихся, роль родителей в профориентации учащихся; организация проектной деятельности обучающихся технической направленности; участие в конкурсных мероприятиях разного уровня в рамках формирования индивидуальных образовательных траекторий учащихся.	Открытые родительские собрания	Октябрь, Апрель
3.4	«День открытых дверей»	Экскурсия, мастер-классы	Сентябрь
4. Модуль «Наставничество»			
№	Наименование мероприятия	Сроки, дата	
4.1	Сопровождение учащихся на мероприятиях различного уровня	Согласно поступающим предложениям	
4.2	Организация системы наставничества	Сентябрь, далее – по мере приема на	

		работу молодых педагогов
4.3	Знакомство с нормативными документами по организации образовательного процесса	Сентябрь - октябрь
4.4	Оказание помощи в овладении методами обучения и воспитания	В течение года
4.5	Практические семинары молодых педагогов: • знакомство и освоение оборудования; • правила заполнения документации; • формирование и оформление программ дополнительного образования; • типы, формы и структура занятий; • психофизиологические особенности детей разного возраста; • деятельностные образовательные практики	Сентябрь - ноябрь
4.6	Посещение учебных занятий и мероприятий	В течение года
4.7	Подготовка молодых педагогов к участию в конкурсных мероприятиях	В течение года
5. Модуль «Профессиональное самоопределение»		
№	Наименование мероприятия	Сроки, дата
5.1	Участие в работе всероссийских профорientационных проектов, созданных в сети интернет: просмотр лекций, решение учебно-тренировочных задач, участие в мастер-классах	В течение года
5.2	Экскурсии с последующим аналитическим обсуждением IT-компаний, промышленных предприятий в г. Хабаровске	В течение года
6. Модуль «Ключевые культурно-образовательные события»		
№	Наименование мероприятия	Срок, дата
6.1	День знаний	Сентябрь
6.2	Дни открытых дверей	Сентябрь, октябрь, май
6.3	День Учителя	Октябрь
6.4	День Народного единства	Ноябрь
6.5	День Матери в России	Ноябрь
6.6	День Героев Отечества	09 Декабря
6.7	Комплекс мероприятий «Новый год в ЦЦОД «IT- куб»»	Декабрь
6.8	День Конституции Российской Федерации	12 Декабря
6.9	День Российской науки	08 февраля
6.10	День защитника Отечества	23 февраля
6.11	Международный женский день	08 марта
6.12	День космонавтики	Апрель
6.13	День памяти о геноциде советского народа нацистами и их пособниками в годы Великой Отечественной войны	19 апреля
6.14	Праздник Весны и Труда	01 мая
6.15	День Победы	09 мая
6.16	День защиты детей	01 июня
6.17	День России	12 июня
6.18	День Памяти и скорби	Июнь
6.19	День любви, семьи и верности	Июль

Раздел 3. Список литературы

Список литературы для педагога:

1. Гаско, Рик. Простой Python просто с нуля [Электронный ресурс] / Рик Гаско. – М.: СОЛОН-Пресс, 2019.
2. Златопольский, Д.М. Программирование: типовые задачи, алгоритмы, методы [Электронный ресурс] / Д.М. Златопольский. – М.: БИНОМ. Лаборатория знаний, 2007.
3. Мюллер, Джон Пол. Python для чайников [Электронный ресурс] / Джон Пол Мюллер. - 2-е изд.: Пер с англ. – СПб.: ООО «Диалектика», 2019.
4. Окулов, С.М. Основы программирования: учебное пособие [Электронный ресурс] / С.М. Окулов. – 10-е изд., электрон. – Москва : Лаборатория знаний, 2020.
5. Пупышев, В.В. 128 задач по началам программирования [Электронный ресурс] / В.В. Пупышев. – М.: БИНОМ. Лаборатория знаний, 2009.
6. Свейгарт, Эл. Учим Python, делая крутые игры [Электронный ресурс] / Эл. Свейгарт. - М: Эксмо, 2018.
7. Седер, Наоми. Python. Экспресс-курс [Электронный ресурс] / Наоми Седер. - 3-е изд. – СПб.: Питер, 2019.
8. К. Ю. Поляков, Е. А. Еремин. Информатика. Углублённый уровень. Учебник для 10 класса в 2 частях. М.: БИНОМ. Лаборатория знаний, 2014. 9. М. Лутц. Изучаем Python. СПб.: Символ-Плюс, 2011.
10. Задачи по программированию. Под ред. С.М.Окулова, М.: БИНОМ. Лаборатория знаний, 2006.
11. С. М. Окулов. Основы программирования. М.: Бином. Лаборатория знаний, 2012.

Список литературы для обучающихся:

1. Босова, Л.Л. Занимательные задачи по информатике [Электронный ресурс] / Л.Л. Босова, А.Ю. Босова, Ю.Г. Коломенская. – 5-е изд. – М.: БИНОМ. Лаборатория знаний, 2013.
2. Златопольский, Д. М. Сборник задач по программированию [Электронный ресурс] / Д.М. Златопольский. – 3-е изд., перераб. и доп. – СПб.: БХВ-Петербург, 2011.
3. Прохоренок, Н.А. Python 3. Самое необходимое [Электронный ресурс] / Н. А. Прохоренок, В. А. Дронов. – 2-е изд., перераб. и доп. – СПб.: БХВ-Петербург, 2019.
4. Пэйн, Брайсон. Python для детей и родителей [Электронный ресурс] / Брайсон Пейн [пер. с англ. М.А. Райтмана]. - М.: Издательство «Э», 2017.
5. Столяров, А.В. Оформление программного кода [Электронный ресурс] / А.В. Столяров. – 2-е изд., испр. и доп. – Москва: МАКС Пресс, 2019.
6. Шень, А. Программирование: теоремы и задачи [Электронный ресурс] / А. Шень. - 6-е изд., дополненное. М.: МЦНМО, 2017.

7. Шуман, Х.Г. Python для детей [Электронный ресурс] / Х.Г. Шуман [пер. с нем. М.А. Райтман]. – М.: ДМК Пресс, 019.
8. М. Лутц. Изучаем Python. СПб.: Символ-Плюс, 2011.
9. Информатика и ИКТ. Задачник-практикум в 2 частях. Под ред. И.Г. Семакина и Е. К. Хеннера. М.: БИНОМ. Лаборатория знаний, 2014.

Интернет-ресурсы

1. Stepik – образовательная платформа и конструктор - <https://stepik.org>
2. Онлайн редактор языка программирования Python - <https://www.onlinepython.com>
3. Облачный блокнот для программирования на Python - <https://colab.research.google.com/>

Календарный учебный график

Месяц	Дата	Тема	Формы контроля	Кол-во часов
Сентябрь	02.09.25	Основы языка Python	Собеседование	2
	04.09.25	Ввод и вывод данных	Собеседование, проверка решения задач	2
	09.09.25	Ввод и вывод данных	Собеседование, проверка решения задач	2
	11.09.25	Работа с целыми числами	Собеседование, проверка решения задач	2
	16.09.25	Работа с целыми числами	Собеседование, проверка решения задач	2
	18.09.25	Условный оператор	Собеседование, проверка решения задач	2
	23.09.25	Условный оператор	Собеседование, проверка решения задач	2
	25.09.25	Типы данных	Собеседование, проверка решения задач	2
	30.09.25	Типы данных	Собеседование, проверка решения задач	2
Октябрь	02.10.25	Типы данных	Собеседование, проверка решения задач	2
	07.10.25	Цикл for и while	Собеседование, проверка решения задач	2
	09.10.25	Цикл for и while	Собеседование, проверка решения задач	2
	14.10.25	Цикл for и while	Собеседование, проверка решения задач	2
	16.10.25	Цикл for и while	Собеседование, проверка решения задач	2
	21.10.25	Цикл for и while	Собеседование, проверка решения задач	2

	23.10.25	Тестирование по модулю	Тест	2
	28.10.25	Проект 1. Игра «Камень, ножницы, бумага»	Собеседование, проверка решения задач	2
	30.10.25	Проект 1. Игра «Камень, ножницы, бумага»	Собеседование, проверка решения задач	2
Ноябрь	06.11.25	Проект 1. Игра «Камень, ножницы, бумага»	Собеседование, проверка решения задач	2
	11.11.25	Строковый тип данных	Собеседование, проверка решения задач	2
	13.11.25	Строковый тип данных	Собеседование, проверка решения задач	2
	18.11.25	Строковый тип данных	Собеседование, проверка решения задач	2
	20.11.25	Строковый тип данных	Собеседование, проверка решения задач	2
	25.11.25	Строковый тип данных	Собеседование, проверка решения задач	2
	27.11.25	Списки	Собеседование, проверка решения задач	2
	02.12.25	Списки	Собеседование, проверка решения задач	2
	04.12.25	Списки	Собеседование, проверка решения задач	2
Декабрь	09.12.25	Списки	Собеседование, проверка решения задач	2
	11.12.25	Списки	Собеседование, проверка решения задач	2
	16.12.25	Проект 2. «Бот»	Собеседование, проверка решения задач	2
	18.12.25	Проект 2. «Бот»	Собеседование, проверка решения задач	2
	23.12.25	Проект 2. «Бот»	Собеседование, проверка решения задач	2
	25.12.25	Проект 2. «Бот»	Собеседование, проверка решения	2

			задач	
	30.12.25	Тестирование по модулю	Тест	2
	13.01.26	Работа с функциями	Собеседование, проверка решения задач	2
Январь	15.01.26	Работа с функциями	Собеседование, проверка решения задач	2
	20.01.26	Работа с функциями	Собеседование, проверка решения задач	2
	22.01.26	Работа с модулями	Собеседование, проверка решения задач	2
	27.01.26	Работа с модулями	Собеседование, проверка решения задач	2
	29.01.26	Работа с модулями	Собеседование, проверка решения задач	2
	03.02.26	Проект «Автоответчик»	Собеседование, проверка решения задач	2
Февраль	05.02.26	Проект «Автоответчик»	Собеседование, проверка решения задач	2
	10.02.26	Проект «Автоответчик»	Собеседование, проверка решения задач	2
	12.02.26	Графический модуль Turtle	Собеседование, проверка решения задач	2
	17.02.26	Графический модуль Turtle	Собеседование, проверка решения задач	2
	19.02.26	Графический модуль Turtle	Собеседование, проверка решения задач	2
	24.02.26	Графический модуль Turtle	Собеседование, проверка решения задач	2
	26.02.26	Проект 4 «Мой город»	Собеседование, проверка решения задач	2
	03.03.26	Проект 4 «Мой город»	Собеседование, проверка решения задач	2
Март	05.03.26	Проект 4 «Мой город»	Собеседование, проверка решения задач	2
	10.03.26	Проект 4 «Мой город»	Собеседование, проверка решения задач	2
	12.03.26	Тестирование	Тест	2
	17.03.26	Разработка игр	Собеседование, проверка решения задач	2
	19.03.26	Разработка игр	Собеседование,	2

			проверка решения задач	
	24.03.26	Разработка игр	Собеседование, проверка решения задач	2
	26.03.26	Проект 5 «Шутер»	Собеседование, проверка решения задач	2
	31.03.26	Проект 5 «Шутер»	Собеседование, проверка решения задач	2
Апрель	02.04.26	Проект 5 «Шутер»	Собеседование, проверка решения задач	2
	07.04.26	Проект 5 «Шутер»	Собеседование, проверка решения задач	2
	09.04.26	Публикация и распространение ПО	Собеседование, проверка решения задач	2
	14.04.26	Публикация и распространение ПО	Собеседование, проверка решения задач	2
	16.04.26	Публикация и распространение ПО	Собеседование, проверка решения задач	2
	21.04.26	Публикация и распространение ПО	Собеседование, проверка решения задач	2
	23.04.26	Введение в объектно-ориентированное программирование	Собеседование, проверка решения задач	2
	28.04.26	Введение в объектно-ориентированное программирование	Собеседование, проверка решения задач	2
	30.04.26	Введение в объектно-ориентированное программирование	Собеседование, проверка решения задач	2
Май	05.05.26	Введение в объектно-ориентированное программирование	Собеседование, проверка решения задач	2
	07.05.26	Введение в объектно-ориентированное программирование	Собеседование, проверка решения задач	2
	12.05.26	Введение в объектно-ориентированное программирование	Собеседование, проверка решения задач	2
	14.05.26	Тестирование по модулю	Тест	2
	19.05.26	Подведение итогов.	Опрос	2
		Итого:		144

Первичный опрос для учащихся по направлению «Программирование на языке Python»

1. Какой из перечисленных инструментов не является средой разработки для Python?
 - PyCharm;
 - VS Code;
 - Google Chrome;
 - IDLE.
2. С помощью какой клавиши можно удалить символы, находящиеся справа от курсора в текстовом редакторе или IDE?
 - Enter;
 - Backspace;
 - Delete;
 - Alt.
3. Какие комбинации клавиш чаще всего используются для переключения раскладки клавиатуры (например, при написании кода)?
 - Ctrl + Shift;
 - Ctrl + C;
 - Alt + Shift;
 - Ctrl + A.
4. Какие операции можно выполнять с файлами Python (.py)?
 - создание, редактирование, выполнение;
 - только создание;
 - только выполнение;
 - только чтение.
5. Щелчок правой кнопкой мыши в IDE используется для:
 - запуска программы;
 - закрытия окна;
 - вызова контекстного меню;
 - перехода к следующей строке кода.
6. С помощью одновременного нажатия буквы и какой клавиши можно напечатать заглавную букву в коде?
 - Ctrl + буква;
 - Alt + буква;
 - Shift + буква;
 - Enter + буква.
7. Программа, которая выполняет код Python построчно, называется:
 - компилятором;

- интерпретатором;
- редактором;
- сервером.

8. На что указывает расширение файла .py?

- на объем данных в файле;
- на местоположение файла;
- на тип приложения, в котором он создан;
- на то, что это файл с кодом Python.

9. Какой из перечисленных операторов не используется для создания циклов в Python?

- for;
- while;
- loop;
- if.

10. Что означает термин «дебаггинг» в программировании?

- удаление ненужных файлов;
- поиск и исправление ошибок в коде;
- запуск программы;
- создание резервной копии кода.

11. Несколько компьютеров, объединенных для совместной работы над проектом, – это:

- топология;
- локальная сеть;
- сетевой адаптер;
- виртуальная машина.

12. Что такое отступ (indentation) в Python?

- специальный символ для обозначения комментариев;
- пробелы или табуляция, определяющие структуру кода;
- запрет на изменение кода;
- автоматическое выравнивание текста.

13. Какой программой нельзя запустить файл Python (.py)?

- Paint;
- Блокнот;
- PyCharm;
- VS Code.

14. Какая клавиша используется для выделения нескольких строк кода, расположенных не подряд?

- Enter;

- Ctrl;
- Shift;
- Alt.

15. Комбинация каких клавиш позволяет быстро сохранить файл с кодом?

- Ctrl + A;
- Ctrl + X;
- Ctrl + S;
- Ctrl + V.

16. Комбинация каких клавиш позволяет отменить последнее действие в редакторе?

- Ctrl + Z;
- Ctrl + C;
- Ctrl + V;
- Ctrl + X.

17. Какого типа данных не существует в Python?

- int;
- str;
- cloud;
- list.

18. Какое расширение имеют файлы с исходным кодом Python?

- .doc, .txt, .rtf;
- .py;
- .mp3, .mpeg, .avi;
- .rar, .zip, .exe.

19. Какая из перечисленных программ не является средой разработки для Python?

- PyCharm;
- GIMP;
- VS Code;
- Thonn.

Образец итоговой аттестации

Краткое описание математической задачи:

Дан список целых чисел. Посчитать, сколько раз в нем встречается каждое число. Например, если дан список [1,1,3,2,1,3,4], то в нем число 1 встречается три раза, число 3 — два раза, числа 2 и 4 — по одному разу.

Для хранения количества каждого встречающегося в списке значения

создадим словарь. В нем ключами будут числа, которые встречаются в списке, а значениями количества в этих чисел в списке. Для примера, приведенного выше, в итоге должен был бы получиться такой словарь: {1:3, 3:2, 2:1, 4:1}.

Пусть в программе будет функция, которая заполняет список случайными числами в диапазоне и количестве, указанными пользователем.

Другая функция будет считать количество каждого значения и заносить данные в словарь. Алгоритм подсчета заключается в следующем. Если очередной элемент списка уже есть в качестве ключа словаря, то следует увеличить значение этого ключа на единицу. Если очередного элемента списка нет в качестве ключа в словаре, то такой ключ следует добавить и присвоить ему значение, равное единице.

Для того, чтобы вывести содержимое словаря в отсортированном по возрастанию ключей виде, используя функции `sorted()`. Она сортирует ключи Словаря и помещает их в список.

Решение:

```
def fill_list(min_bound, max_bound, amount_items,
              filling_list):
    from random import randint
    for i in range(amount_items):
        filling_list.append(randint(min_bound, max_bound))

def analysis(your_list, your_dict):
    for i in your_list:
        if i in your_dict:
            your_dict[i] += 1
        else:
            your_dict[i] = 1

lst = []
dct = {}

mn = int(input('Минимум: '))
mx = int(input('Максимум: '))
qty = int(input('Количество элементов: '))

fill_list(mn, mx, qty, lst)
analysis(lst, dct)

for item in sorted(dct):
    print("%d'%d" % (item, dct[item]))
```

Пример выполнения:

```
Минимум: 1
Максимум: 11
Количество элементов: 20
'1':4
'3':7
'4':2
'6':6
'11':1
```

