

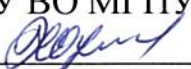
Департамент образования и науки города Москвы

**Государственное автономное образовательное учреждение
высшего образования города Москвы
«Московский городской педагогический университет»**

Средняя общеобразовательная школа

СОГЛАСОВАНО

Председатель экспертного совета
по дополнительному образованию
ГАОУ ВО МГПУ

 /Н.П. Ходакова/
Протокол № 26 от 06 июля 2026 г.

УТВЕРЖДАЮ

И.о. ректора
ГАОУ ВО МГПУ



/Е.Н. Геворкян/

Дополнительная общеразвивающая программа

«Клуб будущих чемпионов (продвинутое программирование)»

(66 часов)

**Уровень программы – базовый
Направленность программы – техническая**

Авторы:
Брынин Г.Э.,
Чеснокова А.В.

Москва, 2026

Раздел 1. ПОЯСНИТЕЛЬНАЯ ЗАПИСКА

Олимпиадное программирование – особый вид деятельности, увлекательный и азартный. Олимпиады по программированию бывают индивидуальными и командными, бывают официальными, дающими льготы при поступлении в высшее учебное заведение. Главное отличие олимпиадного программирования от обычного – необходимость быстро анализировать многословные условия задач, выбирать из обширного набора решаемые, добиваться верного результата любыми средствами.

Для достижения успеха прежде всего необходимо умение читать задачу, увидеть в её описании путь к решению. Затем – опыт программирования, уверенное владение синтаксисом языка и инструментами отладки. Далее – знание стандартных приёмов решения задач, классических алгоритмов. И наконец, воля, внимание, стрессоустойчивость, умение работать и в команде, и самостоятельно.

1.1. Актуальность программы

В современном мире навыки программирования становятся неотъемлемой частью образовательного процесса и профессиональной деятельности. Обучающиеся 8 класса находятся на этапе формирования своих интересов и профессиональных склонностей, что делает изучение программирования особенно актуальным. Дополнительная общеразвивающая программа «Клуб будущих чемпионов (продвинутое программирование)» направлена на углубленное изучение основ программирования, развитие критического мышления, креативности и навыков решения проблем. Это позволит подготовить учащихся к участию в конкурсах, олимпиадах и других мероприятиях, связанных с информационными технологиями.

Цель: формирование у обучающихся углубленных знаний и навыков в области программирования, а также в развитии их творческого потенциала и способности к самостоятельному решению задач.

Задачи:

Обучающие:

- ознакомить учащихся с основными концепциями и языками программирования.

- обучить использованию современных инструментов разработки программного обеспечения.

- развить навыки работы с алгоритмами и структурами данных.

Развивающие:

- способствовать развитию логического мышления и аналитических способностей.

- формировать умение работать в команде и эффективно взаимодействовать с другими участниками проекта.

- развивать навыки проектной деятельности через создание собственных программных продуктов.

Воспитательные:

- воспитывать у учащихся ответственность за качество выполняемой работы.

- формировать интерес к научной и технической деятельности.

- стимулировать активное участие в конкурсах, хакатонах и других мероприятиях.

Планируемые результаты обучения

В результате обучающиеся будут

знать:

- основные концепции программирования и алгоритмизации.

- принципы работы с различными языками программирования.

- основы проектирования программного обеспечения.

уметь:

- разрабатывать простые программы на изучаемых языках.

- анализировать и оптимизировать алгоритмы.

- работать в команде над проектами, делегируя обязанности и идеи.

владеть:

- навыками работы с современными инструментами разработки и отладки программного обеспечения.
- умением представлять и защищать свои проекты перед аудиторией.
- способностью самостоятельно искать информацию и решать возникающие проблемы в процессе программирования.

Категория обучающихся: 8 класс (*предпрофиль*)

Форма обучения: очная

Режим занятий: 2 часа в неделю

Трудоемкость программы: 66 часов

Раздел 2. СОДЕРЖАНИЕ ПРОГРАММЫ

2.1. Учебный план

№ п/п	Наименование разделов (модулей) и тем	Аудиторные учебные занятия, учебные работы			Внеаудитор ная работа	Формы контроля	Трудо емкость
		Всего ауд. часов (ак.час)	Теоретические занятия	Практические занятия	Самостоят. работа		
1	Модуль 1. Введение в программирование	3	2	1			3
1.1	Знакомство с программой и LMS. Установка и настройка среды программирования. PEP8. Ввод? вывод. Переменные	1	1				1
1.2	Условный оператор. Отступы. Операции над строками	2	1	1			2
2	Модуль 2. Алгоритмы и циклы	17	9	8			17
2.1	Сложные условия. Вложенные структуры	2	1	1			2
2.2	Типы данных. Операции над числами	1	1				1
2.3	Приоритет операций. Простейшие функции	2	1	1			2
2.4	Цикл с предусловием. Алгоритм Евклида	2	1	1			2
2.5	Отладка программ	2	1	1			2
2.6	Цикл for. Диапазоны	2	1	1			2
2.7	Булевы переменные. Прерывания и продолжения циклов	2	1	1			2
2.8	Вложенные циклы	2	1	1			2
2.9	Элементы теории множеств. Множества в Python	2	1	1			2
3	Модуль 3. Структуры данных	20	9	11			20
3.1	Строки. Индексация	2	1	1			2
3.2	Строки. Срезы	2	1	1			2

3.3	Знакомство со списками	2	1	1			2
3.4	Кортежи	2	1	1			2
3.5	Списочные выражения. Методы split() и join()	2	1	1			2
3.6	Другие методы списков и строк	2	1	1			2
3.7	Вложенные списки	2	1	1			2
3.8	Знакомство со словарями	2	1	1			2
3.9	Методы словарей	2	1	1			2
3.10	Повторение	2		2			2
4	Модуль 4 Функции и функциональные приемы	18	8	10			18
4.1	Функции. Введение	2	1	1			2
4.2	Функции. Возвращение значений	2	1	1			2
4.3	Возврат из глубины функции. Отладка	2	1	1			2
4.4	Области видимости переменных	2	1	1			2
4.5	Функции. Передача параметров	2	1	1			2
4.6	Функции с переменным числом аргументов	2	1	1			2
4.7	Функции как объект. Лямбда-функции	2	1	1			2
4.8	Функциональная обработка коллекций. Поточковый ввод	2	1	1			2
4.9	Повторение функций	2		2			2
5	Модуль 5 Хранение и обработка данных	8	4	4			8
5.1	Файловая система. Файлы в Python	2	1	1			2

5.2	Хранение данных. Работа с csv-документами.	2	1	1		2
5.3	Работа с файлами json	2	1	1		2
5.4	Библиотеки Python. Модули math, random, datetime	2	1	1		2
	Итого	66	32	34		66

2.2. Рабочая программа

№ п/п	Виды учебных занятий, учебных работ, объем в часах	Содержание
Модуль 1. Введение в программирование		
Тема 1.1 Знакомство с программой и LMS. Установка и настройка среды программирования. PEP8. Ввод? вывод. Переменные	Теоретическое занятие, 1 час	Знакомство с программой дополнительного образования по программированию и Learning Management System (LMS), которая будет использоваться в процессе обучения. Процесс установки и настройки среды программирования для работы с Python, что является основой для дальнейшего изучения языка. Принципы PEP8 — стандарты оформления кода. Ввод и вывод данных, работа с переменными
Тема 1.2 Условный оператор. Отступы. Операции над строками	Теоретическое занятие, 1 час	Основы условных операторов в программировании. Структура условных операторов (if, else, elif), их синтаксис и применение в различных ситуациях. Важность отступов в Python, определяют блоки кода и влияют на его выполнение. Правила оформления отступов и их значение для читаемости кода
	Практическое занятие, 1 час	Создание простых программ, использующих условные конструкции для решения задач, таких как проверка условий и вывод сообщений. Работа с различными методами строк, выполнение задачи на конкатенацию, изменение регистра и форматирование текста
Модуль 2. Алгоритмы и циклы		
Тема 2.1 Сложные условия. Вложенные структуры	Теоретическое занятие, 1 час	Основы сложных условий и вложенных структур в программировании. Структура сложных условий, их формирование с помощью логических операторов (AND, OR, NOT) и применение для более точного контроля за выполнением кода. Важность вложенных условных операторов (if внутри if) для решения более сложных задач и их влияние на логику выполнения программы

	Практическое занятие, 1 час	Использование теоретических знаний о сложных условиях и вложенных структурах для решения практических задач. Разработка программ, применяющих вложенные условные операторы для одновременной проверки нескольких условий
Тема 2.2 Типы данных. Операции над числами	Теоретическое занятие, 1 час	Основные типы данных в программировании: целые числа, числа с плавающей запятой, строки и логические значения. Обсуждение использования типов данных в различных контекстах и их влияние на выполнение программ. Операции над числами: арифметические (сложение, вычитание, умножение, деление), приоритеты операций и правила округления
Тема 2.3 Приоритет операций. Простейшие функции	Теоретическое занятие, 1 час	Понятие приоритета операций в программировании. Операции с более высоким приоритетом, влияние на порядок выполнения выражений, ошибки из-за неправильного понимания приоритетов. Основные арифметические операции: сложение, вычитание, умножение, деление; логические операции: AND, OR, NOT. Правила использования скобок для изменения порядка выполнения
	Практическое занятие, 1 часа	Разработка небольших программ, правильная расстановка операций, использование скобок, получение ожидаемого результата
Тема 2.4 Цикл с предусловием. Алгоритм Евклида	Теоретическое занятие, 1 час	Концепция цикла с предусловием, структура и основные принципы работы. Обсуждение повторения блоков кода при выполнении определенного условия. Синтаксис и семантика циклов в различных языках программирования (например, while и do while). Изучение алгоритма Евклида для нахождения наибольшего общего делителя (НОД) двух чисел, пошаговое выполнение и анализ реализации с использованием цикла с предусловием. Примеры применения алгоритма в реальных задачах
	Практическое занятие, 1 часа	Разработка программы, реализующей алгоритм Евклида для нахождения НОД двух чисел с использованием цикла. Оформление условий, использование циклов для повторения вычислений, обработка ввода данных от пользователя. Отладка кода, проверка корректности работы программы на различных тестовых данных
Тема 2.5 Отладка программ	Теоретическое занятие, 1 час	Основные принципы и методы отладки программ. Понятие отладки, её необходимость и типы ошибок (синтаксические, логические, ошибки времени выполнения). Инструменты и техники отладки: использование отладчиков, вывод отладочной информации, тестирование и логирование. Процесс пошагового анализа кода,

		выявление и исправление ошибок, методы повышения качества кода. Лучшие практики работы с отладкой и основные подходы к тестированию программного обеспечения
	Практическое занятие, 1 час	Работа с заранее подготовленными программами, содержащими различные типы ошибок. Использование отладчиков для пошагового выполнения кода, анализ значений переменных, выявление причин сбоев. Практика написания тестов для проверки корректности работы программ и использование логирования для отслеживания выполнения кода
Тема 2.6 Цикл for. Диапазоны	Теоретическое занятие, 1 час	Понятие цикла for в программировании, применение для работы с диапазонами значений. Синтаксис цикла for, структура и основные элементы: инициализация, условие, шаг. Использование цикла for для итерации по числовым диапазонам, обработка элементов массивов и списков. Примеры использования цикла for: вычисление сумм, нахождение максимальных и минимальных значений, генерация последовательностей. Преимущества цикла for по сравнению с другими циклами
	Практическое занятие, 1 час	Решение задач, использование цикла for для работы с диапазонами. Создание программ для итераций по числовым диапазонам, вычисление сумм, произведений и статистических показателей. Применение вложенных циклов for для решения сложных задач, таких как создание таблиц умножения или графиков
Тема 2.7 Булевы переменные. Прерывания и продолжения циклов	Теоретическое занятие, 1 час	Булевы переменные, их роль в программировании, значения true и false, управление потоком выполнения программ, использование в условиях и циклах. Прерывания и продолжения циклов, команды break и continue, синтаксис и применение для управления выполнением циклов, примеры использования для оптимизации работы программ и повышения эффективности
	Практическое занятие, 1 час	Решение задач с булевыми переменными, управление логикой программ, команды break и continue, управление циклами, условия, поток выполнения, обсуждение решений, обмен опытом, эффективные подходы
Тема 2.8. Вложенные циклы	Теоретическое занятие, 1 час	Понятие вложенных циклов, структура и применение в программировании. Один цикл внутри другого для решения сложных задач, таких как обработка двумерных массивов или создание многоуровневых структур данных. Синтаксис вложенных циклов, примеры использования для решения задач: вывод таблиц, работа с матрицами, генерация паттернов. Дискуссия о преимуществах

		и недостатках вложенных циклов, ситуации неэффективного применения
	Практическое занятие, 1 час	Задачи, требующие использования вложенных циклов для обработки данных и создания сложных алгоритмов. Программы с различными уровнями вложенности: вывод треугольников чисел, создание таблиц умножения, работа с матрицами. Эксперименты с условиями и параметрами циклов, оптимизация решений
Тема 2.9 Элементы теории множеств. Множества в Python	Теоретическое занятие, 1 час	Основы теории множеств: ключевые понятия (множество, элементы, подмножества), операции над множествами (объединение, пересечение, разность, симметричная разность). Способы представления множеств и их свойства. Реализация множеств в Python: встроенный тип данных set, методы работы с множествами (добавление и удаление элементов, проверка принадлежности), применение операций над множествами с использованием встроенных функций. Практические примеры применения теории множеств в программировании
	Практическое занятие, 1 час	Работа с множествами в Python, создание программ, использование операций над множествами для решения задач. Реализация программ для нахождения уникальных элементов из списка, подсчета общих и различных элементов между двумя множествами, создание подмножеств. Эксперименты с методами и функциями, задачи на оптимизацию работы с данными
Модуль 3. Структуры данных		
Тема 3.1 Строки. Индексация	Теоретическое занятие, 1 час	Индексация строк, положительные и отрицательные индексы, доступ к символам, методы работы со строками: конкатенация, срезы, изменение регистра, поиск и замена подстрок. Практические примеры использования строк в задачах программирования
	Практическое занятие, 1 час	Строковые операции в Python, индексация, манипуляция с текстом, извлечение подстрок, изменение формата строк, функции для обработки текстовой информации, эксперименты с методами строк, задачи на обработку данных
Тема 3.2 Строки. Срезы	Теоретическое занятие, 1 час	Понятие строк в Python, свойства и методы работы со строками. Создание строк, операции над строками. Срезы (slice), синтаксис и применение для извлечения подстрок. Манипуляция строками с помощью срезов, работа с индексами, отрицательными индексами и шагом

	Практическое занятие, 1 час	Строковые данные, срезы, подстроки, порядок символов, анализ строк, положительные и отрицательные индексы, шаги, выборка символов, обработка текстовой информации
Тема 3.3 Знакомство со списками	Теоретическое занятие, 1 час	Понятие списков в Python, создание списков, типы данных, свойства списков (изменяемость, порядок элементов), основные операции (добавление, удаление, изменение элементов), методы работы со списками (append(), remove(), sort()), использование индексов и срезов для доступа к элементам списков
	Практическое занятие, 1 час	Работа с различными списковыми данными, операции добавления, удаления и изменения элементов. Решение задач на сортировку и поиск элементов в списках. Использование вложенных списков для представления сложных структур данных. Применение методов списков и разработка небольших программ
Тема 3.4 Кортежи	Теоретическое занятие, 1 час	Понятие кортежей в языке программирования Python. Свойства кортежей: неизменяемость, порядок элементов. Основные операции с кортежами: доступ к элементам по индексам, использование срезов
	Практическое занятие, 1 час	Работа с кортежами, операции доступа к элементам, срезы, задачи на создание и манипуляцию кортежами, использование кортежей в качестве возвращаемых значений из функций, разработка небольших программ, хранение и передача данных
Тема 3.5 Списочные выражения. Методы split() и join()	Теоретическое занятие, 1 час	Понятие списочных выражений (list comprehensions) в языке программирования Python. Структура списочных выражений, включая использование циклов и условий. Применение методов split() и join() для работы со строками: разбиение строк на списки по заданному разделителю и объединение списков в строки с использованием определенного разделителя. Примеры использования этих методов в реальных задачах
	Практическое занятие, 1 час	Создание новых списков с помощью списочных выражений, использование метода split() для разбиения строк на элементы списка, применение метода join() для объединения элементов списка в строку. Работа с реальными данными, обработка текстовых файлов и вводимых строк, решение задач на фильтрацию и преобразование данных
Тема 3.6 Другие методы списков и строк	Теоретическое занятие, 1 час	Методы работы со списками и строками в языке программирования Python. Основные методы списков: append(), extend(), insert(), remove(), pop(), сортировка и поиск (sort(), index). Методы строк: replace(), find(), upper(), lower() и другие.

		Эффективная обработка данных и решение различных задач
	Практическое занятие, 1 час	Решение задач, связанных с добавлением, удалением и изменением элементов в списках, манипуляция строками. Работа с реальными данными, обработка списков значений и текстовых строк, использование методов для фильтрации, сортировки и форматирования
Тема 3.7 Вложенные списки	Теоретическое занятие, 1 час	Концепция вложенных списков в языке программирования Python. Определение вложенных списков, создание и применение. Способы доступа к элементам вложенных списков, включая индексацию и срезы. Применение вложенных списков для организации данных, таких как матрицы и таблицы. Примеры задач, где использование вложенных списков является наиболее эффективным решением
	Практическое занятие, 1 час	Работа с вложенными списками, применение теоретических знаний на практике. Решение задач, связанных с созданием, изменением и доступом к элементам вложенных списков. Выполнение операций, таких как добавление и удаление элементов, перебор вложенных списков с помощью циклов. Работа с реальными данными, представление и обработка матриц или таблиц.
Тема 3.8 Знакомство со словарями	Теоретическое занятие, 1 час	Понятие словарей в языке программирования Python. Устройство и применение словарей. Основные операции со словарями: добавление, изменение и удаление элементов. Преимущества использования словарей для хранения и обработки данных в различных задачах
	Практическое занятие, 1 час	Создание и манипуляция словарями в Python. Задачи, связанные с добавлением, изменением и удалением пар "ключ-значение". Перебор элементов словаря с помощью циклов. Использование методов .get(), .keys(), .values() и .items()
Тема 3.9 Методы словарей	Теоретическое занятие, 1 час	Основные методы словарей в Python. Обсуждение доступных методов, их назначения и синтаксиса использования. Методы: .get(), .keys(), .values(), .items(), .update(), .pop(), .clear(). Примеры, иллюстрирующие манипуляцию данными в словарях
	Практическое занятие, 1 час	Задачи, требующие использования различных методов для извлечения, обновления и удаления данных из словарей. Применение методов .get(), .update() и .pop() в реальных сценариях, например, при работе с данными о студентах или товарах. Эксперименты с методами и наблюдение результатов действий

Тема 3.10 Повторение	Практическое занятие, 2 часа	Решение задач на повторение, изученных в модуле тем
Модуль 4 Функции и функциональные приемы		
Тема 4.1 Функции. Введение	Теоретическое занятие, 1 час	Понятие функции в программировании. Основные аспекты: определение функции, структура, параметры, возвращаемые значения. Преимущества использования функций: повторное использование кода, улучшение читаемости программ. Синтаксис определения и вызова функций на примерах. Основные виды функций: встроенные и пользовательские
	Практическое занятие, 1 час	Создание собственных функций, использование различных параметров и возвращаемых значений. Решение практических задач, требующих написания функций для выполнения конкретных задач, таких как вычисление математических операций или обработка данных
Тема 4.2 Функции. Возвращение значений	Теоретическое занятие, 1 час	Понятие возвращаемых значений в функциях. Обсуждение определения и использования возвращаемых значений в программировании, важность их правильного определения. Изучение различных типов данных, которые могут быть возвращены из функции, синтаксиса ключевого слова <code>return</code> . Рассмотрение примеров использования возвращаемых значений для решения практических задач и влияние возвращаемых значений на поток выполнения программы.
	Практическое занятие, 1 час	Создание функций, возвращающих различные типы данных (числа, строки, списки). Решение задач с использованием возвращаемых значений для расчетов и обработки информации. Правильное использование ключевого слова <code>return</code> и передача данных между функциями
Тема 4.3 Возврат из глубины функции. Отладка	Теоретическое занятие, 1 час	Концепция возврата значений из вложенных функций, механизмы отладки кода, вызов функций на разных уровнях вложенности, использование отладчиков, вывод информации в консоль, методы поиска ошибок в коде. Примеры правильной организации возврата значений из глубины функции, выявление и исправление ошибок в процессе разработки
	Практическое занятие, 1 час	Примеры кода с несколькими уровнями вложенных функций и корректным возвратом значений. Инструменты отладки для выявления ошибок. Техники логирования для отслеживания выполнения программы. Самостоятельное нахождение и исправление ошибок, оптимизация кода
Тема 4.4 Области видимости	Теоретическое занятие, 1 час	Изучение концепции областей видимости переменных в программировании. Определение

переменных		области видимости, типы областей (глобальная, локальная и блочная), влияние на доступность переменных в коде. Правила определения области видимости, понятие "замыкания" и его применение. Примеры, иллюстрирующие влияние областей видимости на поведение программы и возможные ошибки, связанные с неправильным использованием переменных
	Практическое занятие, 1 час	Работа с кодом, определение и изменение областей видимости переменных, исправление ошибок доступа к ним. Создание собственных функций и блоков кода для демонстрации доступности переменных в зависимости от их области видимости
Тема 4.5 Функции. Передача параметров	Теоретическое занятие, 1 час	Основы функций в программировании, структура и назначение. Способы передачи параметров в функции: по значению и по ссылке. Различия между методами, влияние на работу программы и управление памятью. Примеры использования функций с различными параметрами и возвращаемыми значениями
	Практическое занятие, 1 час	Разработка собственных функций, экспериментирование с передачей параметров, написание функций с несколькими параметрами, использование значений по умолчанию, работа с переменными, переданными по ссылке, решение практических задач, применение функций для обработки данных и выполнения расчетов
Тема 4.6 Функции с переменным числом аргументов	Теоретическое занятие, 1 час	Концепция функций с переменным числом аргументов, гибкость и универсальность функций, произвольное количество входных данных, синтаксис и особенности работы с переменными аргументами, использование списков, кортежей, специальные операторы (*args и **kwargs в Python), примеры применения в реальных задачах
	Практическое занятие, 1 час	Разработка функций с переменным числом аргументов, экспериментирование с их использованием, написание функций для обработки списков, вычисление статистических значений, создание агрегирующих функций, решение практических задач, применение функций для обработки данных в различных форматах
Тема 4.7 Функции как объект. Лямбда-функции	Теоретическое занятие, 1 час	Концепция функций как объектов в программировании. Присвоение функций переменным, передача в качестве аргументов другим функциям, возврат из функций. Способы определения и использования функций: обычные функции и лямбда-функции. Синтаксис лямбда-функций, преимущества и недостатки, случаи целесообразного использования. Примеры

		применения лямбда-функций в различных контекстах: работа с коллекциями данных, функциональное программирование
	Практическое занятие, 1 час	Разработка собственных функций с использованием обычных определений и лямбда-выражений. Написание функций для обработки списков, фильтрации данных и применения трансформаций с использованием лямбда-функций. Решение практических задач, требующих применения функций как объектов в контексте функционального программирования
Тема 4.8 Функциональная обработка коллекций. Потоковый ввод	Теоретическое занятие, 1 час	Концепция функциональной обработки коллекций данных и методы потокового ввода. Эффективная обработка и трансформация данных в коллекциях: списки, множества, словари. Основные функции высшего порядка: map, filter, reduce и их применение для обработки данных. Понятие потокового ввода и вывода, работа с файлами и стандартными потоками. Примеры использования функциональных методов для обработки данных из различных источников
	Практическое занятие, 1 час	Разработка программ с использованием функций высшего порядка для обработки списков и других коллекций, чтение данных из файлов, обработка данных с использованием функционального подхода. Работа с реальными наборами данных, применение фильтрации, трансформации и агрегации данных
Тема 4.9 Повторение функций	Практическое занятие, 2 часа	Решение задач на повторение, изученных в модуле тем
Модуль 5 Хранение и обработка данных		
Тема 5.1 Файловая система. Файлы в Python	Теоретическое занятие, 1 час	Основы файловой системы и принципы работы с файлами в Python. Различные типы файлов (текстовые и бинарные). Основные операции: создание, чтение, запись и удаление файлов. Использование встроенных функций Python для работы с файлами, включая контекстные менеджеры для безопасного открытия и закрытия файлов
	Практическое занятие, 1 час	Создание простых программ для чтения и записи текстовых и бинарных файлов; выполнение операций по обработке данных из файлов (подсчет строк, слов и символов); использование контекстных менеджеров для безопасной работы с файлами; реализация обработки исключений для управления возможными ошибками
Тема 5.2 Хранение данных. Работа с csv-документами	Теоретическое занятие, 1 час	Формат CSV (Comma-Separated Values) как популярный способ хранения и обмена табличными данными. Структура CSV-файлов, преимущества и недостатки, основные правила

		работы с ними. Использование библиотеки CSV в Python для чтения и записи данных в формате CSV. Варианты обработки данных: фильтрация, сортировка и преобразование
	Практическое занятие, 1 час	Создание, чтение и запись CSV-файлов с использованием библиотеки CSV в Python. Обработка данных из CSV-файлов: фильтрация строк по критериям, сортировка данных, выполнение простых вычислений. Опыт работы с различными форматами данных и кодировками
Тема 5.3 Работа с файлами json	Теоретическое занятие, 1 час	Формат JSON (JavaScript Object Notation) как популярный формат для обмена данными между клиентом и сервером, а также для хранения структурированных данных. Основные принципы работы с JSON, его структура, синтаксис и особенности, включая работу с вложенными объектами и массивами. Использование библиотеки json в Python для сериализации (преобразование объектов Python в JSON) и десериализации (преобразование JSON в объекты Python)
	Практическое занятие, 1 час	Создание, чтение и запись JSON-файлов с использованием библиотеки json в Python. Сериализация различных структур данных, таких как списки и словари, в формат JSON. Десериализация JSON-данных обратно в объекты Python
Тема 5.4 Библиотеки Python. Модули math, random, datetime	Теоретическое занятие, 1 час	Три ключевых модуля стандартной библиотеки Python: math, random и datetime. Основные функции и методы каждого модуля. Модуль math: работа с математическими функциями и константами (синус, косинус, логарифмы, π). Модуль random: генерация случайных чисел и выборок, полезных для создания игр и моделирования. Модуль datetime: управление датами и временем, включая создание, форматирование и арифметику с датами.
	Практическое занятие, 1 час	Использование функций из модуля math для выполнения математических расчетов, генерация случайных чисел и выборок с помощью модуля random, работа с датами и временем с использованием модуля datetime. Создание небольших проектов, таких как генератор случайных паролей, калькулятор для выполнения математических операций и программа для отслеживания событий по датам

Раздел 3. ФОРМЫ АТТЕСТАЦИИ И ОЦЕНОЧНЫЕ МАТЕРИАЛЫ

Аттестация по общеразвивающей программе не предусмотрена.

Раздел 4. ОРГАНИЗАЦИОННО-ПЕДАГОГИЧЕСКИЕ УСЛОВИЯ РЕАЛИЗАЦИИ ПРОГРАММЫ

4.1. Литература

Основная:

1. Аллен Б. Дауни. Основы Python. Научитесь думать как программист. – М.: Миф, 2023.
2. Бриггс, Джейсон. Python для детей. Самоучитель по программированию / Джейсон Бриггс. пер. с англ. Станислава Ломакина. [науч. Ред. Д. Абрамова]. – М.: Манн, Иванов и Фебер, 2022.
3. Доусон Майкл. Програмируем на Python – Спб.: Питер, 2022
4. Ирв Кальб. Объектно-ориентированное программирование с помощью Python. – М.: Бомбора, 2024.
5. Камалова Ю. Программирование на Python с нуля. – М.: Бомбора, 2026.
6. Мэтиз Э. Изучаем Python: программирование игр, визуализация данных, веб-приложения. – Спб.: Питер, 2025
7. Принс Саймон. Машинное обучение. От основ до продвинутых моделей. – М.: Бомбора, 2025.
8. Фостер Патриша. Программирование на Python с нуля. Учимся думать как программисты, осваиваем логику языка и пишем первый код. – М.: Бомбора, 2022.

Дополнительная:

1. Васильев А.Н. Программирование на Python в примерах и задачах. – М.: Бомбора, 2020
2. Лааксонен Антти Олимпиадное программирование. 2-е издание, обновленное и дополненное. – М.: ДМК Пресс, 2021
3. Маран М.М. Программирование. Сборник задач. – Спб.: Лань, 2021

4.2. Материально-технические условия реализации программы

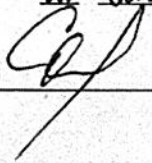
№	виды оборудования
1.	Аудитория

2.	Компьютеры
3.	Проектор
4.	Интерактивная доска (проекционный экран)
5.	Интегрированная среда разработки и обучения на языке Python

Утверждено на педагогическом совете Средней общеобразовательной школы

Протокол № 16 от «20» июня 2026 г.

Зам. директора

 / Омарова С.О. /

**Экспертное заключение
на дополнительную общеразвивающую программу**

Наименование программы, кол-во часов:

«Клуб будущих чемпионов (продвинутое программирование)» (66 часов)

	Критерии экспертизы и вопросы, подлежащие рассмотрению	Экспертная оценка Да/Нет	Примечание эксперта
А. Экспертиза оформления материалов программы			
1.	Наименование программы на титульном листе совпадает с наименованием в тексте	Да	
Б. Соответствие основным нормативным требованиям к структуре, объему и оформлению программы:			
1.	Экспертиза раздела 1 «Пояснительная записка»		
1.1.	Отражена актуальность программы	Да	
1.2.	Наименование программы соответствует ее направленности	Да	
1.3.	Сформулирована цель и задачи программы	Да	
1.4.	Представлена возрастная категория обучающихся	Да	
1.5.	Форма обучения способствует достижению планируемых результатов	Да	
1.6.	Срок обучения, режим обучения способствуют достижению планируемых результатов	Да	
1.7.	Цели, задачи, планируемые результаты, сроки и режим обучения соответствуют уровню программы (ознакомительный, базовый, углубленный)	Да	
2.	Экспертиза раздела 2 «Содержание программы»		
2.1.	Представлен учебный (тематический) план программы	Да	
2.2.	Имеется рабочая программа	Да	
2.3.	В программе кратко раскрыто содержание тем, указаны виды учебных занятий и учебных работ, срок их освоения	Да	
3.	Экспертиза раздела 3 «Форма аттестации и оценочные материалы» на наличие пунктов раздела		
3.1.	Описаны вид аттестации, формы контроля, вид оценочных материалов итоговой (при наличии) и промежуточной (при наличии) аттестации	Да	

3.2.	Описаны оценочные средства контроля (контрольно-измерительные материалы)	–	Аттестация по программе не предусмотрена
4.	Экспертиза раздела 4 «Организационно-педагогические условия реализации программ»		
4.1.	Учебно-методическое и информационное обеспечение программы соответствует всем видам учебной деятельности, предусмотренным программой	Да	
4.2.	Перечень рекомендуемой основной и дополнительной (при наличии) литературы содержит современные и общедоступные источники. Перечень основной литературы должен содержать источники последних 5 лет	Да	
4.3.	Перечисленные Интернет-ресурсы достоверны (при наличии) ¹	–	
4.4.	Указанное материально-техническое обеспечение программы соответствует направленности и содержанию программы	Да	
4.5.	Соблюдение требований к оформлению программы	Да	

ИТОГОВОЕ ЗАКЛЮЧЕНИЕ

Программа рекомендована к реализации в образовательном процессе

Эксперт:

К.п.н., доцент
кафедра педагогических технологий
непрерывного образования
Институт непрерывного
образования ГАОУ ВО МГПУ



(подпись)

Пичугин С.С.
(Ф.И.О.)

¹ Могут не указываться авторами программы. В этом случае ставится прочерк

**Экспертное заключение
на дополнительную общеразвивающую программу**

Наименование программы, кол-во часов:

«Клуб будущих чемпионов (продвинутое программирование)» (66 часов)

	Критерии экспертизы и вопросы, подлежащие рассмотрению	Экспертная оценка Да/Нет	Примечание эксперта
А. Экспертиза оформления материалов программы			
1.	Наименование программы на титульном листе совпадает с наименованием в тексте	Да	
Б. Соответствие основным нормативным требованиям к структуре, объему и оформлению программы:			
1.	Экспертиза раздела 1 «Пояснительная записка»		
1.1.	Отражена актуальность программы	Да	
1.2.	Наименование программы соответствует ее направленности	Да	
1.3.	Сформулирована цель и задачи программы	Да	
1.4.	Представлена возрастная категория обучающихся	Да	
1.5.	Форма обучения способствует достижению планируемых результатов	Да	
1.6.	Срок обучения, режим обучения способствуют достижению планируемых результатов	Да	
1.7.	Цели, задачи, планируемые результаты, сроки и режим обучения соответствуют уровню программы (ознакомительный, базовый, углубленный)	Да	
2.	Экспертиза раздела 2 «Содержание программы»		
2.1.	Представлен учебный (тематический) план программы	Да	
2.2.	Имеется рабочая программа	Да	
2.3.	В программе кратко раскрыто содержание тем, указаны виды учебных занятий и учебных работ, срок их освоения	Да	
3.	Экспертиза раздела 3 «Форма аттестации и оценочные материалы» на наличие пунктов раздела		
3.1.	Описаны вид аттестации, формы контроля, вид оценочных материалов итоговой (при наличии) и промежуточной (при наличии) аттестации	Да	

3.2.	Описаны оценочные средства контроля (контрольно-измерительные материалы)	Да	
4.	Экспертиза раздела 4 «Организационно-педагогические условия реализации программы»		
4.1.	Учебно-методическое и информационное обеспечение программы соответствует всем видам учебной деятельности, предусмотренным программой	Да	
4.2.	Перечень рекомендуемой основной и дополнительной <i>(при наличии)</i> литературы содержит современные и общедоступные источники. Перечень основной литературы должен содержать источники последних 5 лет	Да	
4.3.	Перечисленные Интернет-ресурсы достоверны <i>(при наличии)</i> ¹	Да	
4.4.	Указанное материально-техническое обеспечение программы соответствует направленности и содержанию программы	Да	
4.5.	Соблюдение требований к оформлению программы	Да	

ИТОГОВОЕ ЗАКЛЮЧЕНИЕ

Программа рекомендована к реализации в образовательном процессе

А.Е. Васильева
ФИО эксперта


 Подпись

¹ Могут не указываться авторами программы. В этом случае ставится прочерк