

**ПРИНЯТА**  
Педагогическим советом АНО ОШ ЦПМ  
(протокол от 28 августа 2023 г. № 73)

**УТВЕРЖДЕНА**  
приказом директора АНО ОШ ЦПМ  
от 29 августа 2023 г. № 408

**РАБОЧАЯ ПРОГРАММА**  
**учебного предмета «Информатика»**  
для обучающихся 8–9 классов математико-информационного предпрофиля

Москва, 2023 год

## **Пояснительная записка**

Программа по информатике составлена на основе требований к результатам освоения основной образовательной программы основного общего образования, представленных в ФГОС ООО, а также федеральной программы воспитания.

Программа по информатике даёт представление о целях, общей стратегии обучения, воспитания и развития обучающихся средствами информатики на углублённом уровне, устанавливает обязательное предметное содержание, предусматривает его структурирование по разделам и темам, определяет распределение его по классам (годам изучения).

Программа по информатике определяет количественные и качественные характеристики учебного материала для каждого года изучения, в том числе для содержательного наполнения разного вида контроля (промежуточной аттестации обучающихся, всероссийских проверочных работ, государственной итоговой аттестации). Программа по информатике является основой для составления авторских учебных программ и учебников, тематического планирования курса учителем.

## **Содержание учебного предмета**

### **8 класс**

#### **Раздел 1. Язык программирования Python.**

##### **Тема 1.1. Арифметические операции.**

Ввод и вывод целых чисел, переменные, операции сложения, вычитания, деления с остатком (в т.ч. для отрицательных чисел), взятие остатка от деления, возведение в степень, именование переменных.

##### **Тема 1.2. Условный оператор.**

Переменные логического типа, логические выражения, логические связки, условный оператор if, оператор «иначе» else, оператор «иначе если» elif, вложенные условные операторы.

##### **Тема 1.3. Цикл while.**

Цикл «пока» while, понятие пошаговой отладки, задачи о подсчете суммы последовательности и поиске минимума и максимума в последовательности, операторы управления циклом break и continue, else для циклов while, вложенные циклы.

##### **Тема 1.4. Вещественные числа и строки.**

Хранение вещественных чисел в памяти, операции над вещественными числами, погрешности и точность вычислений, преобразования вещественных чисел. Хранение строк в памяти, операции над строками, срезы и стандартные методы обработки строк.

##### **Тема 1.5. Функции и рекурсия.**

Область применения функций, механизм запуска функций, создание собственных функций, передача параметров и получение результата работы функции, локальные и глобальные переменные, выделение памяти в стеке, рекурсия, примеры решения задач на обработку последовательностей с помощью рекурсивных функций.

### **Тема 1.6. Цикл for.**

Кортежи, итерируемые объекты (кортежи, строки), функция range с одним, двумя и тремя параметрами, вложенные циклы for, инструкции управления циклом for: break и continue, примеры задач, решаемых с помощью циклов for.

### **Тема 1.7. Списки.**

Понятие списка, отличие от кортежа, изменяемые объекты и ссылки на них, чтение списка и добавление элементов, передача и получение списков в качестве параметров в функции, вложенные списки, рекурсивная обработка вложенных списков, примеры решений задач на списки.

### **Тема 1.8. Использование встроенной сортировки.**

Метод sort для списков, функция sorted для итерируемых объектов, сортировка составных объектов, параметр reversed, параметр key и примеры его использования, лямбда-функции для сортировок.

### **Тема 1.9. Множества.**

Множества из чисел, строк и кортежей, основы хеширования объектов, хеш-таблица как структура данных для представления множества, работа с отдельными элементами множеств (добавление, поиск, удаление), работа с множествами (объединение, пересечение, симметрическая разность), создание множеств из других типов данных.

### **Тема 1.10. Словари.**

Понятие ключ-значение, создание словарей, поиск в словаре, добавление и удаление элементов, проход по всем значениям словаря, создание обратного словаря для поиска по ключу, типы ключей.

### **Тема 1.11. Функциональное программирование.**

Зачем нужно функциональное программирование, основные встроенные функции для работы с последовательностями: map, enumerate, zip и др., комбинаторные функции, логические функции для последовательностей, методика написания программ в функциональном стиле.

### **Тема 1.12. Объектно-ориентированное программирование.**

Основные понятия объектно-ориентированного программирования (ООП): наследование, инкапсуляция и полиморфизм, классы и объекты, методы и поля класса, статические переменные, создание собственных классов, переопределение стандартных методов, иерархия классов, создание классов-наследников, функции isinstance и type.

## **Раздел 2. Алгоритмы и структуры данных для олимпиад по информатике начального уровня.**

### **Тема 2.1. Олимпиадные задачи на моделирование.**

Составные части олимпиадной задачи: легенда, форматы ввода и вывода, ограничения и примеры, интуитивное понятие сложности алгоритма, оценка времени работы, построение математической модели по условию задачи.

### **Тема 2.2. Линейный поиск.**

Задача поиска минимума и максимума в последовательности, минимум и максимум с дополнительными требованиями, поиск индексов минимального и максимального элементов, поиск элемента по значению, подсчет количества вхождений, второй максимум и минимум, барьерный метод.

### **Тема 2.3. Частичные суммы, обработка двумерных массивов.**

Предварительный подсчет суммы на префиксах для быстрого подсчета суммы элементов на отрезке, подсчет произведения на отрезке, перебор соседних элементов в двумерных массивах с помощью массивов  $dx$ ,  $dy$  (в зависимости от топологии), барьеры вокруг двумерных массивов для избавления от краевых эффектов, заполнение двумерных массивов разными способами: змейкой, спиралью и т.п.

### **Тема 2.4. Квадратичные сортировки.**

Сортировка пузырьком, вставка элемента в список, сортировка вставкой, сдвиг элементов, сортировка выбором, оценка сложности сортировок, сложность сортировок по сравнениям и присваиваниям, простые задачи на сортировку и жадный алгоритм.

### **Тема 2.5. Арифметика и арифметические алгоритмы.**

Поиск делителей числа, оценка количества делителей, проверка числа на простоту, разложение на простые делители, решето Эратосфена, наибольший общий делитель, наименьшее общее кратное, алгоритм Евклида, быстрое возведение в степень, основные операции над матрицами.

## **9 класс**

### **Раздел 2. Алгоритмы и структуры данных для олимпиад по информатике начального уровня (продолжение).**

### **Тема 2.6. Простые структуры данных.**

Стек, проверка правильности скобочного выражения, проверка корректности xml-документа, подсчет значения арифметического выражения с использованием обратной польской записи, поиск ближайшего меньшего соседа за линейное время для всех элементов массива, использование очереди, использование дека, поиск минимума в скользящем окне за линейное время.

### **Тема 2.7. Бинарный поиск.**

Реализация бинарного поиска, бинарный поиск в массивах, подсчет числа элементов, бинарный поиск по ответу, бинарный поиск для вещественных чисел, тернарный поиск, бинарный поиск по производной функции.

### **Тема 2.8. Эффективные алгоритмы сортировок.**

Сортировка подсчетом, поразрядная сортировка, побайтовая сортировка, слияние отсортированных массивов, сортировка слиянием, сортировка во внешней памяти, эффективный подсчет числа перестановок элементов при сортировке пузырьком.

### **Тема 2.9. Быстрая сортировка и порядковые статистики.**

Понятие опорного элемента, перестановка массива относительно опорного элемента, быстрая сортировка и ее сложность в среднем и худшем случае, оптимизации быстрой сортировки, нахождение порядковой статистики за линейное время, медиана массива.

### **Тема 2.10. Методы, использующие сортировку.**

Метод двух указателей, сортировка событий, события на прямой и на окружности, отрезки и интервалы событий, подсчет числа пересекающихся событий, двумерные алгоритмы для получения сертификата, сжатие координат.

### **Тема 2.11. Генерация комбинаторных объектов.**

Рекурсивная генерация комбинаторных объектов: перестановок, подмножеств, подмножеств заданного размера, разбиение множества на все возможные подмножества, решение NP-полных задач полным перебором и базовые методы оптимизации перебора.

### **Тема 2.12. Одномерное динамическое программирование.**

Метод динамического программирования, реализация динамического программирования с помощью рекурсии, числа Фибоначчи, обобщенные числа Фибоначчи, наибольшая возрастающая последовательность, задачи подсчета количества решений и поиска оптимального решения с помощью динамического программирования.

### **Тема 2.13. Двумерное динамическое программирование.**

Использование двух параметров в задачах на динамическое программирование, задача о наибольшей общей подпоследовательности, задачи подсчета числа решений и поиска оптимального решения с использованием двух параметров, динамическое программирование на подотрезках, рекурсия с меморизацией.

### **Тема 2.14. Задача о рюкзаке.**

Формулировка задачи о рюкзаке, NP-полнота задачи, методы оптимизации задачи о рюкзаке для случая целых весов предметов, использование кольцевого буфера для экономии памяти, задача о рюкзаке с ценностью предметов, восстановление сертификата в задаче о рюкзаке.

### **Тема 2.15. Задачи на лабиринтах.**

Определение лабиринта как двумерного массива с проходимыми и непроходимыми клетками, существование пути между клетками, барьерный метод, кратчайший путь между клетками, кратчайший путь с несколькими концами, кратчайший путь с несколькими концами, кратчайший путь с несколькими началами и концами.

### **Тема 2.16. Представление графов, обход в ширину.**

Способы хранения графов: матрицы смежности, списки ребер, списки смежности, преобразования типов представления графа, обход в ширину на графе и его реализации с помощью списков вершин, находящихся на заданном расстоянии и с помощью очереди.

### **Тема 2.17. Обход в глубину и его применения.**

Рекурсивная реализация обхода в глубину, проверка графа на связность, дерево обхода в глубину, типы ребер: прямые, обратные и поперечные, поиск компонента связности, раскраска графа в два цвета, проверка наличия циклов в ориентированном и неориентированном графе, восстановление циклов.

## **Раздел 3. Язык программирования C++.**

### **Тема 3.1. Арифметика, условный оператор, циклы.**

Ввод и вывод на языке C++, особенности целочисленной арифметики на C++, создание переменных, особенности условных операторов в C++, оператор switch, циклы while, for и do-while.

### **Тема 3.2. C++: структуры данных и функции.**

Вектор, итераторы произвольного доступа, стандартные функции для работы с вектором, реаллокация памяти и порча итераторов вектора при расширении, undefined behavior, кортежи, пары, сравнение кортежей и пар.

### **Тема 3.3. C++: символы и строки.**

Типы данных char и string в C++, коды символов, таблица ascii, чтение строки до перевода строки и по словам, стандартные методы и функции для строк, строка как массив символов, посимвольный ввод, чтение числовых и символьных данных одновременно.

### **Тема 3.4. C++: указатели и итераторы.**

Понятие указателя и ссылки, стандартные функции STL для работы с векторами и массивами, отличие итератора от указателя, односвязные и двусвязные списки, внутреннее устройство структур данных stack, deque и queue в STL.

### **Тема 3.5. C++: set, multiset, map.**

Понятие о бинарных деревьях поиска: построение, поиск, вставка и удаление элементов, использование бинарного дерева поиска для реализации множества, реализация бинарного дерева поиска в STL, удаление элементов по значению и итератору, мультимножество, словари.

## **Планируемые результаты освоения учебного предмета**

Предметные результаты освоения обязательного предметного содержания, установленного данной рабочей программой, отражают сформированность у обучающихся умений:

– пояснять на примерах различия между позиционными и непозиционными системами счисления;

- записывать и сравнивать целые числа от 0 до 1024 в различных позиционных системах счисления (с основаниями 2, 8, 16); выполнять арифметические операции над ними;
- раскрывать смысл понятий «высказывание», «логическая операция», «логическое выражение»;
- записывать логические выражения с использованием дизъюнкции, конъюнкции и отрицания, определять истинность логических выражений, если известны значения истинности входящих в него переменных, строить таблицы истинности для логических выражений;
- раскрывать смысл понятий «исполнитель», «алгоритм», «программа», понимая разницу между употреблением этих терминов в обыденной речи и в информатике;
- описывать алгоритм решения задачи различными способами, в том числе в виде блок-схемы;
- составлять, выполнять вручную и на компьютере несложные алгоритмы с использованием ветвлений и циклов для управления исполнителями, такими как Робот, Черепашка, Чертёжник;
- использовать константы и переменные различных типов (числовых, логических, символьных), а также содержащие их выражения; использовать оператор присваивания;
- использовать при разработке программ логические значения, операции и выражения с ними;
- анализировать предложенные алгоритмы, в том числе определять, какие результаты возможны при заданном множестве исходных значений;
- создавать и отлаживать программы на одном из языков программирования (Python, C++, Паскаль, Java, C#, Школьный Алгоритмический Язык), реализующие несложные алгоритмы обработки числовых данных с использованием циклов и ветвлений, в том числе реализующие проверку делимости одного целого числа на другое, проверку натурального числа на простоту, выделения цифр из натурального числа;
- разбивать задачи на подзадачи; составлять, выполнять вручную и на компьютере несложные алгоритмы с использованием ветвлений, циклов и вспомогательных алгоритмов для управления исполнителями, такими как Робот, Черепашка, Чертёжник;
- составлять и отлаживать программы, реализующие типовые алгоритмы обработки числовых последовательностей или одномерных числовых массивов (поиск максимумов, минимумов, суммы или количества элементов с заданными свойствами) на одном из языков программирования (Python, C++, Паскаль, Java, C#, Школьный Алгоритмический Язык);
- раскрывать смысл понятий «модель», «моделирование», определять виды моделей; оценивать адекватность модели моделируемому объекту и целям моделирования;
- использовать графы и деревья для моделирования систем сетевой и иерархической структуры; находить кратчайший путь в графе;
- выбирать способ представления данных в соответствии с поставленной задачей (таблицы, схемы, графики, диаграммы) с использованием соответствующих программных средств обработки данных;
- использовать электронные таблицы для обработки, анализа и визуализации числовых данных, в том числе с выделением диапазона таблицы и упорядочиванием (сортировкой) его элементов;
- создавать и применять в электронных таблицах формулы для расчётов с использованием встроенных арифметических функций (суммирование и подсчёт значений,

отвечающих заданному условию, среднее арифметическое, поиск максимального и минимального значения), абсолютной, относительной, смешанной адресации;

- использовать электронные таблицы для численного моделирования в простых задачах из разных предметных областей;

- использовать современные интернет-сервисы (в том числе коммуникационные сервисы, облачные хранилища данных, онлайн-программы (текстовые и графические редакторы, среды разработки)) в учебной и повседневной деятельности;

- приводить примеры использования геоинформационных сервисов, сервисов государственных услуг, образовательных сервисов сети Интернет в учебной и повседневной деятельности;

- использовать различные средства защиты от вредоносного программного обеспечения, защищать персональную информацию от несанкционированного доступа и его последствий (разглашения, подмены, утраты данных) с учётом основных технологических и социально-психологических аспектов использования сети Интернет (сетевая анонимность, цифровой след, аутентичность субъектов и ресурсов, опасность вредоносного кода);

- распознавать попытки и предупреждать вовлечение себя и окружающих в деструктивные и криминальные формы сетевой активности (в том числе кибербуллинг, фишинг).

#### Тематическое планирование учебного предмета

Раздел / тема	Количество ак. ч.	Ссылка на ЭОР
<b>8 класс</b>		
<b>Раздел 1. Язык программирования Python</b>	<b>96</b>	<a href="#">Онлайн-курс</a>
Тема 1.1. Арифметические операции	8	<a href="#">Онлайн-курс</a>
Тема 1.2. Условный оператор	8	<a href="#">Онлайн-курс</a>
Тема 1.3. Цикл while	8	<a href="#">Онлайн-курс</a>
Тема 1.4. Вещественные числа и строки	8	<a href="#">Онлайн-курс</a>
Тема 1.5. Функции и рекурсия	8	<a href="#">Онлайн-курс</a>
Тема 1.6. Цикл for	8	<a href="#">Онлайн-курс</a>
Тема 1.7. Списки	8	<a href="#">Онлайн-курс</a>
Тема 1.8. Использование встроенной сортировки	8	<a href="#">Онлайн-курс</a>
Тема 1.9. Множества	8	<a href="#">Онлайн-курс</a>
Тема 1.10. Словари	8	<a href="#">Онлайн-курс</a>
Тема 1.11. Функциональное программирование	8	<a href="#">Онлайн-курс</a>
Тема 1.12. Объектно-ориентированное программирование	8	<a href="#">Онлайн-курс</a>
<b>Раздел 2. Алгоритмы и структуры данных для олимпиад по информатике начального уровня</b>	<b>40</b>	
Тема 2.1. Олимпиадные задачи на моделирование	8	<a href="#">Онлайн-курс</a>
Тема 2.2. Линейный поиск	8	<a href="#">Онлайн-курс</a>
Тема 2.3. Частичные суммы, обработка двумерных массивов	8	<a href="#">Онлайн-курс</a>

Тема 2.4. Квадратичные сортировки	8	<a href="#">Онлайн-курс</a>
Тема 2.5. Арифметика и арифметические алгоритмы	8	<a href="#">Онлайн-курс</a>
<b>ИТОГО</b>	<b>136</b>	
<b>9 класс</b>		
<b>Раздел 2. Алгоритмы и структуры данных для олимпиад по информатике начального уровня (продолжение)</b>	<b>96</b>	
Тема 2.6. Простые структуры данных	8	<a href="#">Онлайн-курс</a>
Тема 2.7. Бинарный поиск	8	<a href="#">Онлайн-курс</a>
Тема 2.8. Эффективные алгоритмы сортировок	8	<a href="#">Онлайн-курс</a>
Тема 2.9. Быстрая сортировка и порядковые статистики	8	<a href="#">Онлайн-курс</a>
Тема 2.10. Методы, использующие сортировку	8	<a href="#">Онлайн-курс</a>
Тема 2.11. Генерация комбинаторных объектов	8	<a href="#">Онлайн-курс</a>
Тема 2.12. Одномерное динамическое программирование	8	<a href="#">Онлайн-курс</a>
Тема 2.13. Двумерное динамическое программирование	8	<a href="#">Онлайн-курс</a>
Тема 2.14. Задача о рюкзаке	8	<a href="#">Онлайн-курс</a>
Тема 2.15. Задачи на лабиринтах	8	<a href="#">Онлайн-курс</a>
Тема 2.16. Представление графов, обход в ширину	8	<a href="#">Онлайн-курс</a>
Тема 2.17. Обход в глубину и его применения	8	<a href="#">Онлайн-курс</a>
<b>Раздел 3. Язык программирования C++</b>	<b>40</b>	
Тема 3.1. Арифметика, условный оператор, циклы	8	<a href="#">Онлайн-курс</a>
Тема 3.2. C++: структуры данных и функции	8	<a href="#">Онлайн-курс</a>
Тема 3.3. C++: символы и строки	8	<a href="#">Онлайн-курс</a>
Тема 3.4. C++: указатели и итераторы	8	<a href="#">Онлайн-курс</a>
Тема 3.5. C++: set, multiset, map	8	<a href="#">Онлайн-курс</a>
<b>ИТОГО</b>	<b>136</b>	
<b>ВСЕГО</b>	<b>272</b>	