

ПРИНЯТА  
Педагогическим советом АНО ОШ ЦПМ  
(протокол от 6 июля 2022 г. № 50)

УТВЕРЖДЕНА  
приказом директора АНО ОШ ЦПМ  
от 6 июля 2022 г. № 123-ОД22

**РАБОЧАЯ ПРОГРАММА**  
по учебному предмету **«Информатика»**  
для обучающихся 10 – 11 классов  
(профильный уровень)

Составитель:  
М.С. Густокашин

Москва, 2022 г.

**Оглавление**

Планируемые результаты освоения учебного предмета .....	3
Содержание учебного предмета .....	7
10 класс .....	7
11 класс .....	8
Тематическое планирование учебного предмета .....	10

## Планируемые результаты освоения учебного предмета

В результате изучения учебного предмета «Информатика» на уровне среднего общего образования:

Выпускник на **профильном уровне** научится:

- кодировать и декодировать тексты по заданной кодовой таблице; строить неравномерные коды, допускающие однозначное декодирование сообщений, используя условие Фано; понимать задачи построения кода, обеспечивающего по возможности меньшую среднюю длину сообщения при известной частоте символов, и кода, допускающего диагностику ошибок;

- строить логические выражения с помощью операций дизъюнкции, конъюнкции, отрицания, импликации, эквиваленции; выполнять эквивалентные преобразования этих выражений, используя законы алгебры логики (в частности, свойства дизъюнкции, конъюнкции, правила де Моргана, связь импликации с дизъюнкцией);

- строить таблицу истинности заданного логического выражения; строить логическое выражение в дизъюнктивной нормальной форме по заданной таблице истинности; определять истинность высказывания, составленного из элементарных высказываний с помощью логических операций, если известна истинность входящих в него элементарных высказываний; исследовать область истинности высказывания, содержащего переменные; решать логические уравнения;

- строить дерево игры по заданному алгоритму; строить и обосновывать выигрышную стратегию игры;

- записывать натуральные числа в системе счисления с данным основанием; использовать при решении задач свойства позиционной записи числа, в частности признак делимости числа на основание системы счисления;

- записывать действительные числа в экспоненциальной форме; применять знания о представлении чисел в памяти компьютера;

- описывать графы с помощью матриц смежности с указанием длин ребер (весовых матриц); решать алгоритмические задачи, связанные с анализом графов, в частности задачу построения оптимального пути между вершинами ориентированного ациклического графа и определения количества различных путей между вершинами;

- формализовать понятие «алгоритм» с помощью одной из универсальных моделей вычислений (машина Тьюринга, машина Поста и др.); понимать содержание тезиса Черча – Тьюринга;

- понимать и использовать основные понятия, связанные со сложностью вычислений (время работы и размер используемой памяти при заданных исходных данных; асимптотическая сложность алгоритма в зависимости от размера исходных данных); определять сложность изучаемых в курсе базовых алгоритмов;

- анализировать предложенный алгоритм, например определять, какие результаты возможны при заданном множестве исходных значений и при каких исходных значениях возможно получение указанных результатов;

- создавать, анализировать и реализовывать в виде программ базовые алгоритмы, связанные с анализом элементарных функций (в том числе приближенных вычислений), записью чисел в позиционной системе счисления, делимостью целых чисел; линейной

обработкой последовательностей и массивов чисел (в том числе алгоритмы сортировки), анализом строк, а также рекурсивные алгоритмы;

- применять метод сохранения промежуточных результатов (метод динамического программирования) для создания полиномиальных (не переборных) алгоритмов решения различных задач; примеры: поиск минимального пути в ориентированном ациклическом графе, подсчет количества путей;

- создавать собственные алгоритмы для решения прикладных задач на основе изученных алгоритмов и методов;

- применять при решении задач структуры данных: списки, словари, деревья, очереди; применять при составлении алгоритмов базовые операции со структурами данных;

- использовать основные понятия, конструкции и структуры данных последовательного программирования, а также правила записи этих конструкций и структур в выбранном для изучения языке программирования;

- использовать в программах данные различных типов; применять стандартные и собственные подпрограммы для обработки символьных строк; выполнять обработку данных, хранящихся в виде массивов различной размерности; выбирать тип цикла в зависимости от решаемой подзадачи; составлять циклы с использованием заранее определенного инварианта цикла; выполнять базовые операции с текстовыми и двоичными файлами; выделять подзадачи, решение которых необходимо для решения поставленной задачи в полном объеме; реализовывать решения подзадач в виде подпрограмм, связывать подпрограммы в единую программу; использовать модульный принцип построения программ; использовать библиотеки стандартных подпрограмм;

- применять алгоритмы поиска и сортировки при решении типовых задач;

- выполнять объектно-ориентированный анализ задачи: выделять объекты, описывать на формальном языке их свойства и методы; реализовывать объектно-ориентированный подход для решения задач средней сложности на выбранном языке программирования;

- выполнять отладку и тестирование программ в выбранной среде программирования; использовать при разработке программ стандартные библиотеки языка программирования и внешние библиотеки программ; создавать многокомпонентные программные продукты в среде программирования;

- устанавливать и деинсталлировать программные средства, необходимые для решения учебных задач по выбранной специализации;

- пользоваться навыками формализации задачи; создавать описания программ, инструкции по их использованию и отчеты по выполненным проектным работам;

- разрабатывать и использовать компьютерно-математические модели; анализировать соответствие модели реальному объекту или процессу; проводить эксперименты и статистическую обработку данных с помощью компьютера; интерпретировать результаты, получаемые в ходе моделирования реальных процессов; оценивать числовые параметры моделируемых объектов и процессов;

- понимать основные принципы устройства и функционирования современных стационарных и мобильных компьютеров; выбирать конфигурацию компьютера в соответствии с решаемыми задачами;

- понимать назначение, а также основные принципы устройства и работы современных операционных систем; знать виды и назначение системного программного обеспечения;

- владеть принципами организации иерархических файловых систем и именования файлов; использовать шаблоны для описания группы файлов;
- использовать на практике общие правила проведения исследовательского проекта (постановка задачи, выбор методов исследования, подготовка исходных данных, проведение исследования, формулировка выводов, подготовка отчета); планировать и выполнять небольшие исследовательские проекты;
- использовать динамические (электронные) таблицы, в том числе формулы с использованием абсолютной, относительной и смешанной адресации, выделение диапазона таблицы и упорядочивание (сортировку) его элементов; построение графиков и диаграмм;
- владеть основными сведениями о табличных (реляционных) базах данных, их структуре, средствах создания и работы, в том числе выполнять отбор строк таблицы, удовлетворяющих определенному условию; описывать базы данных и средства доступа к ним; наполнять разработанную базу данных;
- использовать компьютерные сети для обмена данными при решении прикладных задач;
- организовывать на базовом уровне сетевое взаимодействие (настраивать работу протоколов сети TCP/IP и определять маску сети);
- понимать структуру доменных имен; принципы IP-адресации узлов сети;
- представлять общие принципы разработки и функционирования интернет-приложений (сайты, блоги и др.);
- применять на практике принципы обеспечения информационной безопасности, способы и средства обеспечения надежного функционирования средств ИКТ; соблюдать при работе в сети нормы информационной этики и права (в том числе авторские права);
- проектировать собственное автоматизированное место; следовать основам безопасной и экономичной работы с компьютерами и мобильными устройствами; соблюдать санитарно-гигиенические требования при работе за персональным компьютером в соответствии с нормами действующих СанПиН.

Выпускник на **профильном уровне** получит возможность научиться:

- *применять коды, исправляющие ошибки, возникшие при передаче информации; определять пропускную способность и помехозащищенность канала связи, искажение информации при передаче по каналам связи, а также использовать алгоритмы сжатия данных (алгоритм LZW и др.);*
- *использовать графы, деревья, списки при описании объектов и процессов окружающего мира; использовать префиксные деревья и другие виды деревьев при решении алгоритмических задач, в том числе при анализе кодов;*
- *использовать знания о методе «разделяй и властвуй»;*
- *приводить примеры различных алгоритмов решения одной задачи, которые имеют различную сложность; использовать понятие рекурсивного алгоритма;*
- *использовать понятие универсального алгоритма и приводить примеры алгоритмически неразрешимых проблем;*
- *использовать второй язык программирования; сравнивать преимущества и недостатки двух языков программирования;*
- *создавать программы для учебных или проектных задач средней сложности;*

- использовать информационно-коммуникационные технологии при моделировании и анализе процессов и явлений в соответствии с выбранным профилем;*
- осознанно подходить к выбору ИКТ-средств и программного обеспечения для решения задач, возникающих в ходе учебы и вне ее, для своих учебных и иных целей;*
- проводить (в несложных случаях) верификацию (проверку надежности и согласованности) исходных данных и валидацию (проверку достоверности) результатов натурных и компьютерных экспериментов;*
- использовать пакеты программ и сервисы обработки и представления данных, в том числе – статистической обработки;*
- использовать методы машинного обучения при анализе данных; использовать представление о проблеме хранения и обработки больших данных;*
- создавать многотабличные базы данных; работе с базами данных и справочными системами с помощью веб-интерфейса.*

## Содержание учебного предмета

### 10 класс

#### Раздел 1. Алгоритмы и структуры данных. Часть 1.

##### Тема 1.1. Арифметические алгоритмы.

Быстрое возведение в степень, разложение на делители, простые делители, НОД и НОК, решето Эратосфена.

##### Тема 1.2. Длинная арифметика и эффективные арифметические алгоритмы.

Реализация длинной арифметики, алгоритм Карацубы.

##### Тема 1.3. Динамическое программирование.

Классические задачи динамического программирования: кузнечик, черепашка, наибольшая возрастающая подпоследовательность, наибольшая общая подпоследовательность, одномерная и двумерная динамика.

##### Тема 1.4. Динамическое программирование по подотрезкам.

Подотрезки, задача сжатия повторяющихся подряд идущих подстрок, примеры задач, решаемых динамическим программированием по подотрезкам.

##### Тема 1.5. Стеки, очереди, деки и их использование.

Правильная скобочная последовательность, минимум в окне, обратная польская запись.

##### Тема 1.6. STL (Стандартная библиотека шаблонов).

Алгоритмы и структуры данных стандартной библиотеки шаблонов. Итераторы.

##### Тема 1.7. Сканирующая прямая.

Решение задач методом сортировки событий и сканирующей прямой, одномерный и двумерный случаи.

##### Тема 1.8. Обход графа в глубину.

Способы задания и хранения графа. Обход в глубину. Связность, компоненты связности и 2-раскрашиваемость.

##### Тема 1.9. Применения обхода в глубину.

Примеры задач, решаемых обходом в глубину. Дерево обхода в глубину, классификация ребер.

##### Тема 1.10. Обход в ширину и его применения.

Понятие топологической сортировки ациклического ориентированного графа, задачи, решаемые с помощью топологической сортировки. Лексикографически минимальная топологическая сортировка.

##### Тема 1.11. Циклы, компоненты сильной связности, конденсация.

Поиск и восстановление циклов в графах. Определение и алгоритмы поиска компонент сильной связности, конденсация графа.

##### Тема 1.12. Циклы, компоненты сильной связности, мосты и точки сочленения.

Время входа и выхода из вершины. Поиск мостов и точек сочленения. Реберная и вершинная двусвязность.

## **Раздел 2. Алгоритмы и структуры данных. Часть 2.**

### **Тема 2.1. Обход графа в ширину.**

Алгоритм обхода в ширину для решения задачи поиска кратчайшего пути в невзвешенном графе. Восстановление пути.

### **Тема 2.2. Кратчайшие пути в невзвешенном графе.**

Примеры задач, использующих обход в ширину. Несколько начальных и конечных вершин. Кратчайшие пути в 0-1 графе и 0-k графе.

### **Тема 2.3. Префиксные суммы.**

Префиксные суммы и их применения. Подсчет числа вхождений элемента. Префиксные произведения. Сумма на прямоугольнике.

### **Тема 2.4. Дерево отрезков.**

Понятие дерева отрезков. Реализация сверху и снизу. Решение задачи поиска максимума и суммы на отрезке. Интервальные модификации. Отложенные операции.

### **Тема 2.5. Хеши.**

Хеширование и хеш-функции. Использование хешей для быстрого поиска подстрок.

## **11 класс**

## **Раздел 2. Алгоритмы и структуры данных. Часть 2 (продолжение).**

### **Тема 2.6. Корневая декомпозиция.**

Понятие корневой декомпозиции. Структура данных для решения задач RMQ и RSQ. Алгоритм Mo.

### **Тема 2.7. Динамическое программирование по подмножествам.**

Реализация динамического программирования по подмножествам. Хранение подмножества. Эффективное решение задачи коммивояжера.

### **Тема 2.8. Динамическое программирование по поддеревьям.**

Понятие дерева и поддерева. Подсчет функций на поддеревьях. Размеры и глубины поддеревьев.

### **Тема 2.9. Кратчайшие пути в графе: алгоритм Дейкстры.**

Сложность и корректность алгоритма Дейкстры. Различные варианты реализации алгоритма Дейкстры. Примеры решения задач с помощью модифицированного алгоритма Дейкстры.

### **Тема 2.10. Кратчайшие пути в графе: алгоритм Флойда.**

Сложность и корректность алгоритма Флойда. Отрицательные ребра и циклы отрицательного веса. Восстановление пути. Транзитивное замыкание.

### **Тема 2.11. Кратчайшие пути в графе: алгоритм Форда – Беллмана.**

Сложность и корректность алгоритма Форда – Беллмана. Отрицательные ребра и циклы отрицательного веса. Оптимизация алгоритма.



**Тема 2.12. Паросочетания.**

Двудольные графы и 2-раскрасшиваемость. Определение максимального паросочетания. Алгоритм Куна.

**Тема 2.13. Потоки.**

Потоки в графах. Исток и сток, пропускная способность ребер. Метод Форда – Фалкерсона. Остаточные сети.

**Тема 2.14. Минимальные остовные деревья в графе.**

Минимальное остовное дерево. Алгоритм Прима. Алгоритм Борувки. Минимизация максимального ребра.

**Тема 2.15. Система непересекающихся множеств.**

Система непересекающихся множеств. Стягивающиеся корневые деревья. Алгоритм Крускала для поиска минимального остовного дерева.

**Тема 2.16. Оптимизация перебора.**

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 2.17. Метод «разделяй и властвуй» для решения задач.**

Метод «разделяй и властвуй» и примеры его применения. Сортировка слиянием. Основная теорема.

**Раздел 3. Подготовка к ЕГЭ по информатике.****Тема 3.1. Решение задач 1 – 9.****Тема 3.2. Решение задач 10 – 15.****Тема 3.3. Решение задач 16 – 21.****Тема 3.4. Решение задач 22 – 24.****Тема 3.5. Решение задач 25 – 27.**

## Тематическое планирование учебного предмета

Раздел / тема	Количество ак. ч.
<b>10 класс</b>	
<b>Раздел 1. Алгоритмы и структуры данных. Часть 1</b>	<b>96</b>
Тема 1.1. Арифметические алгоритмы	8
Тема 1.2. Длинная арифметика и эффективные арифметические алгоритмы	8
Тема 1.3. Динамическое программирование	8
Тема 1.4. Динамическое программирование по подотрезкам	8
Тема 1.5. Стеки, очереди, деки и их использование	8
Тема 1.6. STL (Стандартная библиотека шаблонов)	8
Тема 1.7. Сканирующая прямая	8
Тема 1.8. Обход графа в глубину	8
Тема 1.9. Применения обхода в глубину	8
Тема 1.10. Топологическая сортировка графа	8
Тема 1.11. Циклы, компоненты сильной связности, конденсация	8
Тема 1.12. Мосты и точки сочленения	8
<b>Раздел 2. Алгоритмы и структуры данных. Часть 2</b>	<b>40</b>
Тема 2.1. Обход графа в ширину	8
Тема 2.2. Кратчайшие пути в невзвешенном графе	8
Тема 2.3. Префиксные суммы	8
Тема 2.4. Дерево отрезков	8
Тема 2.5. Хеши	8
<b>ИТОГО</b>	<b>136</b>
<b>11 класс</b>	
<b>Раздел 2. Алгоритмы и структуры данных. Часть 2 (продолжение)</b>	<b>96</b>
Тема 2.6. Корневая декомпозиция	8
Тема 2.7. Динамическое программирование по подмножествам	8
Тема 2.8. Динамическое программирование по поддеревьям	8
Тема 2.9. Кратчайшие пути в графе: алгоритм Дейкстры	8
Тема 2.10. Кратчайшие пути в графе: алгоритм Флойда	8
Тема 2.11. Кратчайшие пути в графе: алгоритм Форда – Беллмана	8
Тема 2.12. Паросочетания	8
Тема 2.13. Потoki	8
Тема 2.14. Минимальные остовные деревья в графе	8
Тема 2.15. Система непересекающихся множеств	8
Тема 2.16. Оптимизация перебора	8
Тема 2.17. Метод «разделяй и властвуй» для решения задач	8
<b>Раздел 3. Подготовка к ЕГЭ по информатике</b>	<b>40</b>
Тема 3.1. Решение задач 1 – 9	8
Тема 3.2. Решение задач 10 – 15	8
Тема 3.3. Решение задач 16 – 21	8
Тема 3.4. Решение задач 22 – 24	8

<b>Тема 3.5. Решение задач 25 – 27</b>	8
<b>ИТОГО</b>	<b>136</b>
<b>ВСЕГО</b>	<b>272</b>