

**УТВЕРЖДАЮ**

Директор МБОУ «Хибинская гимназия»

Новикова Л.А. \_\_\_\_\_

Приказ от 31 августа 2023 г. № 34

**М.П.**

# **ПРОГРАММА**

## **Элективного курса**

---

**«Программирование»**

**для 10-11 класса**

Программу разработал:

Зима Д.А.

Учитель информатики

**МБОУ «Хибинская гимназия»**

**Мурманская область, г. Кировск**

## 1. Пояснительная записка

Элективный курс «Программирование на Си» для учащихся 10 - 11 классов информационно-технологического профиля МБОУ «Лицей № 1» разработан на основе учебного пособия для общеобразовательных организаций К.Ю. Полякова «Программирование. Python. C++» в 4-х частях. Включает 138 часов аудиторных занятий и (при возможности) самостоятельную работу учащихся, из них 70 часов в 10 классе и 68 часов в 11 классе.

Курс может быть использован для профильной подготовки учащихся в классах физико-математического, информационно-технологического профилей и рассчитан на 2 часа в неделю сроком реализации - 2 года. Предполагается, что учащиеся предварительно изучили программирование на языке Паскаль, имеют навыки основ алгоритмирования.

Электронный материал авторских разработок доктора технических наук К.Ю. Полякова для сопровождения курса представлен в полном объёме на сайте автора <http://kpolyakov.spb.ru/school/c.htm> и включает презентации, решение задач и сборник программ в среде [Dev-C++](#).

Предметом изучения являются принципы и методы программирования в среде бесплатно распространяемой оболочки [Dev-C++](#) с открытым исходным кодом, включающая компилятор [GCC](#).

Сотрудник компании Bell Labs Деннис Ритчи в начале 1970-х годов разработал один из самых востребованных языков программирования и остаётся на лидирующих позициях до настоящего времени - язык C.

Независимый от процессора язык C работает напрямую с аппаратурой, заменяя чрезвычайно сложное программирование в машинных кодах и на языке ассемблера.

Для моделирования сложных систем, в начале 1980-х годов Бьёрн Страуструп, сотрудник той же компании Bell Labs, дополнил язык C новыми возможностями и создал язык C++, который стал одним из самых популярных языков программирования. Все программы на «чистом» C будут работать и в C++.

Одно из достоинств современного языка C++ — обширная стандартная библиотека: набор готовых подпрограмм для решения часто встречающихся задач (например, для обработки символьных строк).

Целесообразность изучения данного курса определяется тем, что язык C++ – основной язык для создания игр и операционных систем. Программы на C++ работают значительно быстрее, чем на других языках, например Python, предназначенный более для решения небольших задач, в которых скорость выполнения не очень важна и применяемый для разработки сайтов, решения задач биоинформатики и обработки больших данных. Но программы, созданные на Си сложнее писать и отлаживать. «... Умение программировать на разных языках – это обязательное требование к разработчику программного обеспечения»[1]

### Цели курса:

- ознакомление учащихся с принципами и методами современного объектно – ориентированного программирования высокого уровня;
- формирование знаний, умений и навыков в области программирования на языке C++;
- развитие алгоритмических способностей учащихся по программированию.

### Задачи курса: научить учащихся

- создавать программы для решения прикладных задач разного уровня в среде Dev-C++ с открытым исходным кодом, включающая компилятор GCC;

- понимать основные принципы объектно-ориентированного программирования.

Данный курс имеет выраженную практическую направленность, которая и определяет логику построения материала учебных занятий. Основной формой обучения является практикум. Для работы необходим персональный компьютер (один на каждого ученика) и среда Dev-C++ с открытым исходным кодом, включающая компилятор GCC. Для выполнения большинства заданий достаточно использовать среду, бесплатно загруженную с Web-сайта <http://www.bloodshed.net/dev/devcpp.html>

Для поддержки курса К.Ю. Поляковым разработано электронное учебное пособие в формате PDF, которое содержит теоретический материал, задания для выполнения практических работ, презентации, которое используется во время уроков для самостоятельной работы и в качестве справочника. Это позволяет успешно организовывать занятия в группах, в которых есть ученики с разным темпом усвоения материала.

Знания, полученные при изучении элективного курса «Программирование на Си++», учащиеся могут применить для решения прикладных задач разного рода, повышения качества подготовки к успешной сдаче ЕГЭ. Полученные знания и умения являются основой для последующего изучения программирования в высших профессиональных образовательных учреждениях.

## **2. Требования к результатам освоения обучающимися программы элективного курса «Программирование на Си++»**

В рамках данного курса учащиеся получают следующие знания и умения:

### **Уметь**

- самостоятельно определять цели деятельности и составлять планы деятельности; самостоятельно осуществлять, контролировать и корректировать деятельность; использовать все возможные ресурсы для достижения поставленных целей и реализации планов деятельности; выбирать успешные стратегии в различных ситуациях;
- продуктивно общаться и взаимодействовать в процессе совместной деятельности, учитывать позиции других участников деятельности, эффективно разрешать конфликты;
- владеть навыками познавательной, учебно-исследовательской и проектной деятельности, навыками разрешения проблем; осуществлять самостоятельный поиск методов решения практических задач;
- вести самостоятельную информационно-познавательную деятельность, включая ориентирование в различных источниках информации, критически оценивать и интерпретировать информацию, получаемую из различных источников;
- применять навыки алгоритмического мышления и понимать необходимость применения формального описания алгоритмов;
- использовать понятие сложности алгоритма, применять основные алгоритмы обработки числовой и текстовой информации, алгоритмы поиска и сортировки;
- владеть приёмами написания программы в среде Dev-C++ для решения прикладных задач с использованием основных конструкций программирования; осуществлять отладку таких программ; тестировать и оптимизировать готовые прикладные компьютерные программы;
- применять базовые типы данных и структуры данных; использовать основные управляющие конструкции;
- осуществлять формализацию задач и производить документирование программ.

### **Знать**

- алгоритмические конструкции, лежащие в основе программирования в среде Dev-C++ с открытым исходным кодом, включающей компилятор GCC;

- синтаксис операторов и их применение в решении задач;
- основные понятия объектно-ориентированного программирования (объект, свойство, метод, наследование).

### **Формы подведения итогов**

Текущий контроль уровня усвоения материала осуществляется по результатам выполнения учащимися практических заданий. В конце разделов курса каждый учащийся выполняет индивидуальный проект в качестве зачетной работы.

## **3.Содержание программы курса**

### *10 класс*

#### **Программирование на языке C++ (часть 1, глава 2).**

#### **Основные управляющие конструкции.– 35 ч.**

##### *Тема 1.1. Первые программы.*

Повторение: понятие алгоритма, свойства алгоритма, исполнитель алгоритма (виды, основные характеристики), способы записи алгоритма.

Структура программы. Операторы ввода-вывода. Переменные.

Ключевые слова: компилятор, препроцессор, отдельная компиляция, вывод данных, поток вывода, поток ввода, пространство имен, символьная строка.

##### *Тема 1.2. Диалоговые программы.*

Диалоговые программы. Переменные и их типы. Сумма чисел.

Ключевые слова: ввод данных, переменная, объявление переменной, входной поток.

##### *Тема 1.3. Компьютерная графика.*

Библиотека TX Library. Управляем пикселями. Линии и фигуры. Замкнутые фигуры.

Ключевые слова: окно, координаты, оси координат, пиксель, цвет контура, цвет заливки, код цвета, прозрачный цвет.

##### *Тема 1.4. Процедуры.*

Длинная программа. Рефакторинг. Процедуры с параметрами.

Ключевые слова: подпрограмма, процедура, рефакторинг, аргументы, параметры, базовая точка.

##### *Тема 1.5. Обработка целых чисел.*

Предельные значения чисел. Различие между вещественными и целыми числами.

Дискретное представление чисел. Программное повышение точности вычислений.

Хранение в памяти целых чисел: со знаком, без знака. Повторение: получение дополнительного кода числа.

Арифметические операции с целыми числами: сложение и вычитание, умножение и деление. Операция сравнения чисел. Повторение: поразрядные логические операции, сдвиги.

Ключевые слова: арифметические выражения, частное, остаток, форматный вывод, случайные числа, зерно.

##### *Тема 1.6. Обработка вещественных чисел.*

Повторение: хранение в памяти вещественных чисел. Нормализованное представление вещественных чисел в компьютере.

Арифметические операции с вещественными числами: сложение и вычитание, умножение и деление.

Ключевые слова: вещественное число, научный формат, мантисса, форматный вывод, округление.

##### *Тема 1.7. Ветвления.*

Условный оператор. Полная и неполная формы записи условного оператора. Сложные условия в условном операторе и их применение в написании программ. Множественный выбор - переключатель switch.

Ключевые слова: условный оператор, полная форма, неполная форма, составной оператор, вложенный условный оператор, логические переменные.

*Тема 1.8. Циклы.*

Понятие циклического алгоритма. Циклы с предусловием. Циклы с постусловием. Циклы с переменными (счетчиком). Вложенные циклы.

Ключевые слова: цикл с предусловием, цикл с постусловием, цикл по переменной.

*Тема 1.9. Анимация*

Принципы анимации. Рисуем шарик. Начальное положение. Анимация движения. Обработка нажатия клавиши. Ключевые слова: анимация, процедура, пауза, нажатие клавиши.

*Учащиеся должны знать:*

- понятие и свойства алгоритма, способы записи, типы алгоритмических структур;
- структуру программы, операторы ввода-вывода, переменные;
- типы переменных, стандартные функции;
- условный оператор, циклы;
- принципы анимации.

*Учащиеся должны уметь:*

- применять полученные знания к решению задач;
- применять операторы ввода-вывода в программе при создании диалога с пользователем в режиме компиляции;
- соблюдать структуру программы;
- применять стандартные функции к решению задач;
- владеть приемами применения типов переменных;
- составлять программы с использованием конструкций ветвления и циклов;
- применять принципы анимации при создании анимированных объектов.

*Предметные результаты*

- 1) Владение универсальным языком программирования высокого уровня (по выбору), представлениями о базовых типах данных и структурах данных; умением использовать основные управляющие конструкции.
- 2) Владение умением понимать программы, написанные на выбранном для изучения универсальном алгоритмическом языке высокого уровня; знанием основных конструкций программирования; умением анализировать алгоритмы с использованием таблиц.
- 3) Владение навыками алгоритмического мышления и понимание необходимости формального описания алгоритмов.
- 4) Овладение понятием сложности алгоритма, знание основных алгоритмов обработки числовой и текстовой информации.

## **Программирование на языке C++ (часть 2, глава 2)**

**Процедуры и функции — элементы структуризации программ.**

**Массив — фундаментальная структура данных. — 35 ч.**

*Тема 2.1. Процедуры.*

Вспомогательные алгоритмы. Процедуры. Запись процедуры на языке программирования. Процедура с параметрами.

Ключевые слова: процедура, интерфейс, реализация, параметр, локальная переменная, глобальная переменная, область видимости, передача по ссылке.

#### *Тема 2.2. Рекурсия.*

Рекурсия. Примеры рекурсий в математике и литературе. Рекурсивная процедура (функция). Использование рекурсий при написании программ.

Ключевые слова: рекурсивная процедура, фрактал, базовые объекты, условия останова, анимация

#### *Тема 2.3. Функции.*

Функция. Виды функций. Запись функций на языке программирования.

Ключевые слова: функция, параметры, вызов функции, результат функции, рекурсивная функция.

#### *Тема 2.4. Символьные строки.*

Символьная строка. Операции со строками: сцепление, удаление, копирование элементов. Функции поиска подстроки. Преобразование из строки в число и наоборот. Применение строковых данных в процедурах и функциях. Рекурсивный перебор символов. Сравнение и сортировка строк.

Ключевые слова: символьная строка, длина строки, сцепление строк, выход за границы строки, подстрока, удаление символов, вставка символов, поиск подстроки, замена подстроки, преобразование типов.

#### *Тема 2.5. Массивы.*

Массивы в C++. Обращение к элементу массива. Перебор элементов массива. Вывод и ввод массива. Заполнение массива случайными числами. Алгоритмы обработки массивов.

Ключевые слова: массив, индекс элемента, значение элемента, константа, заполнение массива, вывод массива, ввод массива.

#### *Тема 2.6. Используем массивы.*

Используем массивы. Игра «Стрельба по тарелкам».

Ключевые слова: массив, константа, инициализация, глобальные переменные, обработчик события.

#### *Тема 2.7. Матрицы.*

Матрица. Основные операции с матрицами: объявление, заполнение, вывод на экран. Обработка элементов матрицы.

Ключевые слова: матрица, строка, столбец, перебор элементов, вложенный цикл, квадратная матрица, главная диагональ, побочная диагональ, перестановка строк.

#### *Тема 2.8. Системы управления версиями. Основные приемы работы с Git.*

Системы управления версиями. Основные приемы работы с Git. Операции с файлами. Работа с удалённым архивом.

#### *Учащиеся должны знать:*

- применение вспомогательных алгоритмов: процедур и функций.
- применений рекурсивных алгоритмов к решению задач;
- понятие массива; основные операции с массивами: объявление, заполнение, вывод на экран;
- методы сортировки массивов: метод пузырька, метод выбора, быстрая сортировка;
- понятие символьной строки и операции со строками: объединение, удаление, копирование элементов строк;
- понятие матрицы и основные операции с матрицами: объявление, заполнение, вывод на экран, обработка элементов матрицы.

#### *Учащиеся должны уметь:*

- применять полученные знания к решению задач;

- осуществлять поиск и обработку элементов в массиве, определять минимальный и максимальный элементы массива, осуществлять реверс массива, сдвиг элементов массива, отбор элементов массива по условиям, сортировку в массивах, двоичный поиск в массиве;
- осуществлять операции со строками: объединение, удаление, копирование элементов, функции поиска подстроки, преобразование из строки в число и наоборот;
- применять строковые данные в процедурах и функциях;
- осуществлять рекурсивный перебор символов, сравнивать и сортировать строки;
- производить основные операции с матрицами: объявление, заполнение, вывод на экран, обработку элементов матрицы.

#### *Предметные результаты*

1) Владение универсальным языком программирования высокого уровня (по выбору), представлениями о базовых типах данных и структурах данных; умением использовать основные управляющие конструкции.

2) Владение умением понимать программы, написанные на выбранном для изучения универсальном алгоритмическом языке высокого уровня; знанием основных конструкций программирования; умением анализировать алгоритмы с использованием таблиц.

3) Владение навыками алгоритмического мышления и понимание необходимости формального описания алгоритмов.

4) Овладение понятием сложности алгоритма, знание основных алгоритмов обработки числовой и текстовой информации, алгоритмов поиска и сортировки.

## **11 класс**

### **Программирование на языке С++ (часть 3, глава 2).**

#### **Динамические структуры данных. – 35 ч.**

##### *Тема 3.1. Простые алгоритмы сортировки.*

Сортировка в массивах. Методы сортировки: метод пузырька, метод выбора, сортировка вставками. Массивы в подпрограммах.

Ключевые слова: сортировка, метод пузырька, сортировка вставками, перестановка элементов, сложность алгоритма.

##### *Тема 3.2. Быстрые алгоритмы сортировки и поиска.*

Сортировка в массивах. Методы сортировки: сортировка слиянием, быстрая сортировка. Стандартная сортировка на языке С++. Двоичный поиск.

Ключевые слова: сортировка слиянием, быстрая сортировка, двоичный поиск, лямбда-функция

##### *Тема 3.3. Обработка файлов.*

Понятие файла. Типы файлов. Этапы работы с файлами: открытие файла, запись в файл, удаление записей из файла, чтение из файла, закрытие файла. Обработка массивов, записанных в файле. Обработка строк, записанных в файле. Обработка смешанных данных, записанных в файле.

Ключевые слова: файл, файловый поток, открытие файла, закрытие файла, чтение из файла, запись в файл, конец файла, аргументы командной строки.

##### *Тема 3.4. Целочисленные алгоритмы.*

Целочисленные алгоритмы. Решето Эратосфена. Длинные числа.

Ключевые слова: решето Эратосфена, длинные числа, разряды, перенос.

##### *Тема 3.5. Динамические массивы и словари.*

Понятие динамического массива. Объявление и заполнение динамического массива. Использование динамических массивов в подпрограммах. Расширение массива.

Принципы работы динамического массива. Тип `vector` из библиотеки STL. Итераторы. Словари. Перебор элементов словаря.

Ключевые слова: динамический массив, указатель, контейнер, вектор, словарь, ключ, значение, перебор элементов, сортировка по ключу, итератор.

*Тема 3.6. Структуры.*

Понятие структуры. Объявление структур. Обращение к полю структуры. Работа с файлами. Сортировка.

Ключевые слова: структура, поле, точечная запись, сортировка, ключ.

*Тема 3.7. Стеки, очереди, деки*

Понятие стека. Использование динамического массива для создания стека. Вычисление арифметических выражений. Скобочные выражения. Очередь и деки как разновидности линейного списка.

Ключевые слова: стек, очередь, дек.

*Тема 3.8. Деревья.*

Дерево. Двоичное дерево. Дерево поиска. Обход двоичного дерева. Вычисление арифметических выражений. Использование связанных структур. Хранение двоичного дерева в массиве.

Ключевые слова: дерево, двоичное дерево, обход в глубину, обход в ширину, дерево поиска, ключ, хранение в массиве, модуль, проект, заголовочный файл.

*Тема 3.9. Графы*

Понятие графа. «Жадные» алгоритмы. Алгоритм Прима-Крускала. Поиск кратчайших путей в графе. Алгоритм Дейкстры. Алгоритм Флойда-Уоршелла.

Ключевые слова: жадный алгоритм, остовное дерево, задача коммивояжера, Гамильтонов цикл, полный перебор, рекурсия, хвостовая рекурсия, случайный поиск, глобальные переменные, структуры.

*Тема 3.10. Динамическое программирование.*

Динамическое программирование. Поиск оптимального решения. Количество решений.

Ключевые слова: динамическое программирование, перебор вариантов, мемоизация, оптимальная программа, редактирование строк, расстояние Левенштейна.

*Учащиеся должны знать:*

- понятие файла, типы, этапы работы;
- алгоритмы работы с файлами;
- алгоритм поиска простых чисел с помощью «решета Эратосфена»;
- понятие «длинного числа», принципы хранения и выполнения операций с «длинными» числами;
- понятие структуры (записи, множества), основные операции со структурами;
- понятие «дерево» и области применения этой структуры данных;
- понятия «граф», «узел», «ребро»;
- простые алгоритмы на графах;
- принципы динамического программирования.

*Учащиеся должны уметь:*

- осуществлять обработку массивов строк, смешанных данных, записанных в файле;
- использовать текстовый и типизированные файлы;
- программировать простые операции с «длинными» числами;
- использовать различные структуры, грамотно выбирать структуру для конкретной задачи;
- программировать простые алгоритмы на графах;
- программировать алгоритмы, использующие динамическое программирование.



### *Предметные результаты*

- 1) Владение универсальным языком программирования высокого уровня (по выбору), представлениями о базовых типах данных и структурах данных; умением использовать основные управляющие конструкции.
- 2) Владение умением понимать программы, написанные на выбранном для изучения универсальном алгоритмическом языке высокого уровня; знанием основных конструкций программирования; умением анализировать алгоритмы с использованием таблиц.
- 3) Владение навыками алгоритмического мышления и понимание необходимости формального описания алгоритмов.
- 4) Овладение понятием сложности алгоритма, знание основных алгоритмов обработки числовой и текстовой информации, алгоритмов поиска и сортировки.

Динамические структуры данных. Связанный список. Создание элемента списка. Добавление узла. Проход по списку. Поиск узла в списке. Алфавитно-частотный словарь. Удаление узла. Барьеры. Двусвязный список. Операции с двусвязным списком. Циклические списки.

Стек. Реализация стека с помощью массива. Реализация стека с помощью списка. Системный стек в программах. Очередь. Реализация очереди с помощью массива. Реализация очереди с помощью списка. Дек.

Что такое деревья? Реализация двоичных деревьев в языке Си. Поиск с помощью дерева. Разбор арифметического выражения. Дерево игр.

Основные понятия. Задача Прима-Крускала. Кратчайший путь. Оптимальное размещение. Задача коммивояжера. Задача о паросочетаниях.

## **Программирование на языке C++ (часть 4, глава 2). Объектно-ориентированное программирование -33 ч.**

### *Тема 4.1. Классы и объекты в C++.*

Классы и объекты в C++: новая задача и ее анализ. Класс CMap. Пишем свой конструктор. Рефакторинг. Рисуем карты.

Ключевые слова: класс, объект, экземпляр, модификатор доступа, конструктор, инициализация, метод, рефакторинг.

### *Тема 4.2. Программа с классами (практикум).*

Класс CCar. Рефакторинг. Основная программа. Разбиение на модули. Основная программа. Разбиение на модули.

Ключевые слова: класс, объект, ссылка, конструктор, метод, рефакторинг.

### *Тема 4.3. Инкапсуляция.*

Объект защищает свои данные. Изменение внутреннего устройства. Свойство «только для чтения». Свойства в C#.

Ключевые слова: инкапсуляция, поле, метод, модификатор доступа, интерфейс, свойство.

### *Тема 4.4. Наследование.*

Моделирование жизни в океане. Иерархия классов. Базовый класс. Абстрактный класс. Защищенные поля и методы (protected). Неподвижные объекты. Подвижные объекты. Рыбы. Хищники. Вспомогательные процедуры и функции. Основная программа.

Ключевые слова: класс, иерархия классов, базовый класс, наследование, абстрактный класс, виртуальный метод, защищенные поля, утечка памяти.

### *Тема 4.5. Полиморфизм.*

Полиморфизм в действии. Класс COcean. Деструктор.

Ключевые слова: базовый класс, класс-наследник, виртуальный метод, полиморфизм, позднее связывание, таблица виртуальных методов, деструктор.

### *Тема 4.6. Взаимодействие объектов.*

Столкновения. Изменение базового класса. Изменение других классов. Умные указатели.

Ключевые слова: столкновение, «мертвые» объекты, итератор, преобразование типов.

*Тема 4.7. Простая программа на C#.*

RAD – среды для разработки программ. Язык C# и среда .NET. Проект C#. Свойства объектов. Обработчики событий.

Ключевые слова: RAD – среда, среда NET., язык C#, виртуальная машина, проект, форма, свойство, событие, обработчик события, компонент, статистический класс.

*Тема 4.8. Использование компонентов.*

Программы для просмотра рисунков. Ввод и вывод данных. Обработка ошибок.

Ключевые слова: компонент, свойство, событие, обработчик события, родительский объект, дочерний объект, поле ввода, метка, статический метод, обработка ошибок, исключение.

*Тема 4.9. Создание новых классов.*

Разработка программы для вычисления арифметических выражений: модель, новый класс.

Методы класса. Вычисление арифметических выражений: представление. Добавляем новый компонент. Изменение поведения компонента.

Ключевые слова: модуль, модель, представление, класс, компонент, статический класс, методы класса, свойство.

*Учащиеся должны знать:*

- понятия «объект», «свойства объектов»;
- понятие «классы», «иерархия классов их классификацию»;
- понятие «базовый класс, классы-наследники, модульность»;
- принципы работы в RAD-средах;
- сообщения между объектами.

*Учащиеся должны уметь:*

- составлять простые программы в RAD-средах;
- оценивать вычислительную сложность изученных алгоритмов.

*Предметные результаты*

*1) Владение стандартными приёмами написания на алгоритмическом языке программы для решения стандартной задачи с использованием основных конструкций программирования и отладки таких программ; использование готовых прикладных компьютерных программ по выбранной специализации.*

Понятие объектно-ориентированного программирования. Объекты. Классы.

Создание объектов в программе.

Скрытие внутреннего устройства.

Иерархия классов. Классификация. Иерархия логических элементов. Базовый класс. Классы-наследники. Модульность. Сообщения между объектами.

Программы с графическим интерфейсом. RAD-среды для разработки программ.

Основы программирования в RAD-средах. Общий подход. Создание простейшей программы. Свойства объектов. Обработчики событий.

Программа с компонентами. Ввод и вывод данных. Обработка ошибок.

Модификация компонентов.

Разработка программы вычисления арифметического выражения, записанного в символьной строке.

**4. Тематическое планирование с указанием количества часов на освоение каждой темы по 2 часа в неделю в 10 и 11 классах (всего 138 часов)**

№ п/п	Наименование разделов и тем	Количество часов	Виды занятий	
			Теория	Практика
<b>10 класс</b>				
<b>1.</b>	<b>Программирование на языке С++ (часть 1, глава 2) Основные управляющие конструкции.</b>	<b>35</b>	<b>9</b>	<b>26</b>
1.1	Первые программы.	2	1	1
1.2	Диалоговые программы.	3	1	2
1.3	Компьютерная графика.	2	1	1
1.4	Процедуры.	3	1	2
1.5	Обработка целых чисел.	4	1	3
1.6	Обработка вещественных чисел.	4	1	3
1.7	Ветвления.	7	1	6
1.8	Циклы.	6	1	5
1.9	Анимация.	4	1	3
<b>2.</b>	<b>Программирование на языке С++ (часть 2, глава 2) Процедуры и функции — элементы структуризации программ. Массив — фундаментальная структура данных.</b>	<b>35</b>	<b>10</b>	<b>25</b>
2.1	Процедуры.	4	1	3
2.2	Рекурсия.	3	1	2
2.3	Функции.	3	1	2
2.4	Символьные строки.	10	2	8
2.5	Массивы.	2	1	1
2.6	Используем массивы.	3	1	2
2.7	Матрицы.	7	1	6
2.8	Системы управления версиями. Основные приемы работы с Git.	2	1	1
	Резерв	1	1	0
<b>Итого</b>		<b>70</b>	<b>19</b>	<b>51</b>
<b>11 класс</b>				
<b>3.</b>	<b>Программирование на языке С++ (часть 3, глава 2). Динамические структуры данных.</b>	<b>35</b>	<b>8</b>	<b>27</b>
3.1	Простые алгоритмы сортировки.	3	1	2
3.2	Быстрые алгоритмы сортировки и поиска.	4	1	3
3.3	Обработка файлов.	3	1	2
3.4	Целочисленные алгоритмы.	2	0	2

3.5	Динамические массивы и словари.	4	1	3
3.6	Структуры.	3	1	2
3.7	Стеки, очереди, деки.	3	0	3
3.8	Деревья.	5	1	4
3.9	Графы.	5	1	4
3.10	Динамическое программирование.	3	1	2
<b>4.</b>	<b>Программирование на языке C++(часть 4, глава 2). Объектно-ориентированное программирование.</b>	<b>33</b>	<b>3</b>	<b>30</b>
4.1	Классы и объекты в C++.	3	1	2
4.2	Программа с классами (практикум).	3	0	3
4.3	Инкапсуляция.	3	1	2
4.4	Наследование.	6	0	6
4.5	Полиморфизм.	3	0	3
4.6	Взаимодействие объектов.	4	0	4
4.7	Простая программа на C#.	3	0	3
4.8	Использование компонентов.	3	0	3
4.9	Создание новых классов.	4	0	4
	Резерв.	1	1	0
<b>Итого</b>		<b>68</b>	<b>11</b>	<b>57</b>
Всего		138	30	108

## Описание модулей, изучаемых в рамках элективного курса Программирование на С++ в 10-11 классах информационно-технологического профиля.

Используемые сокращения: СР – самостоятельная работа, ПР – практическая работа.

### 10 класс (70 часов)

#### Раздел «Алгоритмы и программирование»

Номер урока	Тема урока	Параграф учебника (номер, название)	Практические работы (номер, название)	Работы компьютерного практикума (источник, номер, название)	Количество часов на тему	Содержание программы
1.	Язык С++. Самая простая программа.	<b>Часть 1. § 14.</b> Первые программы.			<b>1</b>	Повторение: понятие алгоритма, свойства алгоритма, исполнитель алгоритма (виды, основные характеристики), способы записи алгоритма.
2.	Вывод текста на экран.	<b>Часть 1. § 14.</b> Первые программы.	Тест № 1. Оператор вывода.	ПР № 1. Вывод текста на экран.	<b>1</b>	Структура программы. Операторы ввода-вывода. Переменные.  Ключевые слова: компилятор, препроцессор, отдельная компиляция, вывод данных, поток вывода, поток ввода, пространство имен, символьная строка.
3.	Диалоговые программы.	<b>Часть 1. § 15.</b> Диалоговые программы.		ПР № 2. Программа «Как тебя зовут?»	<b>1</b>	Диалоговые программы. Переменные и их типы. Сумма чисел.
4.	Переменные и их типы.	<b>Часть 1. § 15.</b> Диалоговые программы.	Тест № 2. Исправление ошибок в диалоговых программах		<b>1</b>	Ключевые слова: ввод данных, переменная, объявление переменной, входной поток.
5.	Сумма чисел.	<b>Часть 1. § 15.</b> Диалоговые программы.		ПР № 3. Сумма чисел.	<b>1</b>	
6.	Ограниченность значений целых чисел.	<b>Часть 1. § 18.</b> Обработка целых чисел.	Тест № 3. Сокращенная запись операций.		<b>1</b>	Предельные значения чисел. Различие между вещественными и целыми числами. Дискретное представление чисел. Программное повышение
7.	Арифметические выражения.	<b>Часть 1. § 18.</b> Обработка целых чисел.		ПР № 4. Простые вычисления.	<b>1</b>	

Номер урока	Тема урока	Параграф учебника (номер, название)	Практические работы (номер, название)	Работы компьютерного практикума (источник, номер, название)	Количество часов на тему	Содержание программы
8.	Деление и остаток. Вывод данных на экран.	<b>Часть 1. § 18.</b> Обработка целых чисел.	Тест № 4. Операции «/» и «%».	ПР № 5. Применени е операций «/» и «%» к решению задач.	1	точности вычислений.  Хранение в памяти целых чисел: со знаком, без знака. Повторение: получение дополнительного кода числа.
9.	Случайные целые числа.	<b>Часть 1. § 18.</b> Обработка целых чисел.		ПР № 6. Случайное целое число на заданном отрезке [a, b].	1	Арифметические операции с целыми числами: сложение и вычитание, умножение и деление. Операция сравнения чисел. Повторение: поразрядные логические операции, сдвиги. Ключевые слова: арифметические выражения, частное, остаток, форматный вывод, случайные числа, зерно.
10.	Вещественные числа в языке C++.	<b>Часть 1. § 19.</b> Обработка вещественных чисел.	Тест № 5. Типы вещественных чисел.		1	Повторение: хранение в памяти вещественных чисел. Нормализованное представление вещественных чисел в компьютере.
11.	Ввод и вывод.	<b>Часть 1. § 19.</b> Обработка вещественных чисел.		ПР № 7. Формы записи вещественных чисел в программах.	1	Арифметические операции с вещественными числами: сложение и вычитание, умножение и деление.
12.	Операции с вещественными числами.	<b>Часть 1. § 19.</b> Обработка вещественных чисел.		ПР № 8. Операции с вещественными числами.	1	Ключевые слова: вещественное число, научный формат, мантисса, форматный вывод, округление.
13.	Случайные вещественные числа.	<b>Часть 1. § 19.</b> Обработка вещественных чисел.	Тест № 6. Случайные вещественные числа.	ПР № 9. Случайные вещественные числа.	1	
14.	Условный оператор.	<b>Часть 1. § 20.</b> Ветвления.	Тест № 7. Ветвления.	ПР № 10. Ветвления.	1	Условный оператор. Полная и неполная формы записи условного оператора. Сложные условия в условном операторе. Множественный выбор.
15.	Вложенный условный оператор.	<b>Часть 1. § 20.</b> Ветвления.		ПР № 11. Вложенные условные операторы.	1	
16.	Логические переменные.	<b>Часть 1. § 20.</b> Ветвления.	Тест № 8. Логические переменные.	ПР № 12. Логические переменные.	1	

Номер урока	Тема урока	Параграф учебника (номер, название)	Практические работы (номер, название)	Работы компьютерного практикума (источник, номер, название)	Количество часов на тему	Содержание программы
17.	Сложные условия.	<b>Часть 1. § 20.</b> Ветвления.	Тест № 9. Сложные условия.	ПР № 13. Сложные условия.	<b>1</b>	Ключевые слова: условный оператор, полная форма, неполная форма, составной оператор, вложенный условный оператор, логические переменные.
18.	Множественный выбор.	<b>Часть 1. § 20.</b> Ветвления.		ПР № 14. Множественный выбор.	<b>1</b>	
19.	Практикум: использование ветвлений.	<b>Часть 1. § 20.</b> Ветвления.		ПР № 15. Задачи на ветвления.	<b>1</b>	
20.	Контрольная работа «Ветвления».	<b>Часть 1. § 20.</b> Ветвления.			<b>1</b>	
21.	Цикл с предусловием.	<b>Часть 1. § 21.</b> Циклы.	Тест № 10. Циклы с предусловием.	ПР № 16. Циклы с предусловием.	<b>1</b>	Понятие циклического алгоритма. Циклы с предусловием. Циклы с постусловием. Циклы с переменными (счетчиком). Вложенные циклы.  Ключевые слова: цикл с предусловием, цикл с постусловием, цикл по переменной.
22.	Циклы с постусловием.	<b>Часть 1. § 21.</b> Циклы.	Тест № 11. Циклы с постусловием.	ПР № 17. Циклы с постусловием.	<b>1</b>	
23.	Вычисление квадратного корня.	<b>Часть 1. § 21.</b> Циклы.		ПР № 18. Вычисление квадратного корня.	<b>1</b>	
24.	Циклы по переменной.	<b>Часть 1. § 21.</b> Циклы.	Тест № 12. Циклы по переменной.	ПР № 19. Циклы по переменной.	<b>1</b>	
25.	Вложенные циклы.	<b>Часть 1. § 21.</b> Циклы.		ПР № 20. Вложенные циклы.	<b>1</b>	
26.	Контрольная работа «Циклы».				<b>1</b>	
27.	Библиотека TX Library. Управляем пикселями.	<b>Часть 1. § 16.</b> Компьютерная графика.	Тест № 13. Компьютерная графика.		<b>1</b>	
28.	Линии и фигуры. Замкнутые фигуры.	<b>Часть 1. § 16.</b> Компьютерная графика.		ПР № 21. Линии и фигуры. Замкнутые фигуры.	<b>1</b>	Ключевые слова: окно, координаты, оси координат, пиксель, цвет контура, цвет заливки, код цвета, прозрачный цвет.
29.	Длинная программа.	<b>Часть 1. § 17.</b> Процедуры.		ПР № 22. Процедуры	<b>1</b>	Длинная программа. Рефакторинг. Процедуры с

Номер урока	Тема урока	Параграф учебника (номер, название)	Практические работы (номер, название)	Работы компьютерного практикума (источник, номер, название)	Количество часов на тему	Содержание программы
30.	Рефакторинг.	<b>Часть 1. § 17.</b> Процедуры.		ПР № 23. Рефакторинг.	<b>1</b>	параметрами.
31.	Процедуры с параметрами.	<b>Часть 1. § 17.</b> Процедуры.		ПР № 24. Процедуры с параметрами.	<b>1</b>	Ключевые слова: подпрограмма, процедура, рефакторинг, аргументы, параметры, базовая точка.
32.	Принципы анимации. Рисуем шарик. Начальное положение.	<b>Часть 1. § 22.</b> Анимация.		ПР № 25. Рисуем шарик.	<b>1</b>	Принципы анимации. Рисуем шарик. Начальное положение. Анимация движения. Обработка нажатия клавиши. Обработка нажатия клавиши.
33.	Анимация движения.	<b>Часть 1. § 22.</b> Анимация.		ПР № 26. Анимация движения.	<b>1</b>	Ключевые слова: анимация, процедура, пауза, нажатие клавиши.
34.	Обработка нажатия клавиши.	<b>Часть 1. § 22.</b> Анимация.		ПР № 27. Обработка нажатия клавиши.	<b>1</b>	
35.	Зачетная проектная работа «Графика на C++»			ПР № 28. Рисуем и анимируем «С Новым годом!»	<b>1</b>	
36.	Простые процедуры.	<b>Часть 2. § 12.</b> Процедуры.	Тест № 14. Процедуры.		<b>1</b>	Вспомогательные алгоритмы. Процедуры. Запись процедуры на языке программирования. Процедура с параметрами. Ключевые слова: процедура, интерфейс, реализация, параметр, локальная переменная, глобальная переменная, область видимости, передача по ссылке.
37.	Процедуры с параметрами. Несколько параметров.	<b>Часть 2. § 12.</b> Процедуры.		ПР № 29. Процедуры с параметрами.	<b>1</b>	
38.	Локальные и глобальные переменные. Процедуры, изменяющие аргументы.	<b>Часть 2. § 12.</b> Процедуры.	Тест № 15. Локальные и глобальные переменные.	ПР № 30. Процедуры, изменяющие аргументы.	<b>1</b>	
39.	Процедуры в графике.	<b>Часть 2. § 12.</b> Процедуры.		ПР № 31. Процедуры в графике.	<b>1</b>	



Номер урока	Тема урока	Параграф учебника (номер, название)	Практические работы (номер, название)	Работы компьютерного практикума (источник, номер, название)	Количество часов на тему	Содержание программы
40.	Рекурсия.	<b>Часть 2. § 13.</b> Рекурсия.	Тест № 16. Рекурсия.	ПР № 32. Рекурсия.	<b>1</b>	Рекурсия. Примеры рекурсий в математике и литературе. Рекурсивная процедура (функция). Использование рекурсий при написании программ. Ключевые слова: рекурсивная процедура, фрактал, базовые объекты, условия остановки, анимация
41.	Дерево Пифагора.	<b>Часть 2. § 13.</b> Рекурсия.		ПР № 33. Дерево Пифагора.	<b>1</b>	
42.	Анимация.	<b>Часть 2. § 13.</b> Рекурсия.		ПР № 34. Анимация.	<b>1</b>	
43.	Функция. Функции в C++ +. Примеры функций.	<b>Часть 2. § 14.</b> Функции.	Тест № 17. Функция.	ПР № 35. Функции.	<b>1</b>	Функция. Виды функций. Запись функций на языке программирования.  Ключевые слова: функция, параметры, вызов функции, результат функции, рекурсивная функция.
44.	Логические функции.	<b>Часть 2. § 14.</b> Функции.		ПР № 36. Логические функции.	<b>1</b>	
45.	Рекурсивные функции.	<b>Часть 2. § 14.</b> Функции.		ПР № 37. Рекурсивные функции.		
46.	Символьные строки. Что такое символьная строка?	<b>Часть 2. § 15.</b> Символьные строки.	Тест № 18. Символьные строки.	ПР № 38. Рекурсия.	<b>1</b>	Символьная строка. Операции со строками: сцепление, удаление, копирование элементов. Функции поиска подстроки. Преобразование из строки в число и наоборот. Применение строковых данных в процедурах и функциях. Рекурсивный перебор символов. Сравнение т сортировка строк.  Ключевые слова: символьная строка, длина строки, сцепление строк, выход за границы строки, подстрока, удаление символов, вставка символов, поиск подстроки, замена подстроки, преобразование типов.
47.	Сравнение, сцепление строк.	<b>Часть 2. § 15.</b> Символьные строки.		ПР № 39. Сравнение, сцепление строк.	<b>1</b>	
48.	Обращение к символам. Перебор всех символов.	<b>Часть 2. § 15.</b> Символьные строки.		ПР № 40. Обращение к символам.	<b>1</b>	
49.	Подстрока. Удаление и вставка.	<b>Часть 2. § 15.</b> Символьные строки.		ПР № 41. Подстрока. Удаление и вставка.	<b>1</b>	
50.	Поиск в символьных строках. Замена символов.	<b>Часть 2. § 15.</b> Символьные строки.		ПР № 42. Поиск в символьных строках. Замена символов.	<b>1</b>	

Номер урока	Тема урока	Параграф учебника (номер, название)	Практические работы (номер, название)	Работы компьютерного практикума (источник, номер, название)	Количество часов на тему	Содержание программы
51.	Преобразования «строка – число».	<b>Часть 2. § 15.</b> Символьные строки.		ПР № 43. Преобразования «строка – число».	<b>1</b>	
52.	Символьные строки в функциях.	<b>Часть 2. § 15.</b> Символьные строки.	Тест № 19. Символьные строки в функциях.		<b>1</b>	
53.	Рекурсивный перебор.	<b>Часть 2. § 15.</b> Символьные строки.	Тест № 20. Рекурсивный перебор.	ПР № 44. Рекурсивный перебор.	<b>1</b>	
54.	Решение задач по теме «Процедуры и функции».	<b>Часть 2. § 12-15.</b>		ПР № 45. Решение задач по теме «Процедуры и функции».	<b>1</b>	
55.	Контрольная работа «Процедуры и функции».				<b>1</b>	
56.	Массивы в C++. Обращение к элементу массива. Перебор элементов массива. Вывод и ввод массива.	<b>Часть 2. § 16.</b> Массивы.	Тест № 21. Массивы.	ПР № 46. Перебор элементов массива.	<b>1</b>	Массивы в C++. Обращение к элементу массива. Перебор элементов массива. Вывод и ввод массива. Заполнение массива случайными числами. Алгоритмы обработки массивов.
57.	Заполнение массива случайными числами. Алгоритмы обработки массивов.	<b>Часть 2. § 16.</b> Массивы.	Тест № 22. Заполнение массива случайными числами.	ПР № 47. Заполнение массива случайными числами, обработка массивов	<b>1</b>	Ключевые слова: массив, индекс элемента, значение элемента, константа, заполнение массива, вывод массива, ввод массива.
58.	Используем массивы. Игра «Стрельба по тарелкам». Движение. Бьём тарелки. Показываем счёт.	<b>Часть 2. § 17.</b> Используем массивы.		ПР № 48. Используем массивы. Игра «Стрельба по тарелкам». Движение. Бьём тарелки. Показываем счёт.	<b>1</b>	Используем массивы. Игра «Стрельба по тарелкам».  Ключевые слова: массив, константа,

Номер урока	Тема урока	Параграф учебника (номер, название)	Практические работы (номер, название)	Работы компьютерного практикума (источник, номер, название)	Количество часов на тему	Содержание программы
59.	Решение задач по теме «Массивы».	<b>Часть 2. § 16-17.</b>		ПР № 49. Решение задач по теме «Массивы».	<b>1</b>	инициализация, глобальные переменные, обработчик события.
60.	Контрольная работа «Массивы».				<b>1</b>	
61.	Что такое матрица? Размещение матрицы в памяти. Заполнение и вывод матрицы на экран.	<b>Часть 2. § 18.</b> Матрицы.		ПР № 50. Заполнение и вывод матрицы на экран.	<b>1</b>	Матрица. Основные операции с матрицами: объявление, заполнение, вывод на экран. Обработка элементов матрицы.  Ключевые слова: матрица, строка, столбец, перебор элементов, вложенный цикл, квадратная матрица, главная диагональ, побочная диагональ, перестановка строк.
62.	Обработка матриц. Квадратные матрицы.	<b>Часть 2. § 18.</b> Матрицы.	Тест № 23. Диагональные элементы.	ПР № 51. Обработка матриц. Квадратные матрицы.	<b>1</b>	
63.	Сложение и вычитание матриц.	<b>Часть 2. § 18.</b> Матрицы.		ПР № 52. Сложение и вычитание матриц.	<b>1</b>	
64.	Умножение матриц.	<b>Часть 2. § 18.</b> Матрицы.		ПР № 53. Умножение матриц.	<b>1</b>	
65.	Транспонирование матриц.	<b>Часть 2. § 18.</b> Матрицы.		ПР № 54. Транспонирование матриц.	<b>1</b>	
66.	Решение задач по теме «Матрицы».	<b>Часть 2. § 18.</b> Матрицы.		ПР № 55. Решение задач по теме «Матрицы».	<b>1</b>	
67.	Контрольная работа «Матрицы».				<b>1</b>	
68.	Системы управления версиями. Основные приемы работы с Git.	<b>Приложение.</b> Управление версиями.	Тест № 24. Системы управления версиями. Основные приемы работы с Git.		<b>1</b>	Системы управления версиями. Основные приемы работы с Git. Операции с файлами. Работа с удалённым архивом.

Номер урока	Тема урока	Параграф учебника (номер, название)	Практические работы (номер, название)	Работы компьютерного практикума (источник, номер, название)	Количество часов на тему	Содержание программы
69.	Операции с файлами. Работа с удалённым архивом.	<b>Приложение.</b> Управление версиями.		ПР № 56. Операции с файлами. Работа с удалённым архивом.	<b>1</b>	
70.	Резерв.				<b>1</b>	

### Предметные результаты

- 1) Владение *универсальным языком программирования высокого уровня C++*, представлениями о базовых типах данных и структурах данных; умением использовать основные управляющие конструкции.
- 2) Владение умением *понимать программы*, написанные на выбранном для изучения универсальном алгоритмическом языке высокого уровня; знанием основных конструкций программирования; умением анализировать алгоритмы с использованием таблиц.
- 3) Владение навыками *алгоритмического мышления* и понимание необходимости формального описания алгоритмов.
- 4) Овладение понятием *сложности алгоритма*, знание основных алгоритмов обработки числовой и текстовой информации, алгоритмов поиска и сортировки.

Учебный материал: 69 часов.

Резерв: 1 час.

## **11 класс (68 часов)**

### **Раздел «Алгоритмы и программирование»**

## Модуль «Алгоритмизация и программирование» (68 часа)

Номер урока	Тема урока	Параграф учебника (номер, название)	Практические работы (номер, название)	Работы компьютерного практикума (источник, номер, название)	Количество часов на тему	Содержание программы
1.	Метод пузырька (Сортировка простыми обменами).	<b>Часть 3. § 13.</b> Простые алгоритмы сортировки.		ПР № 1. Метод пузырька (Сортировка простыми обменами).	<b>1</b>	Сортировка в массивах. Методы сортировки: метод пузырька, метод выбора, сортировка вставками. Массивы в подпрограммах.  Ключевые слова: сортировка, метод пузырька, сортировка вставками, перестановка элементов, сложность алгоритма.
2.	Метод выбора. Сортировка вставками.	<b>Часть 3. § 13.</b> Простые алгоритмы сортировки.		ПР № 2. Метод выбора. Сортировка вставками.	<b>1</b>	
3.	Массивы в подпрограммах.	<b>Часть 3. § 13.</b> Простые алгоритмы сортировки.	Тест № 1. Массивы в подпрограммах.		<b>1</b>	
4.	Сортировка слиянием.	<b>Часть 3. § 14.</b> Быстрые алгоритмы сортировки и поиска.		ПР № 3. Массивы в подпрограммах.	<b>1</b>	Сортировка в массивах. Методы сортировки: сортировка слиянием, быстрая сортировка. Стандартная сортировка на языке C++. Двоичный поиск.  Ключевые слова: сортировка слиянием, быстрая сортировка, двоичный поиск, лямбда-функция.
5.	Быстрая сортировка.	<b>Часть 3. § 14.</b> Быстрые алгоритмы сортировки и поиска.		ПР № 4. Быстрая сортировка.	<b>1</b>	
6.	Стандартная сортировка на языке C++.	<b>Часть 3. § 14.</b> Быстрые алгоритмы сортировки и поиска.		ПР № 5. Стандартная сортировка на языке C++.	<b>1</b>	
7.	Двоичный поиск.	<b>Часть 3. § 14.</b> Быстрые алгоритмы сортировки и поиска.		ПР № 6. Двоичный поиск.	<b>1</b>	
8.	Принцип сэндвича. Файловые потоки.	<b>Часть 3. § 15.</b> Обработка файлов.	Тест № 2. Принцип сэндвича. Файловые потоки.		<b>1</b>	Понятие файла. Типы файлов. Этапы работы с файлами: открытие файла, запись в файл, удаление записей из файла, чтение из файла, закрытие файла. Обработка массивов, записанных в файле. Обработка строк, записанных в файле. Обработка смешанных данных, записанных в файле.
9.	Неизвестное количество данных. Обработка массивов. Чтение файлов по словам.	<b>Часть 3. § 15.</b> Обработка файлов.	Тест № 3. Обработка массивов.	ПР № 7. Чтение файлов по словам.	<b>1</b>	
10.	Построчная обработка файлов. Передача имени	<b>Часть 3. § 15.</b> Обработка файлов.		ПР № 8. Построчная обработка файлов.	<b>1</b>	

Номер урока	Тема урока	Параграф учебника (номер, название)	Практические работы (номер, название)	Работы компьютерного практикума (источник, номер, название)	Количество часов на тему	Содержание программы
	файла программе.			Передача имени файла программе.		Ключевые слова: файл, файловый поток, открытие файла, закрытие файла, чтение из файла, запись в файл, конец файла, аргументы командной строки.
11.	Решето Эратосфена.	<b>Часть 3. § 16.</b> Целочисленные алгоритмы.		ПР № 9. Решето Эратосфена.	<b>1</b>	Целочисленные алгоритмы. Решето Эратосфена. Длинные числа.
12.	«Длинные» числа в C++.	<b>Часть 3. § 16.</b> Целочисленные алгоритмы.		ПР № 10. «Длинные» числа в C++.	<b>1</b>	
13.	Динамические массивы в C++.	<b>Часть 3. § 17.</b> Динамические массивы и словари.	Тест № 4. Динамические массивы в C++.		<b>1</b>	Понятие динамического массива. Объявление и заполнение динамического массива. Использование динамических массивов в подпрограммах. Расширение массива. Принципы работы динамического массива. Тип vector из библиотеки STL. Итераторы. Словари. Перебор элементов словаря.
14.	Тип vector из библиотеки STL.	<b>Часть 3. § 17.</b> Динамические массивы и словари.		ПР № 11. Тип vector из библиотеки STL.	<b>1</b>	
15.	Итераторы. Применение.	<b>Часть 3. § 17.</b> Динамические массивы и словари.		ПР № 12. Итераторы. Применение.	<b>1</b>	
16.	Словари. Перебор элементов словаря.	<b>Часть 3. § 17.</b> Динамические массивы и словари.		ПР № 13. Алфавитно-частотный словарь.	<b>1</b>	
17.	Структуры в C++. Обращение к полям структуры.	<b>Часть 3. § 18.</b> Структуры.	Тест № 5. Обращение к полям структуры.	ПР № 14. Ввод и вывод структур.	<b>1</b>	Понятие структуры. Объявление структур. Обращение к полю структуры. Работа с файлами. Сортировка.
18.	Работа с файлами.	<b>Часть 3. § 18.</b> Структуры.	Тест № 6. Работа с файлами.	ПР № 15. Чтение структур из файла.	<b>1</b>	

Номер урока	Тема урока	Параграф учебника (номер, название)	Практические работы (номер, название)	Работы компьютерного практикума (источник, номер, название)	Количество часов на тему	Содержание программы
19.	Сортировка.	<b>Часть 3. § 18.</b> Структуры.		ПР № 16. Сортировка структур.	<b>1</b>	Ключевые слова: структура, поле, точечная запись, сортировка, ключ.
20.	Стек.	<b>Часть 3. § 19.</b> Стек, очередь, дек.		ПР № 17. Функции для работы со стеком из библиотеки STL.	<b>1</b>	Понятие стека. Использование динамического массива для создания стека. Вычисление арифметических выражений. Скобочные выражения. Очередь и деки как разновидности линейного списка.  Ключевые слова: стек, очередь, дек.
21.	Очередь. Хранение очереди в массиве.	<b>Часть 3. § 19.</b> Стек, очередь, дек.		ПР № 18. Тип моделирования очереди queue.	<b>1</b>	
22.	Дек.	<b>Часть 3. § 19.</b> Стек, очередь, дек.		ПР № 19. Тип моделирования дека queue.	<b>1</b>	
23.	Деревья в C++.	<b>Часть 3. § 20.</b> Деревья.	Тест № 7. Деревья в C++.		<b>1</b>	Дерево. Двоичное дерево. Дерево поиска. Обход двоичного дерева. Вычисление арифметических выражений. Использование связанных структур. Хранение двоичного дерева в массиве.  Ключевые слова: дерево, двоичное дерево, обход в глубину, обход в ширину, дерево поиска, ключ, хранение в массиве, модуль, проект, заголовочный файл.
24.	Обходы дерева.	<b>Часть 3. § 20.</b> Деревья.	Тест № 8. Обходы дерева.	ПР № 20. Обходы дерева.	<b>1</b>	
25.	Деревья поиска.	<b>Часть 3. § 20.</b> Деревья.		ПР № 21. Деревья поиска.	<b>1</b>	
26.	Вычисления арифметических выражений.	<b>Часть 3. § 20.</b> Деревья.		ПР № 22. Вычисления арифметических выражений.	<b>1</b>	
27.	Дерево в массиве. Модульность.	<b>Часть 3. § 20.</b> Деревья.		ПР № 23. Дерево в массиве. Модульность.	<b>1</b>	
28.	Графы в C++.	<b>Часть 3. § 21.</b> Графы.	Тест № 9. Графы.		<b>1</b>	
29.	Задача коммивояжера.	<b>Часть 3. § 21.</b> Графы.		ПР № 24. Задача коммивояжера.	<b>1</b>	
30.	Задача коммивояжера: жадный алгоритм.	<b>Часть 3. § 21.</b> Графы.		ПР № 25. Задача коммивояжера: жадный алгоритм.	<b>1</b>	Ключевые слова: жадный алгоритм, остовное дерево, задача коммивояжера, Гамильтонов цикл.

Номер урока	Тема урока	Параграф учебника (номер, название)	Практические работы (номер, название)	Работы компьютерного практикума (источник, номер, название)	Количество часов на тему	Содержание программы
31.	Задача коммивояжёра: случайные перестановки.	<b>Часть 3. § 21.</b> Графы.		ПР № 26. Задача коммивояжёра: случайные перестановки.	1	полный перебор, рекурсия, хвостовая рекурсия, случайный поиск, глобальные переменные, структуры.
32.	Как избавиться от глобальных переменных? Передача данных по ссылке.	<b>Часть 3. § 21.</b> Графы.		ПР № 27. Передача данных по ссылке.	1	
33.	Одномерные задачи.	<b>Часть 3. § 22.</b> Динамическое программирование.		ПР № 28. Одномерные задачи.	1	Динамическое программирование. Поиск оптимального решения. Количество решений. Ключевые слова: динамическое программирование, перебор вариантов, мемоизация, оптимальная программа, редактирование строк, расстояние Левенштейна.
34.	Редактирование строк.	<b>Часть 3. § 22.</b> Динамическое программирование.		ПР № 29. Редактирование строк.	1	
35.	Оптимальная стратегия.	<b>Часть 3. § 22.</b> Динамическое программирование.	Тест № 10. Оптимальная стратегия.	ПР № 30. Оптимальная стратегия.	1	
36.	Классы и объекты в C++: новая задача и ее анализ. Класс CMap.	<b>Часть 4. § 12.</b> Классы и объекты в C++.	Тест № 11. Классы и объекты в C++		1	Классы и объекты в C++: новая задача и ее анализ. Класс CMap. Пишем свой конструктор. Рефакторинг. Рисуем карты. Ключевые слова: класс, объект, экземпляр, модификатор доступа, конструктор, инициализация, метод, рефакторинг.
37.	Пишем свой конструктор. Рефакторинг.	<b>Часть 4. § 12.</b> Классы и объекты в C++.		ПР № 31. Класс CMap. Пишем свой конструктор.	1	
38.	Рисуем карты.	<b>Часть 4. § 12.</b> Классы и объекты в C++.		ПР № 32. Рисуем карты.	1	
39.	Класс CSet.	<b>Часть 4. § 13.</b> Программа с классами (практикум).		ПР № 33. Класс CSet.	1	Класс CSet. Рефакторинг. Основная программа. Разбиение на модули. Основная программа. Разбиение на модули.
40.	Рефакторинг.	<b>Часть 4. § 13.</b> Программа с классами (практикум).		ПР № 34. Рефакторинг.	1	
41.	Основная программа.	<b>Часть 4. § 13.</b> Программа с классами (практикум).		ПР № 35. Основная программа.	1	Ключевые слова: класс, объект, ссылка, конструктор, метод, рефакторинг.



Номер урока	Тема урока	Параграф учебника (номер, название)	Практические работы (номер, название)	Работы компьютерного практикума (источник, номер, название)	Количество часов на тему	Содержание программы
	Разбиение на модули.			Разбиение на модули.		
42.	Объект защищает свои данные.	<b>Часть 4. § 14.</b> Инкапсуляция.		ПР № 36. Объект защищает свои данные.	<b>1</b>	Объект защищает свои данные. Изменение внутреннего устройства. Свойство «только для чтения». Свойства в C#.
43.	Изменение внутреннего устройства.	<b>Часть 4. § 14.</b> Инкапсуляция.		ПР № 37. Изменение внутреннего устройства.	<b>1</b>	Ключевые слова: инкапсуляция, поле, метод, модификатор доступа, интерфейс, свойство
44.	Свойство «только для чтения». Свойства в C#.	<b>Часть 4. § 14.</b> Инкапсуляция.	Тест № 12. Свойства в C#.	ПР № 38. Свойство «только для чтения».	<b>1</b>	
45.	Моделирование жизни в океане. Иерархия классов. Базовый класс.	<b>Часть 4. § 15.</b> Наследование.		ПР № 39. Моделирование жизни в океане.	<b>1</b>	
46.	Абстрактный класс. Защищенные поля и методы (protected).	<b>Часть 4. § 15.</b> Наследование.		ПР № 40. Абстрактный класс. Защищенные поля и методы (protected).	<b>1</b>	Моделирование жизни в океане. Иерархия классов. Базовый класс. Абстрактный класс. Защищенные поля и методы (protected). Неподвижные объекты. Подвижные объекты. Рыбы. Хищники. Вспомогательные процедуры и функции. Основная программа.  Ключевые слова: класс, иерархия классов, базовый класс, наследование, абстрактный класс, виртуальный метод, защищенные поля, утечка памяти.
47.	Неподвижные объекты.	<b>Часть 4. § 15.</b> Наследование.		ПР № 41. Неподвижные объекты.	<b>1</b>	
48.	Подвижные объекты. Рыбы. Хищники.	<b>Часть 4. § 15.</b> Наследование.		ПР № 42. Подвижные объекты. Рыбы. Хищники.	<b>1</b>	
49.	Вспомогательные процедуры и функции.	<b>Часть 4. § 15.</b> Наследование.		ПР № 43. Вспомогательные процедуры и функции.	<b>1</b>	
50.	Основная программа. Указатели на базовый класс.	<b>Часть 4. § 15.</b> Наследование.		ПР № 44. Основная программа. Указатели на базовый класс.	<b>1</b>	
51.	Полиморфизм в действии.	<b>Часть 4. § 16.</b> Полиморфизм.		ПР № 45. Полиморфизм в действии.	<b>1</b>	Полиморфизм в действии. Класс Океан. Деструктор.

Номер урока	Тема урока	Параграф учебника (номер, название)	Практические работы (номер, название)	Работы компьютерного практикума (источник, номер, название)	Количество часов на тему	Содержание программы
52.	Класс Океан.	<b>Часть 4. § 16.</b> Полиморфизм.		ПР № 46. Класс Океан.	1	Ключевые слова: базовый класс, класс - наследник, виртуальный метод, полиморфизм, позднее связывание, таблица виртуальных методов, деструктор.
53.	Деструктор.	<b>Часть 4. § 16.</b> Полиморфизм.		ПР № 47. Дестр уктор.	1	
54.	Столкновения.	<b>Часть 4. § 17.</b> Взаимодействие объектов.		ПР № 48. Столк новения.	1	Ключевые слова: столкновение, «мертвые» объекты, итератор, преобразование типов.
55.	Изменение базового класса.	<b>Часть 4. § 17.</b> Взаимодействие объектов.		ПР № 49. Измен ение базового класса.	1	
56.	Изменение других классов.	<b>Часть 4. § 17.</b> Взаимодействие объектов.		ПР № 50. Измен ение других классов.	1	
57.	Умные указатели.	<b>Часть 4. § 17.</b> Взаимодействие объектов.		ПР № 51. Умны е указатели.	1	
58.	RAD – среды для разработки программ. Язык C# и среда .NET.	<b>Часть 4. § 18.</b> Простая программа на C#.		ПР № 52. Созда ние формы в RAD- среде.	1	
59.	Проект C#. Свойства объектов.	<b>Часть 4. § 18.</b> Простая программа на C#.		ПР № 53. Проек т C#. Свойства объектов.	1	Ключевые слова: RAD – среда, среда NET., язык C#, виртуальная машина, проект, форма, свойство, событие, обработчик события, компонент, статистический класс.
60.	Обработчики событий.	<b>Часть 4. § 18.</b> Простая программа на C#.		ПР № 54. Обраб отчики событий.	1	
61.	Программы для просмотра рисунков.	<b>Часть 4. § 19.</b> Использование компонентов.		ПР № 55. Progr аммы для просмотра рисунков.	1	Ключевые слова: компонент, свойство, событие, обработчик события, родительский объект,
62.	Ввод и вывод данных.	<b>Часть 4. § 19.</b> Использование компонентов.		ПР № 56. Ввод и вывод данных.	1	

Номер урока	Тема урока	Параграф учебника (номер, название)	Практические работы (номер, название)	Работы компьютерного практикума (источник, номер, название)	Количество часов на тему	Содержание программы
63.	Обработка ошибок.	<b>Часть 4. § 19.</b> Использование компонентов.		ПР № 57. Обработка ошибок.	1	дочерний объект, поле ввода, метка, статический метод, обработка ошибок, исключение.
64.	Вычисление арифметических выражений: модель, новый класс. Методы класса.	<b>Часть 4. § 20.</b> Создание новых классов.		ПР № 58. Вычисление арифметических выражений: модель, новый класс. Методы класса.	1	Разработка программы для вычисления арифметических выражений: модель, новый класс. Методы класса. Вычисление арифметических выражений: представление. Добавляем новый компонент. Изменение поведения компонента.
65.	Вычисление арифметических выражений: представление.	<b>Часть 4. § 20.</b> Создание новых классов.		ПР № 59. Вычисление арифметических выражений: представление.	1	Ключевые слова: модуль, модель, представление, класс, компонент, статический класс, методы класса, свойство.
66.	Добавляем новый компонент. Изменение поведения компонента.	<b>Часть 4. § 20.</b> Создание новых классов.		ПР № 60. Добавляем новый компонент. Изменение поведения компонента.	1	
67.	Защита проектов	<b>Часть 4. § 20.</b> Создание новых классов.		Проект на свободную тему.	1	
68.	Резерв				1	

## Предметные результаты

- 1) Владение навыками *алгоритмического мышления* и понимание необходимости формального описания алгоритмов.
- 2) Овладение понятием *сложности алгоритма*, знание основных алгоритмов обработки числовой и текстовой информации, алгоритмов поиска и сортировки.
- 3) Владение стандартными приёмами *написания на алгоритмическом языке программы* для решения стандартной задачи с использованием основных конструкций программирования и отладки таких программ; использование готовых прикладных компьютерных программ по выбранной специализации.

Учебный материал: 67 часов.

Резерв: 1 час.

## I. Описание учебно-методического и материально-технического обеспечения учебного процесса

Для реализации учебного курса «Программирование на Си» необходимо наличие компьютерного класса в соответствующей комплектации:

### Требования к комплектации компьютерного класса

Наиболее рациональным с точки зрения организации деятельности детей в школе является установка в компьютерном классе 13–15 компьютеров (рабочих мест) для школьников и одного компьютера (рабочего места) для педагога.

Предполагается объединение компьютеров в локальную сеть с возможностью выхода в Интернет, что позволяет использовать сетевые цифровые образовательные ресурсы.

Минимальные требования к техническим характеристикам каждого компьютера следующие:

- процессор – не ниже *Celeron* с тактовой частотой 2,8 ГГц;
- оперативная память – не менее 1Гб;
- жидкокристаллический монитор с диагональю не менее 15 дюймов;
- жёсткий диск – не менее 80 Гб;
- клавиатура;
- мышь;
- устройство для чтения компакт-дисков (желательно);
- аудиокарта и акустическая система (наушники или колонки).

Кроме того, в кабинете информатики должны быть:

- принтер на рабочем месте учителя;
- проектор на рабочем месте учителя;
- сканер на рабочем месте учителя;
- телекоммуникационный блок,
- устройства, обеспечивающие подключение к сети;
- устройства вывода звуковой информации;
- устройства для ручного ввода текстовой информации и манипулирования экранными элементами;
- устройства создания графической информации;
- устройства для создания музыкальной информации;
- устройства для записи (ввода) визуальной и звуковой информации.

### Требования к программному обеспечению компьютеров

На компьютерах, которые расположены в кабинете информатики, должна быть установлена операционная система *Windows* или *Linux*, а также **необходимое** программное обеспечение:

- интегрированная среда для программирования на языках С и С++, работающая под управлением операционной системы *Windows*. Среда *Dev-C++*, распространяемая свободно с исходными кодами (на *Delphi*) по лицензии *GPL Dev-C++* <http://www.bloodshed.net/dev/devcpp.html> ,  
а также:

- текстовый редактор (*Блокнот* или *Gedit*) и текстовый процессор (*Word* или *OpenOffice.org Writer*);
- табличный процессор (*Excel* или *OpenOffice.org Calc*);
- средства для работы с баз данных (*Access* или *OpenOffice.org Base*);
- графический редактор *Gimp* (<http://gimp.org>);
- редактор звуковой информации *Audacity* (<http://audacity.sourceforge.net>);
- среда программирования *КуМир* (<http://www.niisi.ru/kumir/>);
- среда программирования *FreePascal* (<http://www.freepascal.org/>);
- среда программирования *Lazarus* (<http://lazarus.freepascal.org/>)

и другие программные средства.

#### Дополнительное программное обеспечение

- файловый менеджер (в составе операционной системы или др.);
- антивирусная программа;
- программа-архиватор;
- клавиатурный тренажер;
- звуковой редактор;
- простая геоинформационная система;
- виртуальные компьютерные лаборатории;
- программа-переводчик;
- система оптического распознавания текста;
- мультимедиа проигрыватель (входит в состав операционных систем или др.);
- почтовый клиент (входит в состав операционных систем или др.);
- браузер (входит в состав операционных систем или др.).

## Учебно-методические средства обучения

### Основная литература

1. Поляков К.Ю. Программирование. Python. C++. Часть 1: учебное пособие /К.Ю. Поляков. – М.:БИНОМ. Лаборатория знаний, 2019.- 144 с.: ил.
2. Поляков К.Ю. Программирование. Python. C++. Часть 2: учебное пособие /К.Ю. Поляков. – М.:БИНОМ. Лаборатория знаний, 2019.- 176 с.: ил.
3. Поляков К.Ю. Программирование. Python. C++. Часть 3: учебное пособие /К.Ю. Поляков. – М.:БИНОМ. Лаборатория знаний, 2019.- 208 с.: ил.
4. Поляков К.Ю. Программирование. Python. C++. Часть 4: учебное пособие /К.Ю. Поляков. – М.:БИНОМ. Лаборатория знаний, 2019.- 192 с.: ил.

### Дополнительная литература

1. Страуструп Б. Программирование: принципы и практика с использованием C++, 2-е изд.; Пер. с англ. - М.: ООО «И. Д. Вильямс», 2016.
2. Липпман С. Б., Лажойе Ж., Му Б.Э. Язык программирования C++. Базовый курс, 5-е изд. Пер. с англ. - М.: ООО «И. Д. Вильямс», 2014.
3. Шилдт Г. C++. Базовый курс, 3-е изд. Пер. с англ. - М.: ООО «И. Д. Вильямс», 2010.
4. Лафоре Р. Объектно-ориентированное программирование в C++, 4-е изд. Пер. с англ. – СПб.: Питер, 2004.
5. Мюссер Д., Дердж Ж., Сейни А. C++ и STL: справочное руководство, 2-е изд. Пер. с англ. - М.: ООО «И. Д. Вильямс», 2010.
6. Литвиненко Н. А. Технология программирования на C++. Начальный курс.. – СПб.: БХВ-Петербург, 2010 . – 281 с.

7. Романов Е.Л. Си++. От дилетанта до профессионала – СПб.: БХВ-Петербург, 2014 . – 600 с.
8. Федеральный государственный образовательный стандарт среднего (полного) общего образования. Утвержден приказом Министерства образования и науки Российской Федерации от 17 мая 2012 г. № 413.
9. К.Ю. Поляков, Е.А. Еремин. Информатика. 10-11 классы. Программа для старшей школы. Углубленный уровень. — М.: Бином, 2016.

### **Дидактический материал**

К учебному пособию прилагаются дополнительные материалы:

- компьютерный практикум в электронном виде с комплектом электронных учебных средств, размещенный на сайте авторского коллектива: <http://kpolyakov.spb.ru/school/пусcpp.htm>;
- материалы для подготовки к итоговой аттестации по информатике в форме ЕГЭ, размещенные на сайте <http://kpolyakov.spb.ru/school/ege.htm>;
- методические материалы для учителя, размещенные на сайте автора <http://kpolyakov.spb.ru/school/пусcpp.htm>;
- подборка электронных образовательных ресурсов (далее ЭОР) с портала ФЦИОР (<http://www.fcior.edu.ru>);
- сетевая методическая служба авторского коллектива для педагогов на сайте издательства <http://metodist.lbz.ru/authors/informatika/7/>;
- интегрированная среда для программирования на языках С и С++, работающая под управлением операционной системы Windows. Среда Dev-C++, распространяемая свободно с исходными кодами (на Delphi) по лицензии GPL [Dev-C++](#) .