

# Программа курса «Основы программирования на языке C++»

Основи програмування на мові C++  
Fundamentals of C++ Programming

## Для групп ПКО Школьник. Версия 1.0.1

Продолжительность курса: 52 пары (26 дней)

### Цель курса

Обучить слушателя основам программирования на языке C++. Научить студента мыслить алгоритмически. Научить использовать блок-схемы, условия, циклы, массивы, функции, указатели и другие базовые конструкции языка программирования C++.

Объектно-ориентированное программирование не является целью данного курса.

### По окончании курса слушатель будет:

- использовать циклы, условия и другие базовые конструкции;
- оперировать основами языка программирования C++;
- понимать и использовать отладчик;
- создавать функции;
- использовать одномерные и многомерные массивы;
- применять указатели и разбираться в тонкостях арифметики указателей;
- уметь работать с динамически выделенной памятью;
- использовать алгоритмы поиска и сортировки данных.

По окончании данного курса студент не сдает экзамен.

Экзамен будет сдан по итогам двух курсов: «Основы программирования на языке C++» и «Создание пользовательских приложений с использованием C++». Экзамен должен содержать теоретическую (все темы двух курсов) и практическую часть. Для допуска к экзамену должны быть сданы все домашние и практические задания по двум курсам. В диплом будет записана полученная экзаменационная оценка.

В качестве названия предмета необходимо использовать «Основы программирования на языке C++».

Перед началом данного предмета необходимо предоставить студентам доступ к следующим курсам Cisco Networking Academy:

- Programming Essentials in C;
- Programming Essentials in C++.

## Тематический план

<b>Модуль 1.</b>	<b>Введение в язык программирования C++. . . . .</b>	<b>5 пар</b>
<b>Модуль 2.</b>	<b>Переменные и типы данных. . . . .</b>	<b>4 пары</b>
<b>Модуль 3.</b>	<b>Логические операторы и операторы ветвлений. . . . .</b>	<b>6 пар</b>
<b>Модуль 4.</b>	<b>Циклы, использование отладчика . . . . .</b>	<b>8 пар</b>
<b>Модуль 5.</b>	<b>Массивы: одномерные и многомерные . . . . .</b>	<b>8 пар</b>
<b>Модуль 6.</b>	<b>Функции . . . . .</b>	<b>10 пар</b>
<b>Модуль 7.</b>	<b>Указатели . . . . .</b>	<b>11 пар</b>

## Модуль 1

### Введение в язык программирования C++

1. Введение.
  - История и этапы развития языка C++.
  - Сравнительный анализ языка C++ с другими языками программирования.
2. Программная среда Microsoft Visual C++.
  - Установка.
  - Основы работы с IDE VC++.
  - Создание проекта.
  - Добавление файла к проекту.
  - Обзор альтернативных средств разработки.
3. Первая программа.
  - Построение первой программы на языке блок-схем.
  - Анализ первой программы.
4. Классификация символов языка.
5. Лексемы.
6. Понятие библиотеки.
7. Анализ понятий: компилятор, линковщик, интерпретатор.
8. Организация вывода данных в консоль.
9. Понятие ESCAPE-последовательности.
  - ESCAPE последовательность \n.
  - ESCAPE последовательность \t.
  - ESCAPE последовательность \b.
  - ESCAPE последовательность \».
  - ESCAPE последовательность \\\.
  - ESCAPE последовательность \a.
10. “Raw” строки.
  - Что такое “raw” строки.
  - Примеры использования “raw” строк.

## 11. Комментарии.

- Однострочные.
- Многострочные.

# Модуль 2

## Переменные и типы данных

### 1. Типы данных.

- Понятие типа данных. Размер, диапазон значений.
- Целые типы данных. Типы данных long long, unsigned long long.
- Типы данных для работы с дробными числами.
- Символьный тип данных. Типы char16\_t, char32\_t.
- Логический тип данных.
- Перечислимый тип данных (enum).

### 2. Переменная.

- Необходимость использования переменных.
- Идентификаторы.
- Ключевые слова.
- Синтаксис объявления переменных.
- Инициализация переменной. Списковая инициализация переменной.

### 3. Организация ввода данных с консоли.

### 4. Константы и литералы.

- Необходимость применения.
- Синтаксис объявления.

### 5. Операторы.

- Понятие оператора.
- Типы операторов.
- Арифметические операторы.
- Логические операторы.
- Операторы ветвлений.
- Унарные операторы.
- Бинарные операторы.
- Тернарный оператор.
- Оператор присваивания.
- Арифметические операторы.

- Оператор сложения.
  - Оператор вычитания.
  - Оператор умножения.
  - Инкремент. Постфиксная и префиксная форма.
  - Декремент. Постфиксная и префиксная форма.
  - Сокращенные формы.
6. Примеры построения программ с использованием блок-схем.

## Модуль 3

### Логические операторы и операторы ветвлений

1. Преобразование типов данных.
  - Необходимость использования.
  - Неявное преобразование типов.
  - Списковая инициализация, сужение и неявное преобразование типов.
  - Явное преобразование типов.
2. Логические операторы.
  - Знакомство с логическими операциями.
  - Таблица результатов применения логических операций.
  - «Логическое отрицание». Оператор !.
  - «Логическое И». Оператор &&.
  - «Логическое ИЛИ». Оператор ||.
3. Таблица приоритетов операторов.
4. Конструкции логического выбора. Операторы ветвлений.
  - Оператор ветвления if.
  - Оператор ветвления if – else.
  - Лестница if – else if.
  - Обозначение условий в блок-схемах. Блок условия.
  - Обозначение объединения ветвей в блок-схемах.
  - Примеры построения программ с использованием операторов ветвлений на языке блок-схем.
  - Понятие составного оператора.
  - Тернарный оператор.
  - Оператор множественного выбора – switch.

5. Понятие enum.

- Понятие enum как перечислимого типа.
- Синтаксис объявления enum.
- Использование enum для switch-конструкций.

## Модуль 4

### Циклы, использование отладчика

1. Циклы.

- Необходимость использования циклов. Примеры использования.
- Цикл while.
- Цикл for.
- Цикл do-while.
- Операторы break и continue.
- Вложенные циклы. Примеры использования.

2. Работа с интегрированным отладчиком в Microsoft Visual C++.

- Что такое отладчик. Цели и задачи отладчика.
- Запуск программы по шагам.
- Окна для работы с отладчиком. Окна переменных, локальных переменных, памяти.
- Исполнение одного шага.
- Установка точки останова (breakpoint).
- Установка умной точки останова (smart breakpoint).

## Модуль 5

### Массивы: одномерные и многомерные

1. Массивы.

- Что такое массивы. Необходимость их использования.
- Синтаксис объявления одномерного массива.
- Способы инициализации массива.
- Схема размещения массивов в памяти.
- Индексация элементов массива.
- Примеры использования массивов на языке блок-схем.

2. Алгоритмы суммирования.

3. Алгоритмы поиска (линейный, бинарный).
4. Алгоритмы сортировки.
  - Пузырьковая сортировка.
  - Сортировка выбором.
  - Сортировка вставками.
5. Многомерные массивы.
  - Многомерные массивы. Цели и задачи их использования.
  - Двумерные массивы, как частный случай многомерных.
  - Синтаксис объявления многомерного массива.
  - Примеры использования многомерных массивов.
6. Понятие статического выделения памяти.

## Модуль 6

### Функции

1. Функции.
  - Необходимость использования функций.
  - Синтаксис объявления функции.
  - Использование ключевого слова `void` при работе с функциями.
  - Вызов функции.
  - Аргументы функции.
  - Возврат значения из функции (оператор `return`).
  - Хвостовой возвращаемый тип.
  - Понятие области видимости. Локальные и глобальные переменные. Классы памяти.
  - Передача массива в функцию.
  - Прототип функции.
  - Аргументы по умолчанию.
  - Встраивание (`inline` функции).
  - Перегрузка функций.
  - Шаблоны функций:
    - понятие шаблона. Определение и объявление шаблона. Инстанционирование шаблона;
    - синтаксис объявления шаблонной функции;
    - ключевые слова `class` и `typename`;
    - отличия обычной и шаблонной функции;

- примеры создания шаблонных функций (например, Максимум, Минимум, Сортировка, Поиск и так далее);
- перегрузка шаблонных функций;
- ключевые слова `auto` и `decltype`. Автоматическое выведение типа;
- использование `decltype` в шаблонных функциях;
- хвостовой возвращаемый тип функций (использование `->`) и `decltype`.
- Рекурсия:
  - что такое рекурсия;
  - цели и задачи рекурсии;
  - примеры рекурсивных функций;
  - алгоритм быстрой сортировки.
- Функция, принимающая неограниченное количество элементов.

## Модуль 7

### Указатели

#### 1. Указатели.

- Необходимость использования указателей.
- Адрес переменной. Оператор `&`.
- Синтаксис объявления указателя.
- Косвенная адресация или оператор разыменования.
- Принцип работы оператора присвоения для указателей.
- Анализ использования `NULL` и `nullptr`.
- Связь массивов и указателей.
- Операции над указателями:
  - арифметические операции;
  - логические операции.
- Примеры работы с указателями.
- Константный указатель и указатель на константу.
- Понятие стека и динамической памяти.
- Средства языка для работы с динамической памятью:
  - оператор `new`;
  - оператор `delete`.
- Понятие ссылки `&`:
  - что такое ссылка;
  - синтаксис объявления ссылки;



- синтаксис объявления const ссылки;
- примеры.
- Понятие &&.
- Передача аргументов внутрь функции:
  - передача по значению;
  - передача по ссылке;
  - передача по указателю.
- Указатель на функцию:
  - понятие адреса функции;
  - необходимость использования указателя на функцию;
  - синтаксис объявления указателя на функцию;
  - примеры использования указателя на функцию, массива указателей на функции.