**УЧЕБНЫЙ ПЛАН**

|  |  |  |
| --- | --- | --- |
| **№****п/п** | **Наименование учебных тем** | **Трудоемкость, ак. час** |
|
|
| **1** | Программирование. Этапы разработки программ. Алгоритмы, требования к ним, рекомендации по созданию. | 1 |
| **2** | Языки программирования, их история и развитие. Языки высшего и низшего уровня. Процедурно-ориентированные и объектно-ориентированные языки. Описание Visual C++. | 1 |
| **3** | Начало работы с Visual C++. Функционал среды. | 1 |
| **4** | Типы данных, константы, арифметические операции, операции сравнения. Математические функции. | 1 |
| **5** | Ввод-вывод данных. Генерирование случайных чисел. | 2 |
| **6** | Условный оператор. Оператор-переключатель. | 2 |
| **7** | Циклические операторы. Счетный цикл, цикл с условием. | 2 |
| **8** | Статические массивы. | 2 |
| **9** | Указатели, динамические массивы. | 2 |
| **10** | Связь указателей и массивов. | 2 |
| **11** | Функции, виды функций, void-функция. | 2 |
| **12** | Рекурсия. Математическое определение и программная реализация. | 2 |
| **13** | Способы передачи данных в функцию. Передача данных по ссылке. | 2 |
| **14** | Принципы объектно-ориентированного программирования. | 2 |
| **15** | Описание классов и создание объектов в Visual C++. Атрибуты доступа, статические поля и методы. | 2 |
| **16** | Конструктор и деструктор. Управление динамической памятью. | 2 |
| **17** | Строки. Экспорт и импорт данных в внешний файл. | 1 |
| **18** | Наследование объектов. Наследование прав доступа. Множественное наследование. Многократное наследование. | 2 |
| **19** | Виртуальные методы. | 2 |
| **20** | Зависимости в рамках иерархии классов. | 2 |
| **21** | Абстрактные классы. Шаблоны. | 2 |
| **22** | Контейнеры. | 2 |
| **23** | Дружественные классы (friend). Перегруженные методы. | 1 |
| **24** | Стандартная библиотека шаблонов (STL). Итераторы. Адаптеры. | 2 |
| **25** | Исключения. Перехват исключений. | 2 |
| **26** | Функциональные объекты. | 2 |
| **27** | Событийно-управляемое программирование. Структура оконного приложения. | 2 |
| **28** | Создание оконного приложения. Элементы управления окна. | 2 |
| **29** | Графика в оконных приложениях. | 2 |
| **30** | Функции WinAPI, библиотеки MFC, Qt. | 2 |
| **31** | Среда DirectX | 1 |
| **32** | Фреймворк, каркасный подход к проектированию программных продуктов. | 1 |
| **33** | Итоговая аттестация | 2 |
|  | **ИТОГО** | **56** |

**СОДЕРЖАНИЕ УЧЕБНОЙ ДИСЦИПЛИНЫ**

|  |  |  |
| --- | --- | --- |
| **№ п/п** | **Наименование учебных тем** | **Содержание обучения по темам** |
| 1 | Программирование. Этапы разработки программ. Алгоритмы, требования к ним, рекомендации по созданию. | Этапы разработки программ: постановка задачи, разработка алгоритма, реализация алгоритма, отладка программы  |
| Основные требования к алгоритмам (универсальность, конечность, результативность, экономичность и пр.)  |
| Асимптотическая временная сложность алгоритма |
| 2 | Языки программирования, их история и развитие. Языки высшего и низшего уровня. Процедурно-ориентированные и объектно-ориентированные языки. Описание Visual C++. | Место языка программирования в системе человек-компьютер. |
| Высшие и низшие языки программирования. Парадигма программирования (функциональная, процедурная, объектно-ориентированная) |
| Язык Visual C++, структура, основные компоненты |
| 3 | Начало работы с Visual C++. Функционал среды. | Создание проекта, структура проекта |
| Отладка и запуск программы |
| Компилятор и сборщик. |
| 4 | Типы данных, константы, арифметические операции, операции сравнения. Математические функции. | Целочисленные типы данных, вещественные типы данных, символьные типы данных.  |
| Основные математические операции, операции сравнения |
| 5 | Ввод-вывод данных. Генерирование случайных чисел. | Заголовочный файл <iostream> и его компоненты.  |
| Алгоритм генерирования случайных чисел. Псевдослучайные числа. |
| 6 | Условный оператор. Оператор-переключатель. | Условный оператор (оператор ветвления), различные его формы. |
| Оператор-переключатель и особенности его использования. |
| 7 | Циклические операторы. Счетный цикл, цикл с условием. | Счетный цикл |
| Цикл с постпроверкой условия |
| Цикл с предпроверкой условия. Операторы прерывания цикла. |
| 8 | Статические массивы. | Область применения массивов. Размер и размерность массива. Выделение памяти под элементы массива. Особенности работы с массивом. |
| 9 | Указатели, динамические массивы. | Указатель как новый тип данных. Унарные операции присваивания и разыменовывания. Динамическое выделение и освобождение память. Нулевой указатель. Утечка памяти. |
| Динамические массивы, их объявление и использование. Двумерный динамический массив. |
| 10 | Связь указателей и массивов. | Имя массива как указатель. Использование алгебры указателей в массивах. |
| 11 | Функции, виды функций, void-функция. | Определение и описание функции. Входные и выходные параметры. Запуск функции. |
| Функция без выходных параметров (процедура). |
| 12 | Рекурсия. Математическое определение и программная реализация. | Математическое определение рекурсии. Конечная и бесконечная рекурсия |
| Примеры рекурсивных алгоритмов их асимптотическая временная сложность. |
| 13 | Способы передачи данных в функцию. Передача данных по ссылке. | Выделение памяти при использовании функций. Ошибка переполнения стека. |
| Передача параметров по ссылке. Константный указатель. |
| 14 | Принципы объектно-ориентированного программирования. | Парадигма объектно-ориентированного программирования. Современные оценка концепции объектно-ориентированного программирования.  |
| Три принципа объектно-ориентрованного программирования. |
| Область применения концепции объектно-ориентированного программирования. |
| 15 | Описание классов и создание объектов в Visual C++. Атрибуты доступа, статические поля и методы. | Синтаксис объявления класса. Сходства и различия между классом и структурой. |
| Атрибуты доступа в классе. Доступ по умолчанию. Средства обеспечения доступа к содержимому класса. |
| Статические поля и методы. |
| 16 | Конструктор и деструктор. Управление динамической памятью. | Конструктор, перегрузка конструкторов. Конструктор копирования. |
| Деструктор, предотвращение ошибок выделения памяти. |
| 17 | Строки. Экспорт и импорт данных в внешний файл. | Строковый класс String и его функции. |
| Файловые переменные. Текстовые файлы. |
| Двоичные файлы, функции и константы для работы с файлами. |
| 18 | Наследование объектов. Наследование прав доступа. Множественное наследование. Многократное наследование. | Синтаксис наследования объектов.  |
| Атрибуты доступа и атрибуты наследования.  |
| Конструкторы и деструкторы при множественном и многократном наследовании. |
| 19 | Виртуальные методы. | Виртуальный метод как средство реализации принципа полиморфизма. Реализация виртуальных методов. |
| 20 | Зависимости в рамках иерархии классов. | Особенности выделения и освобождения памяти, переопределения полей и методов в иерархии классов. |
| 21 | Абстрактные классы. Шаблоны. | Абстрактный класс как средство реализации принципа полиморфизма. |
| Шаблоны классов. |
| 22 | Контейнеры. | Динамически изменяемые структуры хранения данных |
| Последовательные контейнеры |
| Ассоциативные контейнеры. |
| 23 | Дружественные классы (friend). Перегруженные методы. | Друзья класса как средство обеспечения гибкости доступа. Дружественные функции, дружественные методы, дружественные классы |
| Перегрузка методов в классе. |
| 24 | Стандартная библиотека шаблонов (STL). Итераторы. Адаптеры. | Состав стандартной библиотеки шаблонов. Алгоритмы. |
| Итераторы и их отличие от указателей. Виды итераторов. |
| Адаптеры. Контейнеры-адаптеры. |
| 25 | Исключения. Перехват исключений. | Способы обработки исключений. Использование перехвата исключений (try-catch). |
| 26 | Функциональные объекты. | Функциональные объекты и их отличие от функций. Лямбда-функции. |
| 27 | Событийно-управляемое программирование. Структура оконного приложения. | Парадигма событийно-ориентированного программирования. |
| Оконные приложения. Их особенности и структура. |
| 28 | Создание оконного приложения. Элементы управления окна. | Принципы создания оконных и многооконных приложений. |
| Элементы управления в оконных приложениях. Виджеты. |
| 29 | Графика в оконных приложениях. | Использование сторонних ресурсов в оконных прилоениях. |
| Встроенные графические инструменты в оконных приложениях. |
| 30 | Функции WinAPI, библиотеки MFC, Qt. | Среды разработки оконных приложений: библиотека MFC, Qt creator. |
| 31 | Среда DirectX | Пакет средств разработки DirectX |
| 32 | Фреймворк, каркасный подход к проектированию программных продуктов. | Определение фреймворка. Каркасный подход к проектированию как средство увеличения эффективности процесса разработки и преодоления некоторых проблем. |