编辑器及C++基础

数学学院 蒋文馨 2022/03/06

腾讯会议号

蒋文馨 邀请您参加腾讯会议

会议主题: 学业辅导中心-数据结构

会议时间: 2022/03/06 19:00-21:00 (GMT+08:00) 中国标准时间 - 北京

点击链接入会,或添加至会议列表:

https://meeting.tencent.com/dm/Z0z8Hy1R3vo7

#腾讯会议: 209-889-559

手机一键拨号入会

- +8675536550000,,209889559#(中国大陆)
- +85230018898,,,2,209889559# (中国香港)

根据您的位置拨号

- +8675536550000 (中国大陆)
- +85230018898 (中国香港)

复制该信息, 打开手机腾讯会议即可参与

摘要





复习:循环和关系表达式

- For
- While (do-while)
- Break-continue
- If-else
- Switch-case

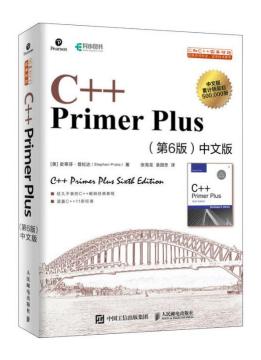
结构: struct

- · 简化的类class
- 在这门课中非常常用
- 一般不包含成员函数

```
template <class T>
struct chainNode
{
    // data members
    T element;
    chainNode<T> *next;
};
```

学会查资料 培养良好的编程习惯

推荐资料: C++



C++ Primer Plus 第6版 中文版

当字典用, 遇到不会的再去查



https://docs.microsoft.com/z h-cn/cpp/cpp/cpp-languagereference

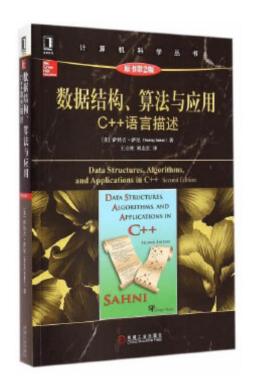
RUNOOB.COM

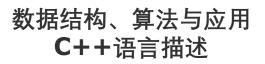
https://www.runoob.com/cpl usplus/cpp-tutorial.html

cppreference.com Create acc		unt			Search
Page Discussion			View	View source	History
C++ reference c++98, c++03, C++11, C++14, C++17, C++20, C++23 Compiler support C++11, C++14, C++17, C++20, C++23					
Freestanding implementations Language Basic concepts Keywords Prepozeros Prepozeros Declaration Initialization Functions Statements Classes Overloading Templates Exceptions Headers Named requirements Feature test macros (C++20) Language support library Source code information (C++20) Type support — traits (C++11) Program utilities Coroutine support (C++20) numeric limits — type info initializer list (C++11) Concepts library (C++20)	Diagnostics library basic stacktrace (C++23) General utilities library Smart pointers and allocators unique_ptr (C++11) shared_ptr (C++11) Date and time Function objects - hash (C++11) String conversions (C++17) Utility functions pair - tuple (C++11) optional (C++17) - any (C++17) variant (C++17) - format (C++20) Strings library basic_string basic_string basic_string basic_string view (C++17) Null-terminated strings: byte - multibyte wide Containers library array (C++11) - vector - deque more dered_map (C++11) set_onordered_set_(C++11) set_onordered_set_(C++11) set_onordered_set_(C++11) set_onordered_set_(C++12) Sequence - associative unordered associative - adaptors library	Algorit Numeri Com Math Math Num Pseu Float Bit n comp rati Localiz Input/o Stree Sync I/O n Filesys Regula basi Atomic atom atom Thread	mematical co eric algorith do-random ting-point er ananipulation lex - val o (C++11) ations libratherizations intronized out ananipulators tem librat r expressi c_regex - operation tic_ref (C+	unctions ecial functions nstants (C++; ims number gener vivronment (C-(C++20) array ary) (C++17) ons library algorithms is library (C++20) (1 c flag +20) ibrary (C++1)	20) ation ++11) (C++11)

https://en.cppreference.com/w/

推荐资料数据结构







大话数据结构

其他:

- 神书《算法导论》
- •《算法》 (第4版)
- •《啊哈!算法》
- •《算法图解》



https://leetcode-cn.com/

编辑器的选择

• DEV C++

₩ Get Dev-C++

新手友好简单好用;

DEV Download Embarcadero Dev-C++ 6.3 for Windows 7 and later

Supports Windows 7, 8, 10, 11

没有自动补全和Debug;

This is a new fork of Dev-C++ sponsored by Embarcadero (developers of Delphi, C++ Builder and RAD Studio). Includes the TDM-GCC compiler

轻量级;

报错提示不友好。

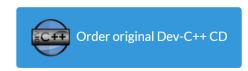


Supports Windows 98, NT, 2000, XP

Dev-C++ 5.0 (4.9.9.2) with Mingw/GCC 3.4.2 compiler and GDB 5.2.1 debugger (9.0 MB)

■ Download original Dev-C++ 5 IDE only

Dev-C++ 5.0 (4.9.9.2), IDE only - no compiler included (2.4 MB)



Includes Dev-C++ 4 & 5, extra Dev-C++ Packages, Dev-Pascal, documentation and other software

💖 Donate to the project

编辑器的选择

Visual Studio

主流的IDE(集成开发环境);

最适合C++的编辑器;

超棒的自动补全和Debug;

启动缓慢, 编译缓慢;

绝大多数功能都用不到;

大约要11GB。

系统驱动器(C) 6.26 GB 其他驱动器 4.1 GB 要求的总空间 10.35 GB 下载时安装 ▼ 安装(I) https://visualstudio.microsoft.com/zh-hans/free-developer-offers/

仅限Windows用户!!



安装注意事项:

- 安装community=免费版就足够 用了;
- 工作负荷选择右图所示的这一个;
- 确保电脑有足够的空间,安装 位置可以更改到非系统盘;
- 如果没有把握可以去B站找安装 教程。



编辑器的选择

https://code.visualstudio.com/

Visual Studio Code (VSCode)

全能的轻量级编辑器(可运行多种语言);

超棒的自动补全和Debug;

自由;

安装、配置有一点点麻烦(B站教程时长10分钟);

大约要300MB。

Windows、Mac和Linux都可以用!

配置提示:下载MinGW;修改编码,不然中文会乱码。



Visual Studio Code

■ 🕻 🐧 版本 1.62

在 Windows、macOS 和 Linux 上运行的独立 源代码编辑器。 JavaScript 和 Web 开发人员 的最佳选择,具有大量扩展以几乎支持任何编 程语言。

使用 Visual Studio Code 即表示你同意其 许可证 & 隐私声明

下载 Visual Studio Code ∨

根据自己的需求选择,适合自己的才是最好的! (也可以都下载试一试)

复习: 循环和关系表达式

- For
- While (do-while)
- Break-continue
- If-else
- Switch-case

休息一下



https://pan.baidu.com/s/1nDf2_aLyDKwR_f2fYap8jg

提取码: o97p

复习: 函数与参数

- 传值参数
- 模板函数
- •参数引用
- 常量引用参数

引用: &

引用是一个<mark>别名</mark>,是某个<mark>已存在变量</mark>的另一个名字。

指针: *

指针是一个变量, 其值为另一个变量的地址(如0xd9051ff730)。

区别:

- 不存在空引用。引用必须连接到一块合法的内存。
- 一旦引用被初始化为一个对象,就不能被指向到另一个对象。指针可以在任何时候指向到另一个对象。
- 引用必须在创建时被初始化。指针可以在任何时间被初始化。

复习: 动态存储分配和异常处理

- 动态存储分配 new, delete
- 异常处理 try-catch, throw

结构: struct

- 简化的类class
- 在这门课中非常常用
- 一般不包含成员函数

```
template <class T>
struct chainNode
{
    // data members
    T element;
    chainNode<T> *next;
};
```

类和对象: class

- 类成员访问属性
- 构造 析构函数 拷贝构造函数
- 静态变量 static
- 模板类
- 友元
- this指针
- 继承
- 多态 虚函数

Public 类外部可访问,

Protected 派生类+类和友元函数可访问,

Private 只有类和友元函数可访问。

创建对象时使用同一类中已存在的对象来初始化新创建的对象。

类中静态对象只有一个副本。

不是类的成员函数, 可以访问类的私有成员和保护成员。

每一个对象都能通过 this来访问自己的地址。

公有继承: is-a关系; 私有保护继承: has-a关系。

多种形态: 同一个函数名在基类和派生类功能不同。

虚函数:在基类中用virtual声明。

代码规范和编程习惯

- 写注释:程序功能,函数功能,变量名解释;
- 规范命名,不要偷懒;
- 代码可读性;
- 0 0 0

结束