面向对象程序设计(00P)

Created with Aspose Slides for Java 22.7.
Copyright 2004-2022 Aspose Pty Ltd.

福州大学·软件学院·软件工程系 王灿辉(wangcanhui@fzu.edu.cn)

学习要求

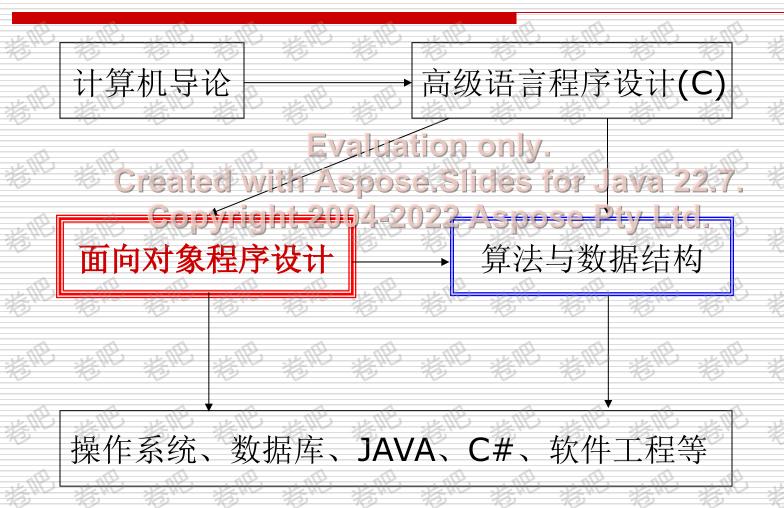
- > 选课结果
- >继续严要求: 除病假外不允许缺课!
- > 要求预习和复约-2022 Aspose Pty Ltd.
- 〉自觉完成作业
- 一不易混过,也无法靠临时抱佛脚考过
- > 坚持4年大学编程不断线。

课程性质

- > 学科基础课
- > 必修课 Evaluation only.

 Created with Aspose Slides for Java 22.7.
- >学分,授课/上机学时: 见教务网
- 》保证课外的课时,自己阅读"网络教学 综合平台"的教学大纲和进度等,以了 解教学要求的知识点和重点、难点。

课程定位



成绩构成

- > 以期末笔试成绩为主(70%--90%)
- > 平时综合成绩(10%m30%)构成详见"网络综合教学平台"(平时综合成绩不及格不允许参加课程的期末考试):
 - 出勤率、课堂练习(小测)
 - 作业情况、访问网络教学平台、0J系统上的做题情况等

主要参考书

- 1. (美) Bjarne Stroustrup著,裘宗燕 译, C++程序设计语言,机械工业出版社, 2002. 7。如果来语水平较高可以直接阅 读该书对应的原版(电子)书。
- 2. (美)Bruce Eckel著,刘宗田等译, c++ 编程思想(第2版、第1、2卷),机械工业出版社,2010.8。

教学的总体安排

- ▶不介绍(考)C++11相关的内容
- ➤不会按照教材的顺序授课,详细教学安排见"网络教学综合平台"的学进度总体安排等,可以据此进行复习和预习。部分章节(知识点)试点采用"翻转课堂式"教学模式。

教学内容及教学顺序

- > 绪论、C++基础
- > 类和对象、运算符重载

- ➤ 群体数据组织、C++流和I/0系统
- > 泛型程序设计与C++标准模板库
- > MFC库与Windows程序开发概述
- 一命名空间和其他高级主题

有最好的编程语言?

- ▶ 再次重申:真正学好一种计算机语言以后,再学同一类语言会非常轻松,这也是很多学校还在开设C/C++课的原因。

面向对象程序设计(00P)

Evaluation only.

Created with Asposasiides for Java 22.7.

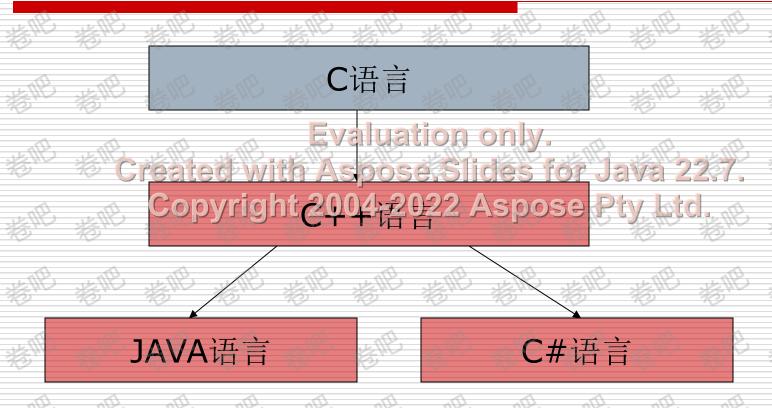
Copyright 2004 2002 Asposa FM Utd.

福州大学·软件学院·软件工程系 王灿辉(wangcanhui@fzu.edu.cn)

C++简介

- ➤ 1979年,[美]AT&T贝尔实验室: Bjarne Stroustrup,常类的C.
- 》继续坚持"程序是由程序员来管理的, 而不是由语言控制的"这个C的基本设 计理念(现在已经受到质疑)。
- ➤ C++不是纯正的00语言。

C++简介



> 语法上几乎一致

C++简介

➤ C++与C语言基本完全兼容。20世纪80年代以 来,C和C++一直是使用最为广泛的商业应用 开发语言。这两种语言在带来强大控制能力 和高度灵活性的同时,代价是相对较长的学 习周期和较低的开发效率,而且对控制能力 和灵活性的滥用也给程序的安全性带来了潜 在的威胁。C++语言过度的功能扩张也破坏了 面向对象的设计理念。

C++简介

▶ C++对C语言的改进表现在C++增加了一 些新的功能并对C中的一些不足之处作 了修匠,使乙更加完善(如:强化函数 原型、对枚举、类型转化等要求更严格 等)。C++对C语言的突破表现在C++是一 个面向对象的程序设计语言(00P),由 此对C进行了很多必须的配套的修改。

C++简介

> 如果有较扎实的C基础,则学习C++会非 常容易,但C++的内容比较庞杂,所以 需求记忆的东西比较多。很多人也会觉 得学习C++更为无聊。当然最好的学习 方法不是死记硬背,而是多动手编程, 这样就可以自然记住。

C++简介

>在VC等IDE里面,创建源程序时加上.c 扩展名后,编译系统就会知道这个是C 源(程序),从而创建的是C的源文件, 编译的时候就会调用C编译器。而不加 .c(默认扩展名是cpp),创建的就是 C++的源文件,编译的时候会调用C++的 编译器。

C++标准

➤ C++真正正式公布的标准就三个: C++98、 C++03、C++11。其中C++98是第一个正式C++ 标准,C++03是在C++98上面进行了小幅度的 修订,C++11则是一次全面的大进化(之前称 C++11为C++0x, 以为会在08~09年公布, 没 想到拖到了11年), C++11现在非常多的编译 器不支持。至于C89、C99这些,是C语言的标 准,不是C++的标准,很容易被混淆。

C++标准&可移植性

- ➤ 其实C++在第一个标准C++98之前就已经广为使用了。只不过那时候还没有一个官方的统一标准,后来才开始起草标准草案,起草了n年,直到98年才最终发布。所以你可以看到一些早期的C++编译器对标准的支持程度很差,比如经典的VC6.0,在研发的时候C++的第一个标准还没公布。
- 可移植性:如果在不修改代码的情况下,重新编译程序后,程序将运行良好,则程序是可移植的。

C++的3种编程方式

- ➤ C++融合了3种不同的编程方式:以C语言代表的过程性语言(结构化编程、自顶向下的设计)、C++在C语言基础上添加的类代表的面向对象语言(使用自下而上的编程方法)、C++模板支持的泛型编程。
- ➤ 00P和过程性编程的区别之一在于,00P强调的是在运行阶段(而不是编译阶段)进行决策,它提供一定的灵活性。

C++简介: C++编译器

Visual C++ Evaluation only.

- Dev Created with Aspose Slides for Java 22.7. Copyright 2004-2022 Aspose Pty Ltd.
- ➤ Borland的C++ Builder
- ➤ G++等

一个简单的C++程序

```
/*This is a simple C++ program. */
#include <iostream>
//A Chaprogram begins atomain (2.7.
int main () right 2004-2022 Aspose Pty Ltd.
  std::cout<< "Hello world!\n"; //输出
  return 0;
```

另一个简单的C++程序

```
#include <iostream>
using namespace std;
int main() { Evaluation only. doublerlangthvith //this deplaces a variable.
  double widthight //this declares another variable
  cout << "Enter the length:";
  cin >> length; //input the length
  cout << "Enter the width:";
  cin >> width; //input the width
  cout << "The area is: " << length*width << endl;
  return 0;
```

cin缓冲输入

```
#include <iostream>
int main() { Evaluation only.
    Chaird with Aspose. Slides for Java 22.7.
    std::cin>>ch;//跳过前头的空白符
    std::cout<<ch;
    return 0;
} //cin和cout均为有缓冲的I/0
//用while (cin.get(ch))//不跳空白符
```

求一元二次方程的根

```
#include <iostream>
#include <cmath>//等价于C的<math.h>
using chamespace pstd Jides for Java 22.7.
void main () (12004-2022 Aspose Pty Ltd.
   // .....
   cin>>a>>b>>c;
                        //完整的程序
```

读入字符并计数

```
#include <iostream>
int main() {
    using namespace std:
    int ch: //为了和EOF比较,最好不定义为char
    int cnto Olght 2004-2022 Aspose Pty Ltd.
    while ((ch=cin.get())!=EOF) {
        cout. put (char (ch));
        cnt++;
    cout << end 1 << cnt << " characters read \n";
    return 0;
```

自动按表达式类型输出

```
#include <iostream>
using namespace std:
int main () with Aspose Slides for Java 22.7.
  cout<< '0' +2<<end1; //输出50
  cout<<char('0'+2)<<end1; //输出
  return 0;
```

运算符的优先级

```
#include <iostream>
using namespace std:
cout<<ii+j<<end1; //输出20
 cout<<ii&j<<end1; //编译时报错!
```

本部分内容讲授到此结束!



福州大学·软件学院·软件工程系 王灿辉(wangcanhui@fzu.edu.cn)