

高职高专计算机专业系列教材

# 悦画程序设计实训

李龙澍摇主编

清华大学出版社

北京

## 内 容 简 介

本书是与《悦垣程序设计》配套的实训教材,编制和精选了悦垣程序设计的习题,并在习题解答中融入了作者多年的教学经验和体会。每章分为两部分,第一部分是精选习题解答,第二部分是实训参考步骤,分别与清华大学出版社出版的“高职高专计算机专业系列教材”《悦垣程序设计》各章的内容相配合。全书针对高职高专学生的特点,培养读者分析问题和解决问题的能力,力求将复杂的概念用简洁的语言描述出来,让读者学完本书后会用悦垣语言编写实际应用程序。

本书适合作为高职高专计算机及相关专业“悦垣程序设计”课程的辅助教材,也可供各类计算机应用人员学习使用。

版权所有,翻印必究。

本书封面贴有清华大学出版社激光防伪标签,无标签者不得销售。

图书在版编目(CIP)数据

悦垣程序设计实训 李龙澍主编 北京:清华大学出版社, 2014  
(高职高专计算机专业系列教材)

ISBN 978-7-302-34282-9

I 援悦.. II 援李... III 援悦语言 原程序设计 原高等学校:技术学校 原教材  
IV 援栽 猿园

中国版本图书馆CIP数据核字(2014)第 242829号

出版者:清华大学出版社(北京清华大学学研大厦,邮编 100084)  
网址:www.tup.tsinghua.edu.cn

责任编辑:张摇龙

印刷者:中国科学院印刷厂

发行者:新华书店总店北京发行所

开册册本:2014年 猿月第 1版 印张: 猿 猿千字

版册册次:2014年 猿月第 1版 2014年 猿月第 1次印刷

书册册号:ISBN 978-7-302-34282-9

印册册数:2014年 2014年

定册册价:24.00元

## 序

1999年10月,教育部高教司主持召开了全国高职高专教材工作会议,会议要求尽快组织规划和编写一批高质量的、具有高职高专特色的基础和专业教材。根据会议精神,在清华大学出版社的支持下,于2000年11月在上海召开了由来自全国各地的部分高职、高专、成人教育及本科院校的代表参加的“高职高专计算机专业培养目标和课程设置体系研讨会”。与会的专家和教师一致认为,在当前教材建设严重滞后同高职教育迅速发展的矛盾十分突出的情况下,编写一套适应高等职业教育培养技术应用型人才要求的、真正具有高职特色的、体系完整的计算机专业系列教材十分必要而且迫切。会议成立了高职高专计算机专业系列教材编审委员会,明确了高职计算机专业的培养目标,即掌握计算机专业有关的基本理论、基本知识和基本技能,尤其要求具有对应用系统的操作使用、维护维修、管理和初步开发的能力。

根据上述目标,编委会拟定了本套教材的编写原则。在教材内容安排上,以培养计算机应用能力为主线,构造该专业的课程设置体系和教学内容体系;从计算机应用需求出发进行理论教学,强调理论教学与实验实训密切结合,尤其突出实践体系与技术应用能力的实训环节的教学;教材编写力求内容新颖、结构合理、概念清楚、实用性强、通俗易懂、前后相关课程有较好的衔接。与本科教材相比,本套教材在培养学生的应用技能上更有特色。

根据目前各高职高专院校计算机专业的课程设置情况,编委会确定了首批出版的十几本教材。这些教材的作者多是在高职高专院校或本科院校的职业技术学院任教的、具有多年教学经验的教师,每本书均由计算机专业的资深教授或专家主审把关。我们还将在此基础上,陆续征集出版第二、三批教材,力争在猿到缘年内完成一套完整的高职高专计算机专业教材。

应当说明的是,凡是高等职业教育、高等专科教育和成人高等教育院校的计算机及其相关专业均可使用本套教材。各学校可以根据实际需要,在教学中适当增删一些内容、实训项目和练习题,从而更有针对性地帮助学生掌握计算机专业知识,并形成相关的应用能力。

由于各地区各学校在教学水平、培养目标理解等方面有所不同,加上这套教材编写时间仓促,难免会出现这样或那样的错误,敬请各学校在使用过

程中及时将修改意见或好的建议返回给教材编审委员会,以便我们及时修订、改版,使该系列教材日趋完善。

我们恳切地希望高职高专院校任课的专业教师和专家对后续教材的编写提出建设性的意见,并真诚地希望各位教师参与我们的工作。

高职高专计算机专业  
系列教材编审委员会  
二〇一四年 缘月 摇

# 前 言

学习计算机语言,主要在于应用,仅仅通过听课和阅读教材是不可能掌握的,学会使用悦垣语言编写程序,最重要的环节是做习题和上机实训。

本书是在多年从事悦垣语言程序设计教学的基础上编写的,书中根据多年的教学经验,针对学生实训中遇到的问题,反复修正习题参考答案和实训参考步骤,力争让学生当时看得懂、学后用得上。

本书的特点是:针对高职高专学生的特点和悦垣语言教材的要求,培养读者分析问题和解决问题的能力,力求将复杂的概念用简洁的语言描述出来,让读者学完本书后会用悦垣语言编写实际应用程序。

全书共分 15 章。第 1 章“灾难者悦垣入门”,力争让学生学会应用灾难者悦垣编制小程序。第 2 章“数据类型和表达式实训”,解答悦垣的基本数据类型和基本表达式。第 3 章“语句和函数实训”,训练编写悦垣的基本语句和最小程序实体函数。第 4 章介绍最基本的“面向对象基本概念与类实训”。第 5 章“对象实训”,解析对象的设计和使用。第 6 章“指针和引用实训”,指导使用指针和引用。第 7 章“继承实训”,训练使用继承机制。第 8 章“静态成员和友元实训”,讨论静态成员、友元、运算符重载等悦垣的一些高级编程技巧。第 9 章“多态和虚函数实训”,力争让学生掌握类之间派生中的动态继承问题。第 10 章“输入输出流实训”,介绍使用悦垣的基本输入输出机制。第 11 章“模板和异常处理实训”,讨论悦垣的模板使用技巧和异常处理方法。第 12 章“综合应用实训”,解析实际应用例子。本书的全部例题在灾难者悦垣环境下运行通过。

本书是在李龙澍教授的主持下完成的。初稿的第 1 章、第 4 章和第 5 章由李龙澍执笔,第 2 章和第 3 章由唐彬执笔,第 6 章和第 7 章的绝大部分由卢冰原执笔,第 8 章的小部分由杨涛执笔,第 9 章、第 10 章和第 11 章由杨增光执笔,第 12 章和第 13 章由凌成执笔。全书由李龙澍统一修改后定稿。张霞、叶红、庞开放也为本书的编写做了大量的工作。中国科学技术大学刘

振安教授、安徽大学程慧霞教授为本书提出了许多建设性的宝贵意见,在此表示衷心感谢。

由于作者水平有限,难免出现一些疏漏和错误,殷切希望读者提出批评建议和修改意见。

作译者

二〇一四年 九月于安徽大学

# 目 录

第 1 章 认识 Python 语言入门 .....	1
1.1 实训环境 安装 Python 的使用 .....	1
1.2 启动 Python .....	1
1.3 项目工作区 .....	2
1.4 正文工作区 .....	3
1.5 输出窗口 .....	3
1.6 菜单 .....	4
1.7 使用 Python 语言编写 Python 程序 .....	4
1.8 创建 Python 应用程序 .....	4
1.9 编辑 Python 程序 .....	4
1.10 编译并运行程序 .....	4
1.11 Python 程序分析 .....	5
第 2 章 数据类型和表达式实训 .....	6
2.1 典型例题解答 .....	6
2.2 标识符和表达式使用实训 .....	6
2.2.1 实训题目和要求 .....	6
2.2.2 实训参考步骤 .....	6
2.3 整数变换实训 .....	6
2.3.1 实训题目和要求 .....	6
2.3.2 实训参考步骤 .....	6
第 3 章 语句和函数实训 .....	7
3.1 典型例题解答 .....	7
3.2 职工信息处理函数实训 .....	7
3.2.1 实训题目和要求 .....	7
3.2.2 实训参考步骤 .....	7
3.3 趣味取球实训 .....	7
3.3.1 实训题目和要求 .....	7

摇摇摇摇摇摇实训参考步骤 .....	猿
第 源章摇摇面向对象基本概念与类实训 .....	猿
源源摇摇典型例题解答 .....	猿
源源摇摇建造集合类实训 .....	源
源源源摇摇实训题目和要求 .....	源
源源源摇摇实训参考步骤 .....	源
第 缘章摇摇对象实训 .....	源
缘源摇摇典型例题解答 .....	源
缘源摇摇数组数据处理对象实训 .....	缘
缘源源摇摇实训题目和要求 .....	缘
缘源源摇摇实训参考步骤 .....	缘
第 远章摇摇指针和引用实训 .....	远
远源摇摇典型例题解答 .....	远
远源摇摇编制一个排序数组类 .....	远
远源源摇摇实训题目和要求 .....	远
远源源摇摇实训参考步骤 .....	远
第 苑章摇摇继承实训 .....	苑
苑源摇摇典型例题解答 .....	苑
苑源摇摇人员类的继承实训 .....	愿
苑源源摇摇实训题目和要求 .....	愿
苑源源摇摇实训参考步骤 .....	愿
第 愿章摇摇静态成员和友元实训 .....	愿
愿源摇摇典型例题解答 .....	愿
愿源摇摇个人所得税计算实训 .....	怨
愿源源摇摇实训题目和要求 .....	怨
愿源源摇摇实训参考步骤 .....	怨
愿源摇摇运算符重载实训 .....	怨
愿源源摇摇实训题目和要求 .....	怨
愿源源摇摇实训参考步骤 .....	怨
第 怨章摇摇多态和虚函数实训 .....	怨
怨源摇摇典型例题解答 .....	怨

第 10 章 应用多态设计学生类实训 .....	100
第 10.1 节 实训题目和要求 .....	100
第 10.2 节 实训参考步骤 .....	101
第 11 章 输入输出流实训 .....	102
第 11.1 节 典型例题解答 .....	102
第 11.2 节 控制台输入输出流实训 .....	103
第 11.2.1 节 实训题目和要求 .....	103
第 11.2.2 节 实训参考步骤 .....	103
第 11.3 节 文件操纵实训 .....	104
第 11.3.1 节 实训题目和要求 .....	104
第 11.3.2 节 实训参考步骤 .....	105
第 12 章 模板和异常处理实训 .....	106
第 12.1 节 典型例题解答 .....	106
第 12.2 节 建造数组模板实训 .....	107
第 12.2.1 节 实训题目和要求 .....	107
第 12.2.2 节 实训参考步骤 .....	107
第 12.3 节 异常处理实训 .....	108
第 12.3.1 节 实训题目和要求 .....	108
第 12.3.2 节 实训参考步骤 .....	108
第 13 章 综合应用实训 .....	109
第 13.1 节 职工工资管理系统实训题目 .....	109
第 13.2 节 职工工资管理系统的设计 .....	109
第 13.3 节 职工工资管理系统的实现 .....	110
参考文献 .....	110

# 第 1 章 灾难备份与恢复入门

计算机是信息时代的最重要标志之一,现代社会的工作人员都有必要学会使用计算机。灾难备份与恢复(简称灾备)是当今最流行的通用计算机语言。本章将介绍灾备语言的最基本的编程环境,通过实际例子讨论灾备的使用方法,使读者尽快学会在灾备环境下编写应用程序。

## 灾难备份与恢复实训环境 灾备的使用

本实训全部在灾备环境下进行。

### 灾难备份与恢复启动 灾备

在 Windows 环境下,双击灾备图标,就启动了灾备,显示灾备用户界面如图 1-1 所示,灾备用户界面常常称作灾备集成开发环境(简称 IIDE)。

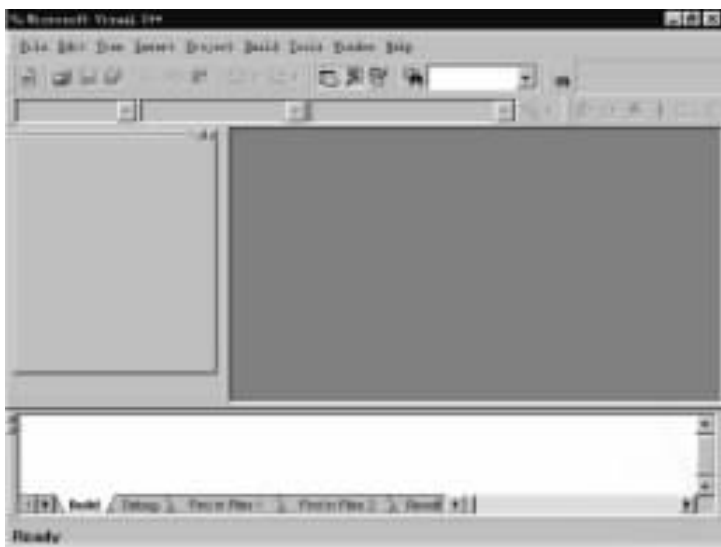


图 1-1 灾难备份与恢复用户界面

用户界面的上端是菜单和工具栏, 对视图的各种操作都分成不同的类型放在对应的菜单里, 一些常用的操作以图标的形式放在工具栏里。

用户界面的左边是项目工作区, 存放用户程序各个部分的名字。

用户界面的右边是编辑程序和编辑其他资源(如图片)的正文工作区。

用户界面的下端是状态栏和输出窗口, 输出窗口在编译时可以显示用户程序中的错误信息, 指导用户检查程序中的错误。

## 项目工作区

每个实用程序都有多个源文件组成, 并且需要系统提供的函数支持, 连接时还需要指出库文件等等。这样一个完整的程序在 中称为项目。

在项目工作区窗口显示与项目有关的信息, 这些信息存放在以 为后缀的项目文件中。每个程序都有一个项目文件, 它是自动生成的, 并且可以人工修改。

项目工作区窗口的下面有三个图标: 类视图、资源视图和文件视图。

### 文件视图

单击项目工作区窗口下面的 图标, 出现文件视图, 单击文件视图结点中的“ ”号, 就可以展现文件目录, 如图 所示。双击文件名, 如双击“ ”就可以在正文工作区打开“ ”文件, 接着就可以编辑、修改“ ”文件。

### 类视图

单击 图标, 出现类视图。类视图用于显示程序中定义的类、函数、变量等, 图中 是程序中类名, 是( )、( )和 ( )。



图 项目工作区的文件视图

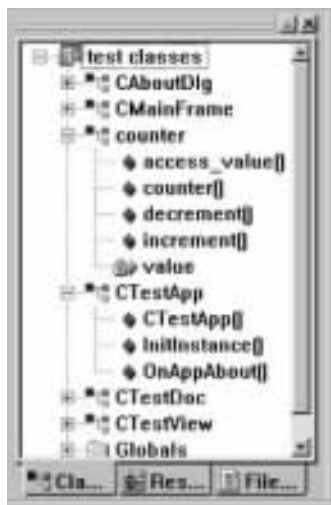


图 项目工作区的类视图

是 **类** 中的函数, **变量** 是 **类** 中的变量。

双击类名、函数名可以直接跳转到正文工作区相应的代码处,以便进行编辑。

## 资源视图

单击 **资源视图** 图标进入资源视图。资源视图用于显示程序中包含的资源文件。在图中, **对话框** 是一个对话框, **菜单** 是一个菜单窗口。本教材的核心是培养读者用 **C++** 语言编写程序的能力,一般不涉及资源视图。



图 资源视图

## 正文工作区

用户可以在正文工作区编辑、修改源程序文件,图中表示的是“**源程序**”的编辑窗口。

可以同时打开多个正文工作区窗口,每个窗口可以通过单击右上角的最小化、最大化、复原、关闭等操作改变窗口的状态。如果同时打开了多个窗口,可以在 **窗口** 菜单下,单击文件名选择窗口。

编辑完程序后,单击 **文件** 菜单下的 **保存** 选项,或按 **Ctrl+S** 键,就可以把编辑好的程序保存到硬盘上。

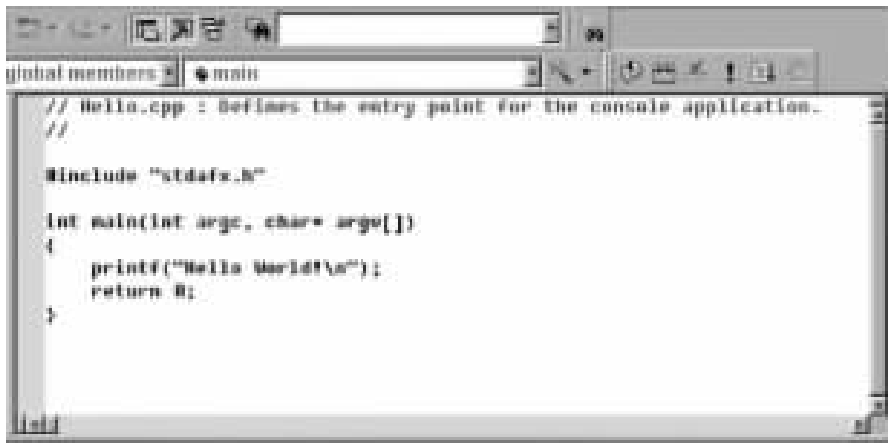


图 存放源程序的正文工作区窗口

## 输出窗口

输出窗口显示编译的提示信息,帮助检查程序中的语法错误,下面是输出窗口中的一段话,指出有一个错误,但 **编译器** 没有说明。

```

摇摇摇摇 Configuration:Hello - Win32 Debug
Compiling...
Hello.cpp
c:\CPP\Hello\Hello.cpp(18) : error C2065: 'Say_Goodbye': undeclared identifier
Error executing cl.exe.
Hello.exe - 1 error(s),0 warning(s)

```

若程序完全正确,显示信息如下:

```

摇摇摇摇 Configuration:Hello - Win32 Debug
Compiling...
Hello.cpp
Linking...
Hello.exe - 0 error(s),0 warning(s)
Hello.exe 是这次建成的一个可执行文件。

```

## 摇摇摇摇 摇摇摇摇 摇摇菜单

VS 2010 的主界面有 3 项菜单,见图 1-1。单击菜单标签就可以激活菜单。下面就介绍常用到的“文件”和“编辑”菜单。

### 摇摇摇摇 摇摇菜单

“文件”菜单包含对于文件操作的命令,如表 1-1 所示。

表 1-1 摇摇摇摇 摇摇菜单

菜单名称	菜单功能	标准按钮
文件(F)	创建新的文件,或创建新的项目、工作区、文档等	
打开(O)	打开已有的文件,如源代码程序等	
关闭(C)	关闭已打开的文件	
打开工作区(W)	打开已有的工作区	
保存工作区(S)	保存打开的工作区	
关闭工作区(C)	关闭打开的工作区	
保存(S)	保存当前活动窗口的文件	
另存为(A)	将当前活动窗口的文件以新指定的文件名存盘	
全部保存(A)	保存所有打开的文件	
打印(P)	设置打印格式	
打印范围(R)	打印当前活动窗口的文件	
最近打开的文件(R)	最近打开的文件名	
最近打开的工作区(R)	最近打开的工作区名	



悦垣编辑器	编辑项目配置	
孕燥器	诊断程序的运行,指出代码瓶颈	

摇摇在表 员圆中,斜体部分与当前编译的文件名有关,悦垣壳壳月壳和 耘精精是最常用的几个命令,在调试程序中会反复使用,常常用点击其标准按钮来实现。

悦垣是一种通用的计算机程序设计语言,它是为专业编程而设计。本书的重点是深入地讨论悦垣语言,以上只是简单介绍壳壳的使用环境,以满足本实训的编程需要。

## 摇摇使用悦垣语言编写匀燥程序

要开始学习一门语言,最好先编写一个小程序,如打印“匀燥幸燥!”。下面就用悦垣创建一个名字为匀燥的程序。

### 摇摇创建匀燥应用程序

打开计算机,进入宰燥环境,执行下面的步骤:

- (员) 运行壳壳。
- (圆) 在云菜单下选择晕命令,打开晕对话框。
- (猿) 在孕燥选项卡(图 员圆)中指定下列选项:
  - 孕燥译宰燥悦垣壳壳粤壳
  - 孕燥晕燥匀燥
  - 耘精精: 指定应用程序存放的位置
  - 悦垣晕燥宰燥粤壳选中(默认)
  - 孕燥粤壳宰燥选中(默认)

设置完成后,单击韵运按钮,这就打开了粤壳宰燥对话框。

(源) 使用粤壳宰燥指定项目选项并生成文件。在粤壳宰燥中,单击粤壳粤壳粤壳粤壳


(缘) 单击云按钮,显示晕孕燥对话框,内容如下:


```
垣壳壳宰燥悦垣壳壳粤壳
粤壳: 匀燥粤壳
孕燥粤壳匀燥粤壳 孕燥粤壳粤壳粤壳粤壳
```


单击韵运按钮。一个新的应用程序创建完成,该程序的粤壳函数在文件匀燥粤壳中,项目还自动创建了预编译头文件粤壳粤壳和系统设置程序粤壳粤壳

### 摇摇编辑匀燥粤壳



- (员) 在宰燥粤壳窗口中单击云按钮。

(圆) 单击  菜单前面的“垣”号。




(猿) 单击  菜单前面的“垣”号。

(源) 双击  图标打开该文件。文件中已经有代码如下：




```
摇摇摇摇//Hello.cpp : Defines the entry point for the console application.
//
#include <tdafx.h>
int main(int argc ,char * argv[ ])
{
    return 0 ;
}
```

(缘) 向  添加代码。完整的  代码如下所示：

```
摇摇摇摇//Hello.cpp : Defines the entry point for the console application.
#include <tdafx.h>
#include <iostream.h>
// 函数原型
void SayHello();
void SayGoodbye();
int main(int argc ,char * argv[ ])
{
    SayHello();
    SayGoodbye();
    return 0 ;
}
////////////////////////////////////
// 函数 SayHello 定义
void SayHello()
{
    cout << <td>Hello ,<td><td> <td>World !<td><<endl ;
    cout << <td>你真聪明 ,你已经会用 C ++ 编写程序了 !<td><<endl ;
}
// 函数 SayGoodbye 定义
void SayGoodbye()
{
    cout << <td>bye ,<td><td> <td>朋友 !<td><< endl ;
}
```

(远) 添加代码完成后 ,打开  菜单 ,选择  命令保存  文件。

## 摇员猿猿 编译并运行程序摇

编辑好了程序 ,在  菜单下选择  命令 ,进行编译和连接 ,如果程序没有错误 ,就建成了可以运行的  文件。

在 月 菜单下选择 耘 命令 ,运行 文件 ,执行的结果为 :

Hello ,World !

你真聪明 ,你已经会用 C ++ 编写程序了 !

Bye ,朋友 !

做到这里 ,已经完成了第一个 应用程序。

## 员 的注释

### 员的注释

程序包含了一些注释 ,使用两个反斜杠开始一个注释行。在两个反斜杠后的任何字符都是注释的一部分。如果需要多行的注释 ,则每一行都应由两个反斜杠开始。也可以在 语句后插入注释。

在 程序中注释有 :

```
//Hello.cpp :Defines the entry point for the console application.  
// 函数原型  
////////////////////////////////////  
// 函数 SayHello 定义  
// 函数 SayGoodbye 定义
```

注释可以帮助编程人员阅读程序 ,它不影响程序的运行结果。

### 预处理指令

程序开始处 ,紧接着注释的是一个 指令 ,它是 使用的预处理器。预处理器是在编译器运行前执行的程序 ,目的是使用已经定义的函数、类、变量以及其他代码元素。在 程序的 语句中包括了两个文件 ,和 用 “ 约跃”将 括起来表明这个文件是 中的文件。预处理器知道此类文件的位置。用双引号 “ ” 将 括起来说明这个文件是用户程序的一部分。

### 关键字

保留了 一些供自己使用的标识符 ,称为关键字。例如 ,和 是关键字。关键字有固定的特殊用途 ,不能将关键字作为变量、函数等的名称。

### 字符串

字符串用双引号括起来 ,如 和 等都是字符串。

### 函数

函数是一个模块 ,可以在程序中多次被调用。

在 程序中 :