

主 编 连卫民
副主编 方党生 周溢辉



C语言上机实训及习题解答



电子工业出版社
PUBLISHING HOUSE OF ELECTRONICS INDUSTRY

<http://www.phei.com.cn>

C 语言上机实训及习题解答

主 编 连卫民

副主编 方党生 周溢辉

電子工業出版社·

Publishing House of Electronics Industry

北京·BEIJING

内 容 简 介

本书突出了 C 语言的基本概念和主要知识点, 注重提高上机实训能力及程序设计能力。全书共 12 章, 内容包括: 程序设计概述、C 语言的基本知识、顺序结构的程序设计、选择结构的程序设计、循环结构的程序设计、数组的程序设计、模块化的程序设计、指针的程序设计、构造的数据类型、编译预处理与位运算、文件的程序设计、图形的程序设计。每章安排四部分内容: 主要知识点、上机实训、习题解答、上机实训分析。所有程序都在 Turbo C 2.0 集成环境下调试通过。

本书是《基于 C 语言程序设计》的配套教材, 可作为各类高等院校计算机专业和非计算机专业基础课程 C 语言的辅助教材, 也可作为计算机爱好者自学的实践参考书。

未经许可, 不得以任何方式复制或抄袭本书之部分或全部内容。
版权所有, 侵权必究。

图书在版编目 (CIP) 数据

C 语言上机实训及习题解答 / 连卫民主编. —北京: 电子工业出版社, 2009.3
ISBN 978-7-121-08405-8

I. C… II. 连… III. C 语言—程序设计—自学参考资料 IV. TP312

中国版本图书馆 CIP 数据核字 (2009) 第 029281 号

责任编辑: 祁玉芹

印 刷: 北京市天竺颖华印刷厂

装 订: 三河市鑫金马印装有限公司

出版发行: 电子工业出版社

北京市海淀区万寿路 173 信箱 邮编 100036

开 本: 787×1092 1/16 印张: 11 字数: 268 千字

印 次: 2009 年 3 月第 1 次印刷

印 数: 4000 册 定价: 19.50 元

凡所购买电子工业出版社图书有缺损问题, 请向购买书店调换。若书店售缺, 请与本社发行部联系, 联系及邮购电话: (010) 88254888。

质量投诉请发邮件至 zltz@phei.com.cn, 盗版侵权举报请发邮件至 dbqq@phei.com.cn。

服务热线: (010) 88258888。

前 言

随着计算机的普及,各行业对计算机应用人才的要求也在不断提高,不仅要能操作计算机,还要会编写一些实用程序。在众多的程序设计语言中,C语言具有功能丰富、表达力强、使用灵活、程序高效、应用面广等特点,它既具有高级语言的形式,又具有低级语言的功能。因此,C语言已被广大计算机专业人士和程序爱好者所钟情。另外,C语言作为C++的一个子集,可以为进一步学习C++和VC++打下良好的基础。基于C语言的程序设计是培养学生程序设计能力的重要课程之一,该课程对掌握基本编程方法,培养基本编程素质具有重要意义。

本书凝聚了作者多年从事C语言教学经验,采用“以用促学”的编写原则,其内容丰富,结构合理,旨在提高读者C语言程序设计的能力。本书共12章,每章由四部分组成:

(1) 主要知识点。对本章的主要概念及知识进行了总结整理。(2) 上机实训。有针对性地提出本章上机实训的目的、要求,并给出丰富的实训内容。(3) 习题解答。对本章给出的大量习题进行了分析解答。(4) 上机实训分析。对前面所提出的部分实训内容进行分析,并给出参考答案。本书所有程序都在Turbo C 2.0下调试通过。

参与本书编写的教师都是多年来从事C语言课程教学的一线教师,他们积累了丰富的知识和教学经验。本书由连卫民担任主编,制定编写大纲,并负责统稿和定稿工作,方党生和周溢辉为本书的副主编,其他参与编写本书的人员还有刘彦楠、张万仑、杜云和张兵,其中连卫民编写本书的第1章至第3章,刘彦楠编写本书的第4章和第5章。

由于作者水平有限,书中不当之处,敬请专家、读者批评指正。作者的E-mail为lianweimin2001@sina.com。

编著者
2008年12月

目 录

第 1 章 程序设计概述	1
1.1 主要知识点	1
1.1.1 基本概念	1
1.1.2 基本知识	2
1.2 上机实训	3
1.2.1 实训一：简单的 C 程序	3
1.2.2 实训二：带子函数的 C 程序	4
1.3 习题解答	5
1.3.1 名词解释	5
1.3.2 填空题	6
1.3.3 简答题	6
1.4 上机实训分析	7
1.4.1 实训一：简单的 C 程序	7
1.4.2 实训二：带子函数的 C 程序	7
第 2 章 C 语言的基本知识	9
2.1 主要知识点	9
2.1.1 基本概念	9
2.1.2 基本知识	9
2.2 上机实训	11
2.2.1 实训一：数据类型	11
2.2.2 实训二：数据运算	12
2.3 习题解答	13
2.3.1 选择题	13
2.3.2 填空题	15
2.3.3 写出下列程序的运行结果	16
2.4 上机实训分析	17
2.4.1 实训一：数据类型	17
2.4.2 实训二：数据运算	18
第 3 章 顺序结构的程序设计	19
3.1 主要知识点	19
3.2 上机实训	20
3.2.1 实训一：输入/输出函数的格式练习	20
3.2.2 实训二：简单计算程序的设计	21
3.3 习题解答	22
3.3.1 单选题	22
3.3.2 写出下列程序的运行结果	24
3.3.3 编程题	25
3.4 上机实训分析	26
3.4.1 实训一：输入/输出函数的格式练习	26

3.4.2 实训二：简单计算程序的设计	27
第 4 章 选择结构的程序设计	29
4.1 主要知识点	29
4.1.1 基本概念	29
4.1.2 基础知识	29
4.2 上机实训	31
4.2.1 实训一：用 if 语句实现选择结构程序设计	31
4.2.2 实训二：用 switch 语句实现选择结构程序设计	31
4.3 习题解答	32
4.3.1 单选题	32
4.3.2 编程题	35
4.4 上机实训分析	38
4.4.1 实训一：用 if 语句实现选择结构程序设计	38
4.4.2 实训二：用 switch 语句实现选择结构程序设计	39
第 5 章 循环结构的程序设计	41
5.1 主要知识点	41
5.1.1 基本概念	41
5.1.2 基本知识	41
5.2 上机实训	43
5.2.1 实训一：当型循环的程序设计	43
5.2.2 实训二：直到型循环的程序设计	44
5.2.3 实训三：for 循环的程序设计	44
5.2.4 实训四：双层循环的程序设计	45
5.2.5 实训五：多层循环的程序设计	46
5.3 习题解答	47
5.3.1 单选题	47
5.3.2 编程题	49
5.4 上机实训分析	52
5.4.1 实训一：当型循环的程序设计	52
5.4.2 实训二：直到型循环的程序设计	53
5.4.3 实训三：for 循环的程序设计	54
5.4.4 实训四：双层循环的程序设计	55
5.4.5 实训五：多层循环的程序设计	56
第 6 章 数组的程序设计	57
6.1 主要知识点	57
6.1.1 基本概念	57
6.1.2 基本知识	57
6.2 上机实训	58
6.2.1 实训一：一维数组的定义与使用	58
6.2.2 实训二：二维数组的定义与使用	59
6.2.3 实训三：字符数组的定义与使用	60
6.3 习题解答	60
6.3.1 单选题	60
6.3.2 填空题	62
6.3.3 编程题	63
6.4 上机实训分析	68

6.4.1	实训一：一维数组的定义与使用	68
6.4.2	实训二：二维数组的定义与使用	68
6.4.3	实训三：字符数组的定义与使用	70
第 7 章	模块化的程序设计	71
7.1	主要知识点	71
7.1.1	基本概念	71
7.1.2	基本知识	72
7.2	上机实训	73
7.2.1	实训一：函数的定义及调用	73
7.2.2	实训二：函数的嵌套与递归	74
7.2.3	实训三：变量的存储属性	75
7.3	习题解答	76
7.3.1	单选题	76
7.3.2	填空题	78
7.3.3	写出下列程序的运行结果	79
7.3.4	编程题	81
7.4	上机实训分析	86
7.4.1	实训一：函数的定义及调用	86
7.4.2	实训二：函数的嵌套与递归	89
第 8 章	指针的程序设计	91
8.1	主要知识点	91
8.1.1	基本概念	91
8.1.2	基本知识	92
8.2	上机实训	93
8.2.1	实训一：指向变量的指针	93
8.2.2	实训二：指向函数的指针和指针函数	94
8.2.3	实训三：指向数组的指针和指针数组的定义与使用	96
8.3	习题解答	96
8.3.1	单选题	96
8.3.2	写出下列程序的运行结果	98
8.3.3	编程题	100
8.4	上机实训分析	105
8.4.1	实训一：指向变量的指针	105
8.4.2	实训二：指向函数的指针和指针函数	106
8.4.3	实训三：指向数组的指针和指针数组的定义与使用	106
第 9 章	构造的数据类型	107
9.1	主要知识点	107
9.1.1	基本概念	107
9.1.2	基本知识	107
9.2	上机实训	109
9.2.1	实训一：结构体类型	109
9.2.2	实训二：共用体类型	110
9.2.3	实训三：枚举类型	111
9.3	习题解答	111
9.3.1	单选题	111
9.3.2	写出下列程序的运行结果	112

9.3.3 编程题	114
9.4 上机实训分析	119
9.4.1 实训一: 结构体类型	119
9.4.2 实训二: 共用体类型	121
9.4.3 实训三: 枚举类型	122
第 10 章 编译预处理与位运算	123
10.1 主要知识点	123
10.1.1 基本概念	123
10.1.2 基本知识	124
10.2 上机实训	127
10.2.1 实训一: 编译预处理	127
10.2.2 实训二: 位运算	127
10.3 习题解答	128
10.3.1 单选题	128
10.3.2 填空题	129
10.3.3 编程题	131
10.4 上机实训分析	133
10.4.1 实训一: 编译预处理	133
10.4.2 实训二: 位运算	134
第 11 章 文件的程序设计	135
11.1 主要知识点	135
11.1.1 基本概念	135
11.1.2 基本知识	136
11.2 上机实训	137
11.2.1 实训一: 文件中简单数据的读/写	137
11.2.2 实训二: 文件中复杂数据的读/写	138
11.3 习题解答	138
11.3.1 单选题	138
11.3.2 填空题	141
11.3.3 编程题	142
11.4 上机实训分析	147
11.4.1 实训一: 文件中简单数据的读/写	147
11.4.2 实训二: 文件中复杂数据的读/写	148
第 12 章 图形程序设计	151
12.1 主要知识点	151
12.1.1 基本概念	151
12.1.2 基本知识	152
12.2 上机实训	156
12.2.1 实训一: 文本方式下字符图形程序设计	156
12.2.2 实训二: 图形方式下图形程序设计	157
12.3 习题解答	158
12.3.1 阅读并上机调试下列程序, 指出它们的功能效果	158
12.3.2 编程实现如下要求	161
12.4 上机实训分析	164
12.4.1 实训一: 文本方式下字符图形程序设计	164
12.4.2 实训二: 图形方式下图形程序设计	166

第 1 章 程序设计概述

本章教学目标：

- ☞ 熟悉程序设计的基本概念和程序设计的基本过程。
- ☞ 熟悉算法的描述和 C 程序的运行过程。

本章内容安排：

- ☞ 主要知识点。
- ☞ 上机实训。
- ☞ 习题解答。
- ☞ 上机实训分析。

1.1 主要知识点

1.1.1 基本概念

1. 计算机系统

计算机系统由计算机硬件和计算机软件两部分组成。计算机硬件是指组成计算机系统的设备或机器，一般包括中央处理器（CPU）、内存储器、外存储器、输入设备和输出设备。中央处理器包括运算器和控制器。计算机软件是指组成计算机系统的程序、数据和文档。

2. 程序

程序=算法+数据结构。程序就是用计算机语言描述的算法。

3. 算法

算法是为了解决某个特定问题而采取的确定且有限的步骤。算法的要素主要是操作和控制结构。即一个算法由一些操作组成，这些操作又是按照一定的控制结构所规定的次序执行的。

4. 数据结构

数据结构是相互之间存在一种或多种特定关系的数据元素的集合。定义中的“关系”描述的是数据元素之间的逻辑关系，称为逻辑结构；数据结构在计算机中的表示（映象）称为数据的物理结构，有顺序结构（如数组）和随机结构（如链表）两种；数据结构要解决的三个问题是：数据的表示、数据元素间的关系，以及在这种关系上的运算。

从计算机语言上看，数据结构体现为数据类型。数据类型是一个值的集合和定义在这个值集上的一组操作的总称，它是信息特性的抽象。

1.1.2 基本知识

1. 程序设计

程序设计是利用某种计算机语言，按照一定的算法，对特定的数据进行处理，程序设计的最终结果是用某种语言描述的计算机程序。

程序设计时应该考虑以下几个问题：数据在计算机内存中如何存储，数据如何输入，数据如何处理，数据如何输出。

程序设计的方法包括结构化程序设计方法和面向对象程序设计方法。

结构化程序设计方法可归结为“程序=算法+数据结构”，其核心是设计算法。它是一种面向过程的设计方法，数据与数据处理分离。其基本思想是采用“自顶向下、逐步求精”的设计方法和单入单出的控制结构。

面向对象的设计思想模拟人们对自然界认识和处理事物的方法，将数据和对数据的处理方法放在一起，形成一个相对独立的整体——对象，对同类型对象抽象出共性，形成类。任何类中的数据都只是用本类自有的方法进行处理，并通过简单的接口与外部联系。对象通过消息进行通信。面向对象的程序设计方法同样遵循分层抽象、逐步细化的原则。和传统的程序设计方法相比，面向对象的程序设计具有抽象性、封装性、继承性和多态性的特点。

2. 算法

算法是为了解决某个特定问题而采取的确定且有限的步骤，算法的描述工具有自然语言、流程图、N-S 图和计算机语言。

算法的功能取决于所选用的操作和控制结构。计算机的基本操作有：算术运算、关系运算和逻辑运算、数据传送即赋值运算。算法的基本控制结构有顺序结构、选择结构和循环结构三种。

算法具有以下三个特性：有穷性、确定性和输入输出。

用计算机解决问题可以分为三步：首先，把要处理的数据输入到计算机中；然后，根据处理要求对输入的数据进行处理，得到处理结果；最后，把结果显示或打印出来。

3. 程序设计评判标准

评判一个程序的好坏标准有四条：（1）正确性。指给出正确的数据，得到正确的结果。（2）易读性。指编写的程序不但自己很容易看懂（即使隔了很长时间），而且别人也很容易看懂。（3）健壮性。指容易维护，其具体内容包括：功能性扩充易于实现；不因为输入数据或使用人员的变换导致输出结果错误；程序的自纠错能力强；安全性高；发生数据输入错，程序运行错，系统有到位的提示功能。（4）可移植性。指编写的程序在不同机型的同一语言环境下仍然能够正确运行。

4. C 语言

C 语言是一种计算机高级算法语言，由于 C 语言提供了某些接近汇编语言的功能，使得 C 语言既是高级语言，又具有低级语言的特点，因此人们又把 C 语言称为中级语言。C 程序的描述都是以函数形式出现的，所以 C 语言又被称为函数式语言。

C 语言的主要优点在于适合编写系统软件，也适合于编写应用软件。C 程序的代码可以比汇编程序的代码减少 20%，且易读性和结构性都比汇编程序要好。

C 语言也存在自身的不足。学习 C 语言与学习其他高级语言相比，有碎、繁、难的特点。由于 C 语言放宽了语法检查，增大了程序编写的灵活性，同时也增大了出错的概率。

5. C 程序上机调试步骤

(1) 编辑。

① 双击 tc.exe 文件进入 C 环境。

② 选择 file→new 命令，建立新文件，开始输入源程序。

(2) 编译

按 F9 键编译源程序。如果存在错误，修改程序后再次编译和修改，直到程序编译完全正确为止。

(3) 运行

按 Ctrl+F9 组合键运行程序，按 Alt+F5 组合键看运行结果。

6. 几个常用按键介绍

(1) F10 键：将光标移动到主菜单。

(2) F2 键：保存文件。

(3) F6 键：这是一个开关键，用于使光标在编辑窗口 (Edit) 和信息窗口 (Message) 之间进行切换。

1.2 上机实训

Turbo C 2.0 是最常用的一种 C 语言调试工具，本节主要训练 Turbo C 2.0 的使用方法，以及调试程序的步骤。

1.2.1 实训一：简单的 C 程序

1. 实训目的

了解简单 C 语言程序的结构和书写规则。

2. 实训要求

(1) 从 Windows 环境进入 Turbo C。

(2) 严格按照程序的缩进格式输入程序代码 (注意使用 Tab 键)。

(3) 观察以下两个程序，体会 C 语言程序的结构和书写规则。

3. 实训内容

(1) 进入 Turbo C，并调试下列程序。

```
#include "stdio.h"          /* 文件包含 */
main()                      /* 定义主函数 */
{
```

```
    printf("Welcome to the world!\n"); /* 输出欢迎语句 */
}
```

(2) 进入 Turbo C，并调试下列程序。

```
#include "stdio.h"
main()
{
    int num1,num2,sum;      /* 定义 3 个整型变量 */
    num1=1;                 /* 赋值语句 */
    num2=99;
    sum=num1+num2;
    printf("sum=%d\n",sum); /* '\n'表示回车换行 */
}
```

1.2.2 实训二：带子函数的 C 程序

1. 实训目的

了解带子函数的较复杂 C 语言程序的结构。

2. 实训要求

- (1) 从 Windows 环境进入 Turbo C。
- (2) 严格按照程序的缩进格式输入程序代码（注意使用 Tab 键）。
- (3) 观察以下程序，体会带子函数的 C 语言程序的结构和书写规则。

3. 实训内容

(1) 下列程序的功能是求两个整数的和与差，要求计算和与差的操作分别用函数实现。

```
int add(int x,int y) /* add 函数,求两数的和 */
{
    int z;
    z=x+y;
    return z;        /* 返回 z 的值到主函数 */
}
int sub(int x,int y) /* sub 函数,求两数的差 */
{
    return x-y;      /* 返回 x-y 的值到主函数 */
}
main()
{
    int a,b,c;
    a=2;b=4;
    c=add(a,b);      /* 调用 add 函数计算两数的和 */
}
```

```

        printf("%d+%d=%d\n", a,b,c);          /* 输出和 */
        printf("%d-%d=%d\n", a,b,sub(a, b));   /* 输出差 */
    }

```

(2) 下列程序的功能是输入两个实数，求最大值。

```

#include "stdio.h"
main()
{
    float a,b,c;
    float max(float x,float y);  /* max 函数原型声明 */
    printf("Input a,b=");
    scanf("%f,%f",&a,&b);      /* 输入两个实数 */
    c=max(a,b);                 /* 调用函数 max, 求 a、b 最大者 */
    printf("bigger: %f\n",c);    /* 输出 c */
}

float max(float x, float y)     /* 定义 max 函数 */
{
    float z;
    if (x>y)                    /* 比较 x 和 y 值, 较大者赋值给 z */
        z=x;
    else
        z=y;
    return z;                  /* 返回 z 的值 */
}

```

程序运行结果（输入两个数）如下：

```

Input a,b=11.2,22.81✓
bigger: 22.8108000

```

以上程序包含了 C 语言的基本要素。学习一门程序设计语言的唯一途径就是使用它编写程序，读者在开始的时候可以在分析简单实例的基础上，通过模仿，编写出自己的程序。

1.3 习题解答

1.3.1 名词解释

1. 计算机硬件

计算机硬件是指组成计算机系统的设备或机器，是“看得见，摸得着”的物理部件，它是组成计算机系统的基础。计算机硬件一般包括中央处理器（CPU）、内存储器、外存储器、输入设备和输出设备。

2. 计算机软件

计算机软件是指组成计算机系统的程序、数据和文档。程序是指令的有序集合；数据是信息在计算机中的表示，是计算机处理的对象；文档是各种说明文本，是软件操作的辅助性资源。

3. 程序设计

程序设计就是利用某种计算机语言，按照一定的算法，对特定的数据进行处理。程序设计的最终结果是用某种语言描述的计算机程序。

4. 计算机数据

能被计算机接收和处理的信息就是计算机数据。

1.3.2 填空题

- (1) 程序 = 算法 + 数据结构。
- (2) 算法 是为解决某个特定问题而采取的确定的有限步骤。
- (3) “结构化程序设计方法”是荷兰学者 迪克特拉(Dijkstra) 等人提出来的。
- (4) 用 Turbo C 编辑产生的结果文件的扩展名是 .C，编译产生的目标文件的扩展名是 .OBJ，连接产生的文件的扩展名是 .EXE。
- (5) 用 Turbo C 运行程序的组合键是 Ctrl+F9，查看结果的组合键是 Alt+F5。

1.3.3 简答题

1. 一个好程序的标准是什么？

好程序的标准有四条：(1) 正确性。指给出正确的数据，得到正确的结果。(2) 易读性。指编写的程序不但自己很容易看懂（即使隔了很长时间），而且别人也很容易看懂。(3) 健壮性。指容易维护，其具体内容包括：功能性扩充易于实现；不因为输入数据或使用人员的变换导致输出结果错误；程序的自纠错能力强；安全性高；发生数据输入错，程序运行错，系统有到位的提示功能。(4) 可移植性。指编写的程序在不同机型的同一语言环境下仍然能够正确运行。

2. 简述算法的基本特性。

算法有以下 3 个特性：

- (1) 有穷性。一个算法通过若干步骤即可实现预定目标，算法的每一步都可以在合理的时间内完成。
- (2) 确定性。算法中的每一条指令都必须有确切的含义，同样的步骤、同样的条件结果也是相同的。
- (3) 输入和输出。首先，把要处理的数据输入到计算机中；然后，根据处理要求对输入的数据进行处理，得到处理结果；最后，把结果显示或打印出来。

3. 简述一个 C 程序上机调试的过程。

(1) 编辑。

① 双击 tc.exe 文件进入 C 环境。

② 选择 file→new 命令，建立新文件，开始输入源程序。

(2) 编译。

按 F9 键编译源程序。如果存在错误，修改程序后再次编译和修改，直到程序编译完全正确为止。

(3) 运行。

按 Ctrl-F9 组合键运行程序，按 Alt-F5 组合键看运行结果。

4. 简述计算机系统的组成。

计算机系统由计算机硬件和计算机软件组成。计算机硬件是指组成计算机系统的设备或机器，是“看得见，摸得着”的物理部件，它是组成计算机系统的基础。计算机硬件一般包括中央处理器（CPU）、内存储器、外存储器、输入设备和输出设备。

1.4 上机实训分析

1.4.1 实训一：简单的 C 程序

实训内容（1）：进入 Turbo C，并调试下列程序。

运行结果如下：

```
Welcome to the world!
```

实训内容（2）：进入 Turbo C，并调试下列程序。

运行结果如下：

```
sum=100
```

1.4.2 实训二：带子函数的 C 程序

实训内容（1）：下列程序的功能是求两个整数的和与差，要求计算和与差的操作分别用函数实现。

运行结果如下：

```
2+4=6
```

```
2-4=-2
```

实训内容（2）：下列程序的功能是输入两个实数，求最大值。

运行结果如下：

```
Input a,b=2.5,6.8  
bigger:6.800000
```

```
Input a,b=2.5,6.8✓  
bigger: 6.800000
```

程序运行时，必须由用户输入数据 2.5,6.8，然后按回车键，程序才能执行。



第2章 C语言的基本知识

本章教学目标：

- ☞ 熟悉C语言中数据的表示。
- ☞ 掌握3种基本类型数据的使用。
- ☞ 掌握基本运算符及表达式的使用。

本章内容安排：

- ☞ 主要知识点。
- ☞ 上机实训。
- ☞ 习题解答。
- ☞ 上机实训分析。

2.1 主要知识点

本章主要介绍了C语言的数据类型，以及基本类型数据的表示，并详细介绍了基本运算符及表达式的使用。

2.1.1 基本概念

1. 常量

常量是指在程序的执行过程中，其值不能发生变化的量和具有固定值的量。

2. 变量

数据被存储在一定的内存单元中，数据连同存储它的内存单元被抽象为变量，每个变量都有一个名字，这个名字被称为变量名。它代表了某段存储单元及其所存储的数据。

3. 表达式

表达式描述了对哪些数据，以什么顺序施进行什么样的操作，它由运算符和数据组成。

2.1.2 基本知识

1. 数据类型

C语言中的数据分为以下几种类型：基本类型（整型、实型、字符型）、构造类型（数组、结构体、共用体、枚举类型）、指针类型和空类型。

2. 标识符的命名规则

- (1) 由字母、数字和下划线组成，且以字母或下划线开头。
- (2) 标识符的长度无统一规定，最多可取到32个字符。考虑到程序的可移植性，建