

云南大学数学与统计学实验教学中心

《高级语言程序设计》实验报告

课程名称：程序设计和算法语言	学期：2016~2017 学年上学期	成绩：
指导教师：赵越	学生姓名：刘鹏	学生学号：20151910042
实验名称：简单程序设计调试		
实验编号：No.01	实验日期：2017 年 1 月 16 日	实验学时：2
学院：数学与统计学院	专业：信息与计算科学	年级：2015 级

一、 实验目的

1. 熟悉 TC 或 Borland C++的运行环境。
2. 了解在 TC 或 Borland C++上如何编辑、编译、连接和运行一个 C 程序。
3. 熟练使用各种 C 语言的数据类型、运算符与表达式。
4. 熟练掌握格式输入与格式输出函数的使用。
5. 通过运行简单的 C 程序，初步了解 C 源程序的特点,初步掌握查错和排错的技能.
6. 学会 Turbo C 2.0 的安装方法（参看本书第一部分），熟悉 C 语言程序的运行环境，了解所用计算机系统软、硬配置。
7. 初步了解在该集成环境下如何编辑、编译、连接和运行一个 C 程序，即运行一个 C 程序的全过程。
8. 通过运行简单的 C 程序，初步了解 C 程序的基本结构及特性。

二、 实验环境

Windows10 Pro Workstation 17096;
Code::Blocks 16.01 GCC 集成开发环境;
Cygwin GCC 编译器。

三、 实验内容

1.1 题 1

从开机开始进行操作，熟悉一些常用的 DOS 命令，包括如何建立子目录，拷贝文件，删除文件等。

Solution

标准 C 总是与 UNIX 系统关联在一起，为了使用符合国际标准的 C 编译器，这里引入 Cygwin 这套软件，它可以提供一个符合 POSIX 标准的运行时环境，同时使用很多 UNIX 环境下的软件，比如这里使用的 GCC 编译器与 Vim 编辑器。

包括本篇在内所有实验报告均采用时下最新的 GCC 编译器进行编译，其他组织或者企业实现的编译器，如 Visual C++等均不在采用之列。也可以采用 MinGW，它主要提供了针对 win32 应用的 GCC、GNU binutils 等工具，以及对等于 Windows SDK 的头文件和用于 MinGW 版本 linker 的库文件，而不是 Visual C 的 lib。

```
~/C_Program_Design
Newton@Newton-PC-2 ~
$ pwd
/home/Newton

Newton@Newton-PC-2 ~
$ mkdir C_Program_Design

Newton@Newton-PC-2 ~
$ cd C_Program_Design/

Newton@Newton-PC-2 ~/C_Program_Design
$ vim Hello_world.c

Newton@Newton-PC-2 ~/C_Program_Design
$ cat Hello_world.c
#include <stdio.h>

int main(){
    printf("Hello, world!\n");
    return 0;
}

Newton@Newton-PC-2 ~/C_Program_Design
$ gcc Hello_world.c -o test

Newton@Newton-PC-2 ~/C_Program_Design
$ ./test.exe
Hello, world!

Newton@Newton-PC-2 ~/C_Program_Design
$
```

其中，Vim 输入部分为

```
~/C_Program_Design
#include <stdio.h>

int main(){
    printf("Hello, world!\n");
    return 0;
}
-- 插入 --
1,1 全部
```

1.2 题 2

建立自己的子目录，以备存放文件。

Solution

根据上一题，已经建立了一个文件夹用来存放代码。以后的代码文件名将与题目联系。这里用命令 `mv Hello_world.c "1.1 Hello World.c"` 进行了重命名。生成的可执行文件根据需要进行及时清理。

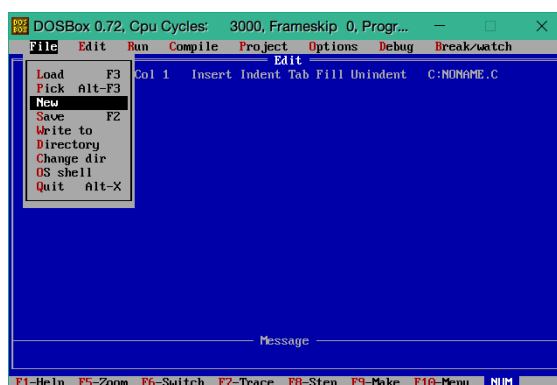
1.3 题 3

进入 Turbo C 集成环境，熟悉 Turbo C 主菜单下各选择项的功能及功能键的使用。

(1) 启动 TC 集成环境

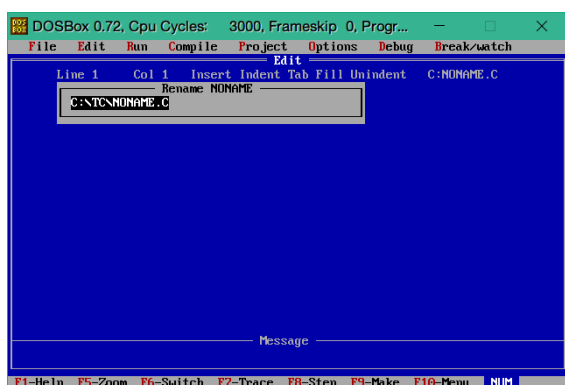


(2) 编辑一个新文件

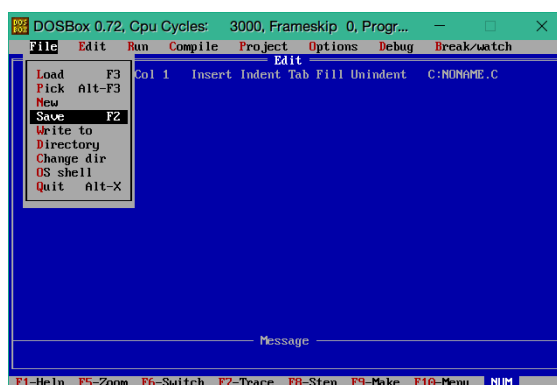


按 F10 将亮条移到 File 处按回车键，再选 New 后按回车键，为简化，以后用 File→New 的形式表示上述操作。

(3) 文件命名



(3) 保存文件

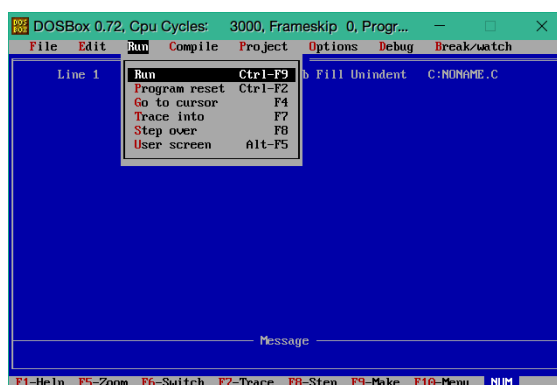


出现子窗口后，在其中输入要创建的文件名，输入文件名后按回车键，出现编辑窗口，即可在此窗口中输入和修改源程序。

(4) 编译文件



(5) 运行文件



按 F10，选择菜单 Compile→Compile to obj，就可以对程序进行编译。在此过程中可能会出现错误信息，并进行修改，修改完后存盘再进行编译，直至成功。

按 F10，选择 Run→Run 命令运行文件。

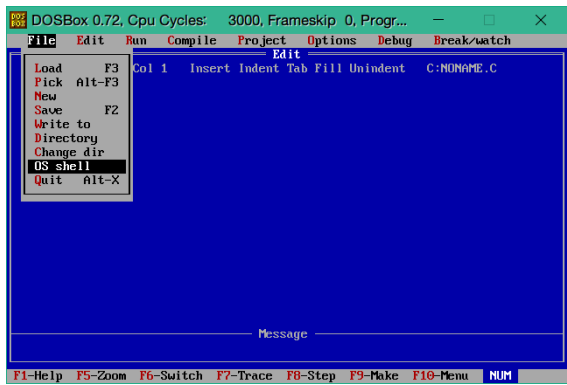
(6) 查看运行结果



按 F10，选择菜单 Run→User screen 即可以查看程序运行结果。注意：若编译没有成功，则程序不能运行；若编译成功，则可以运行程序，但若运行结果不正确，则表明程序有逻辑错误，需要对程序进行调试和修改，直至程序运行结果正确。

以后每次上机运行一个 C 程序都要经过上述操作步骤②—⑥，因此，需要深刻理解和熟练掌握这些步骤。

3. 学会退出 Turbo C



暂时退出：按 F10，选择菜单 File→OS shell。此时按 exit 即可以返回主菜单。

1.4 题 4

输入一个简单 C 程序（可用教科书上的例题），了解 C 程序运行的全过程。

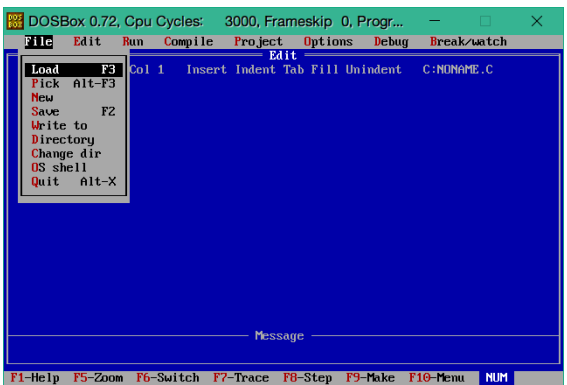
1.5 题 5

编写用 printf 语句将 3 个字符串：good morning, floppy disk, hard disk 在同一行显示的程序。

1.5.1 程序示例

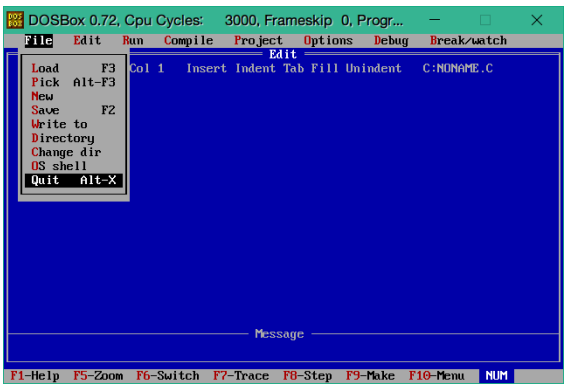
```
1 /*
2  * filename: 1.2 example print.c
3  * property: example
```

2. 学会调入一个已存在的 C 源文件



按 F10，选择菜单 File→Load，出现子窗口，在其中输入已存在的文件名，按回车键后即可以调入一个已存在的文件。

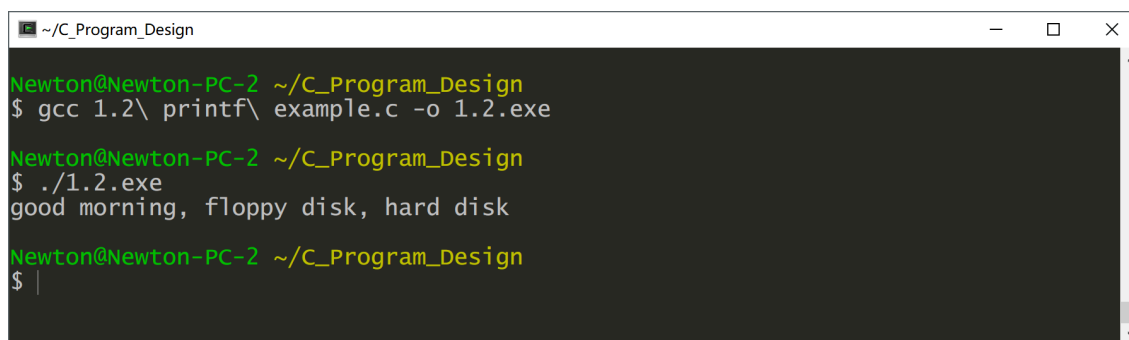
永久退出：按 F10，选择菜单 File→Quit。



```
4  */
5  #include<stdio.h>
6  int main()
7  {
8      printf ("good morning, ");
9      printf ("floppy disk, ");
10     printf ("hard disk\n");
11     return 0;
12 }
```

程序 1

1.5.2 运行结果



```
~/C_Program_Design
Newton@Newton-PC-2 ~/C_Program_Design
$ gcc 1.2\ printf\ example.c -o 1.2.exe

Newton@Newton-PC-2 ~/C_Program_Design
$ ./1.2.exe
good morning, floppy disk, hard disk

Newton@Newton-PC-2 ~/C_Program_Design
$ |
```

运行结果 1

1.6 题 6

把上面的程序改为每行显示一个字串，应如何修改程序，并运行之。

1.6.1 程序代码

```
1  /*
2  * filename: 1.3 print by line.c
3  * property: example
4  */
5  #include<stdio.h>
6  int main()
7  {
8      printf ("good morning\n");
9      printf ("floppy disk\n");
10     printf ("hard disk\n");
11     return 0;
12 }
```

1.6.2 运行结果

```
~/C_Program_Design
Newton@Newton-PC-2 ~/C_Program_Design
$ gcc 1.3\ print\ by\ line.c -o 1.3.exe

Newton@Newton-PC-2 ~/C_Program_Design
$ ./1.3.exe
good morning
floppy disk
hard disk

Newton@Newton-PC-2 ~/C_Program_Design
$ |
```

1.7 题 7

编写一程序，用键盘输入语句输入三个数，然后分别求它们的和与积。

1.7.1 程序代码

```
1  /*
2  * filename: 1.4 simple calculation.c
3  * property: homework
4  */
5  #include<stdio.h>
6  int main()
7  {
8      int x, y, z;
9      printf("please input 3 numbers:\n");
10     scanf("%d %d %d", &x, &y, &z);
11     printf("a + b + c = %d\n", x+y+z);
12     printf("a * b * c = %d\n", x*y*z);
13     return 0;
14 }
```

程序 2

1.7.2 运行结果

```
~/C_Program_Design
Newton@Newton-PC-2 ~/C_Program_Design
$ gcc 1.4\ simple\ calculation.c -o 1.4.exe

Newton@Newton-PC-2 ~/C_Program_Design
$ ./1.4.exe
please input 3 numbers:
3 6 9
a + b + c = 18
a * b * c = 162

Newton@Newton-PC-2 ~/C_Program_Design
$ |
```

1.8 题 8

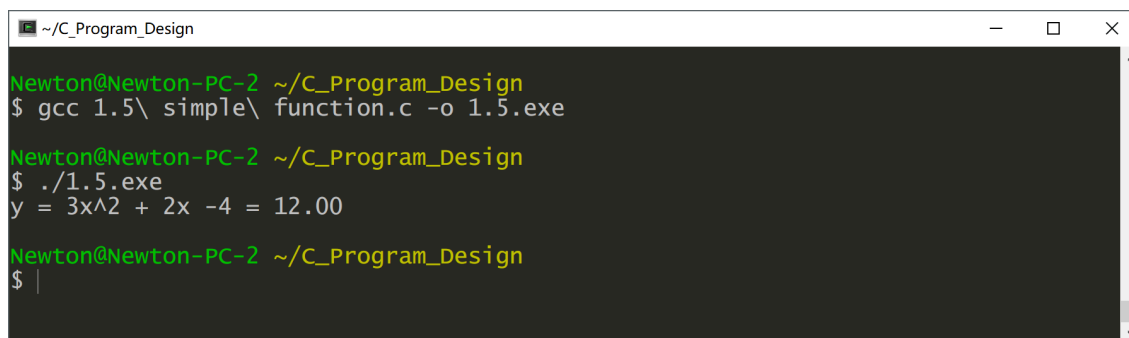
计算函数 $y = 3x^2 + 2x - 4$ (假设 $x = 2$)。

1.8.1 程序代码

```
1  /*
2  * filename: 1.5 simple function.c
3  * property: example
4  */
5
6  #include <stdio.h>
7  #include <math.h>
8  int main() {
9      double x = 2;
10     double y;
11     y = 3 * pow(x, 2) + 2 * x - 4;
12     printf("y = 3x^2 + 2x -4 = %1.2f\n", y);
13     return 0;
14 }
```

程序代码 3

1.8.2 运行结果



```
~/C_Program_Design
Newton@Newton-PC-2 ~/C_Program_Design
$ gcc 1.5\ simple\ function.c -o 1.5.exe

Newton@Newton-PC-2 ~/C_Program_Design
$ ./1.5.exe
y = 3x^2 + 2x -4 = 12.00

Newton@Newton-PC-2 ~/C_Program_Design
$ |
```

1.9 题 9

设一个正圆台的上底半径 $r_1 = 2.5\text{cm}$ ，下底半径 $r_2 = 3.7\text{cm}$ ，高 $h = 3.5\text{cm}$ 。分别计算并输出该圆台的上底面积 s_1 下底面积 s_2 ，侧面积 s_3 ，圆台的体积 V 。

具体要求如下：

- (1) r_1 , r_2 , h 用 `scanf` 函数输入，且在输入前要有提示；
- (2) 在输出结果时要有文字说明，每个输出值占一行，且小数点后取 4 位数字。
- (3) 所有变量均定义为单精度类型。

1.9.1 程序代码

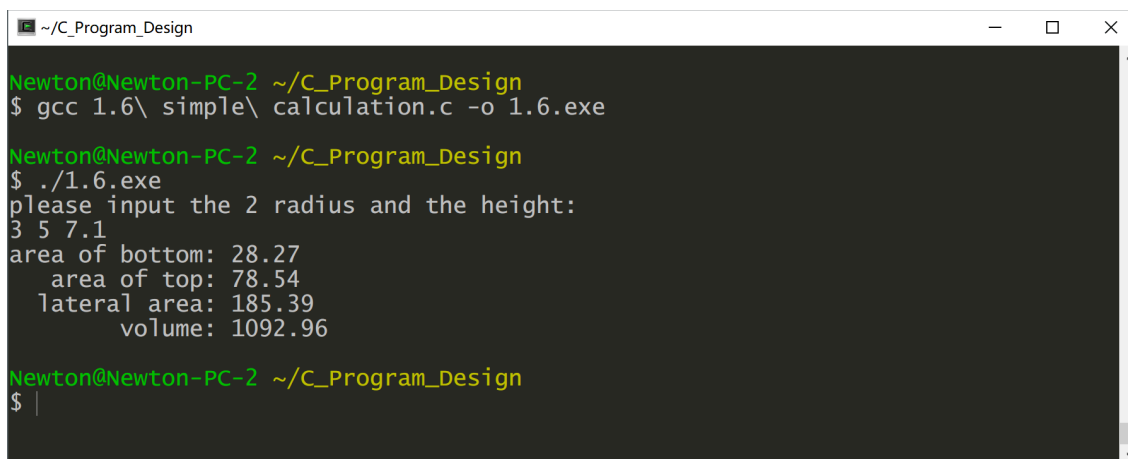
```
1  /*
2  * filename: 1.6 simple calculation.c
3  * property: homework
4  */
```

```

5
6  #include <stdio.h>
7  #include <math.h>
8
9  #define Pi 3.14159265
10
11 int main() {
12     float r1, r2, h, S1, S2, S3, V;
13     printf("please input the 2 radius and the height:\n");
14     scanf("%f %f %f",&r1, &r2, &h);
15     S1=Pi * pow(r1, 2);
16     S2=Pi * pow(r2, 2);
17     S3=Pi * (r1 + r2) * sqrt(pow(r1 - r2, 2) + pow(h, 2));
18     V=Pi * h * (pow(r1, 2) + pow(r2, 2) + r1 * r2);
19     printf("area of bottom: %-3.2f\n", S1);
20     printf("   area of top: %-3.2f\n", S2);
21     printf("  lateral area: %-3.2f\n", S3);
22     printf("          volume: %-3.2f\n", V);
23     return 0;
24 }

```

1.9.2 运行结果



```

~/C_Program_Design
Newton@Newton-PC-2 ~/C_Program_Design
$ gcc 1.6\ simple\ calculation.c -o 1.6.exe

Newton@Newton-PC-2 ~/C_Program_Design
$ ./1.6.exe
please input the 2 radius and the height:
3 5 7.1
area of bottom: 28.27
area of top: 78.54
lateral area: 185.39
volume: 1092.96

Newton@Newton-PC-2 ~/C_Program_Design
$ |

```

1.10 题 10

编程实现：用 1 分、5 分、10 分、25 分的硬币凑成任何给定的钱数 x ，要求硬币的数目最少。

1.10.1 程序代码

```

1  /*
2  * filename: 1.7 coin exchange.c
3  * property: homework
4  */
5
6  #include<stdio.h>
7
8  int main()
9  {

```



```

10     int total;
11     int k_1,k_2,k_3,k_4;
12     printf("please input the total money:\n");
13     scanf("%d", &total);
14     k_1 = total / 25;
15     total = total - k_1 * 25;
16     k_2 = total / 10;
17     total = total - k_2 * 10;
18     k_3 = total / 5;
19     total = total - k_3 * 5;
20     k_4 = total;
21     printf("need %3d 25_cent\n",k_1);
22     printf("need %3d 10_cent\n",k_2);
23     printf("need %3d 5_cent\n",k_3);
24     printf("need %3d 1_cent\n",k_4);
25     return 0;
26 }

```

1.10.2 运行结果

```

~/C_Program_Design
Newton@Newton-PC-2 ~/C_Program_Design
$ gcc 1.7\ coin\ excange.c -o 1.7.exe
Newton@Newton-PC-2 ~/C_Program_Design
$ ./1.7.exe
please input the total money:
66
need  2 25_cent
need  1 10_cent
need  1 5_cent
need  1 1_cent
Newton@Newton-PC-2 ~/C_Program_Design
$

```

1.11 C 基础习题

1.12

以下各标识符中，哪些是合法的用户标识符？

(1) main	(11) \$100
(2) MAIN	(12) I am
(3) a205	(13) max(5)
(4) _exp	(14) begin
(5) a_b	(15) #define
(6) 3.5	(16) NONAME00
(7) A[10]	(17) proc
(8) A.name	(18) and
(9) %d	(19) not
(10) float	(20) WHile

注：字母或下划开始的后跟若干字母或下划线或数字组成的序列；不能和保留字相同；避免和系统函数名相同；大小写字母有别！

2. 选择

(1) `18/4*sqrt(4.0)/8` 的值的类型？ (B)

A. int B. float C. char D. 不确定

(2) 在 C 中，一个 `unsigned int` 型数据的表示范围是 (D)

A. 0-127 B. 0-225 C. 0-32767 D. 0-65535

(3) `(-15)%(-8)` 的值是 (B)

A. -7 B. 7 C. 1.875 D. 非法

3. 下列常量哪一组全是合法的？ (1)

(1) 288, -079, 3.4e-2, 'A', 10111

(2) -0, 1e14, .5678, 'ABC', oxabc

(3) +1, 3.e-3, '4', ox5a, .0

(4) 15, 16.8e+3, 069, 101, oxabcd

(5) 2L, 345e8, '\n', '\\', '\\101'

4. 下列常量中哪一组都是非法的？ ()

(1) 0a, e1.5, ox7, '.', 158

(2) 088, ox9afg, 65538, 3E24, '\089'

(3) 32768, 1.oe584, 3.5e-476, "A", 9FBA

(4) -32769, 45678L, 12e-2.1, OX9ab, '\063'

(5) 9abH, 177Q, 07777, -9999.999, '\t'

5. 计算下列表达式

(1) `15+30%4` 17

(2) `1.5+15/2` 8.5

(3) `10==9+1` 1

(4) `x=10, 3+8, y=20` 赋值语句与空语句

(5) `x=y=500` 赋值语句

(6) 设 `x=1, y=2`，求 `x++`，`y++` 两个变量的数值不变

(7) `5>3` 1

(8) `10<500` 1

(9) `5>3&&5>8` 0

(10) `10>5||5<10` 1

四、 实验总结

随着时间的推移，我的编译器选择历经轮转，从最初的古老的 TC2.0，到 Code::Blocks 集成开发环境，

再到 Visual Studio 2017，最终还是回到了 GNU 平台上来，使用开源的一套库进行实验。在此期间，云南大学也从一个普通的 211 大学跻身双一流大学行列，高级语言程序设计这门课程是否也该升级一下？TC2.0 这个编译器，界面十分古朴，在几十年前绝对算是一流的软件，但是现在，确实落后了，这主要是因为它无法编译在 64 位系统下运行的程序。但是 TC2.0 有很多的优势，比如完全可视化的编译过程，不会生成很多附加文件，这一点 Visual Studio 就太过专业化。经过对《UNIX 环境高级编程》[1]这本书的学习，还有诸如 *Harley Hahn's Guide to Unix and Linux*[2]这本书的阅读，我觉得基于 Shell 的 UNIX 环境似乎是最适合新手学习的。

本次实验，集中主要精力，在以前版本的基础上，对文档结构进行了重整，看起来自然了很多，目录也规范了很多。有关编程的规范性问题，参考林锐高质量 C/C++编程指南的第一版[3]。

五、 参考文献

1. Stevens, W.R. and S.A. Rago, *UNIX 环境高级编程*. 2nd ed. 2005, 北京: 人民邮电出版社.
2. Hahn, H., *Harley Hahn's Guide to Unix and Linux*. 2009, New York: McGraw-Hill.
3. 林锐, *高质量 C++/C 编程指南*. 1.0 ed. 2001.

六、 教师评语