

# 云南大学软件学院

## 实验报告

### 高级语言程序设计

学 期：\_\_\_\_\_年秋季学期

指导教师：\_\_\_\_\_

指导教师：\_\_\_\_\_

姓 名：\_\_\_\_\_

学 号：\_\_\_\_\_

专 业：\_\_\_\_\_

班 级：\_\_\_\_\_

综合成绩：\_\_\_\_\_

# 实验报告说明

软件学院实验中心

- 实验名称** 要用最简练的语言反映实验的内容。如验证某程序、定律、算法，可写成“验证×××”；分析×××。
- 实验目的** 目的要明确，要抓住重点，可以从理论和实践两个方面考虑。在理论上，验证定理、公式、算法，并使实验者获得深刻和系统的理解，在实践上，掌握使用实验设备的技能技巧和程序的调试方法。一般需说明是验证型实验还是设计型实验，是创新型实验还是综合型实验。
- 实验内容（算法、程序、步骤和方法）** 这是实验报告极其重要的内容。这部分要写明依据何种原理、定律算法、或操作方法进行实验，要写明经过哪几个步骤。还应该画出流程图（实验装置的结构示意图），再配以相应的文字说明，这样既可以节省许多文字说明，又能使实验报告简明扼要，清楚明白。如果报告内容写不完，可以后加第二种格式的附页。
- 数据记录和计算** 指从实验中测出的数据以及计算结果。
- 结论（结果）** 即根据实验过程中所见到的现象和测得的数据，作出结论。也可写上实验成功或失败的原因，实验后的心得体会等。
- 注意：
- 实验报告将记入实验课成绩；
  - 实验结束时，将本实验报告册交给实验指导教师，否则将扣除此次实验成绩；
  - 本报告册附录为高级语言程序设计实验指导手册。

**实验报告填写示例**实验二名称: printf()练习实验目的: 1. 熟悉 printf()显示变量值, 表达式结果的功能 2.编写一个顺序计算的程序,要求选写算法, 再写流程图或伪代码, 最终调试完成程序。

日期: 2007 年 10 月 9 日

本次实验成绩:

指导教师签名:

**(一) 实验任务:**

显示按指定格式显示 1—23 个表达式的值, 并记录结果, 完成总结及报告。

**(二) 实验方法: 应用 printf()函数, 显示 1—23 表达式的值。**

显示为十进制整数的控制字符串为: %d

显示为小数的控制字符串为: %8.2f

显示为字符的控制字符串为: %c

**(三) 实验结果记录: (以表述清楚为目的)****1—8 显示为十进制整数;****控制格式****记录结果**

3+4\*6

XXXXXXXX

XXXXXXXX

3\*4/6+6

2\*3/12\*8/4

10\*(1+7\*3)

20-2/6+3

20-2/(6+3)

(20-2)/6+3

(20-2)/(6+3)

**9—16 显示为小数, 保留小数 2 位, 宽度为 8;**

3.0+4.0\*6.0

3.0\*4.0/6.0+6.0

2.0\*3.0/12.0\*8.0/4.0

10.0\*(1.0+7.0\*3.0)

20.0-2.0/6.0+3.0

20.0-2.0/(6.0+3.0)

(20.0-2.0)/6.0+3.0

(20.0-2.0)/(6.0+3.0)

**17—20 显示为字符;**

'm'-5

'm'+5

'G'+6

'G'-6

**21—23 显示为十进制整数。**

'b'-'a'

'g'-'a'+1

'G'-'A'+1

**(四) 实验小结：**

1. 基本数据类型：int, float, double, char
2. 各类型的显示控制字符串为：%d, %f, %lf, %c；在该练习项目中，练习了整型、浮点类型主要是 float 类型、字符类型的显示。
3. 显示格式的控制字符还有：
  - a) +：显示数字的符号
  - b) -：设置数字显示为左对齐方式显示
  - c) 数字：设置数字显示的最小宽度及小数位的精度
    - i. 整数部分：数字显示包括小数点所占用的字符最小宽度；
    - ii. 小数部分：控制小数点后的小数精度
4. 表达式计算的小结
5. /运算符的结果，在整型表达式和浮点数的表达式中的计算值不同
6. 字符类型的数据可以当作整数或字符运算
7. 练习中的问题及分析
  - a) XXXXXX
  - b) XXXXXX

**(五) 额外练习和测试写在最后，前提是你完成了要求的实验内容以后，还有剩余的时间：**

1. 完成的任务简介
2. 任务的记录
3. 结果分析和总结。

**(六) 填写报告的提示：**

总结的思路和方法

1. 先总结自己应用的知识点
2. 再总结自己对知识点的理解和分析，根据实验结果和记录
3. 最后总结实验中常出现的错误和问题

**(七) 完成实验任务的建议：**

1. 事先做好准备工作，不要到了实验室还不清楚自己的工作是什么。
2. 提前先完成实验，及报告不要在实验室一次完成。
3. 实验时间中指导老师的主要工作是检查实验完成情况，审阅报告，并回答或解决大家的问题。

日期:            年    月    日

本次实验成绩: \_\_\_\_\_ 指导教师签名: \_\_\_\_\_









实验二名称： 第 2 次实验 (printf() 练习)

---

实验目的： \_\_\_\_\_

---

日期：            年        月        日

本次实验成绩：

指导教师签名：

---







实验三名称： 第3次实验（scanf()练习）

实验目的：

日期：            年        月        日

本次实验成绩：

指导教师签名：

---









实验四名称：第 4 次实验（控制流）

---

实验目的：

---

日期：            年        月        日

本次实验成绩：

指导教师签名：

---







实验五名称：第 5 次实验（函数-1）

---

实验目的：\_\_\_\_\_

---

日期：        年    月    日

本次实验成绩：

指导教师签名：

---









实验六名称： 第 6 次实验（函数-2）

实验目的： \_\_\_\_\_

日期：            年        月        日

本次实验成绩： \_\_\_\_\_ 指导教师签名： \_\_\_\_\_

---







实验七名称： 第 7 次实验（数组）

实验目的： \_\_\_\_\_

日期：            年        月        日

本次实验成绩： \_\_\_\_\_ 指导教师签名： \_\_\_\_\_

---









实验八名称： 第 8 次实验（字符串）

实验目的： \_\_\_\_\_

日期：            年        月        日

本次实验成绩： \_\_\_\_\_ 指导教师签名： \_\_\_\_\_

---







实验九名称： 第 9 次实验（文件）

实验目的： \_\_\_\_\_

日期：            年        月        日

本次实验成绩： \_\_\_\_\_ 指导教师签名： \_\_\_\_\_

---









实验十名称： 第 10 次实验（数组和指针）

实验目的：

---

日期：            年        月        日

本次实验成绩：

指导教师签名：

---







实验十一名称： 第 11 次实验（结构体）

---

实验目的： \_\_\_\_\_

---

日期：            年        月        日

本次实验成绩：

指导教师签名：

---









实验十二名称： 第 12 次实验（其它杂项）

---

实验目的：

---

日期：            年        月        日

本次实验成绩：

指导教师签名：

---







## 附录

## 高级语言程序设计实验指导手册

李震雄编

2009年9月（第2次修订）

## 第 1 次实验（熟悉实验环境）

## 主要任务：

- 1 学会使用FTP，上传下载资料；
- 2 熟练掌握安装一个C语言集成开发环境，并熟悉集成开发环境的基本使用技巧；
- 3 练习“Hello world”。
- 4 选做《A First Book of ANSI C》中的练习题：

- \_\_\_\_\_
- \_\_\_\_\_

## 用 Microsoft Visual C++ 2008 Express 编写 C 语言程序

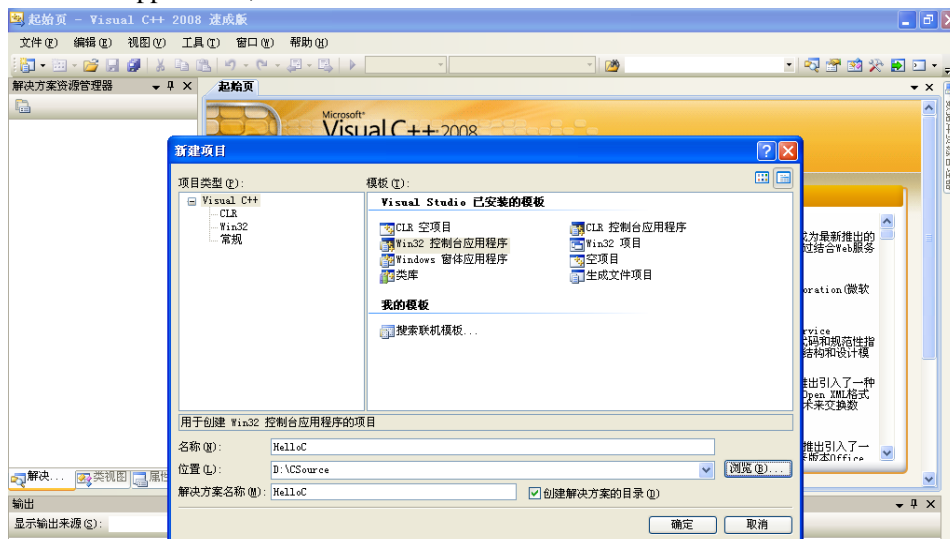
Note: Microsoft Visual C++ 2008 Express can be downloaded from Microsoft website at:

<http://www.microsoft.com/express/download/>

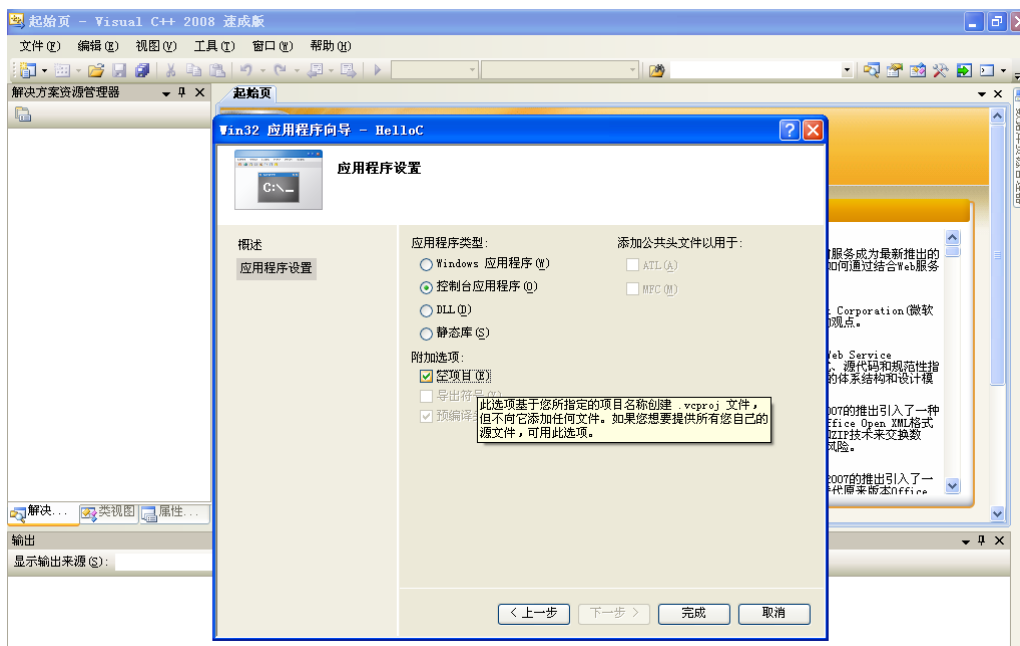
1. Start Visual C++ 2008 Express
2. Build a new project: File/New/Project



3. Select Win32 Console Application, and choose a folder to save it.



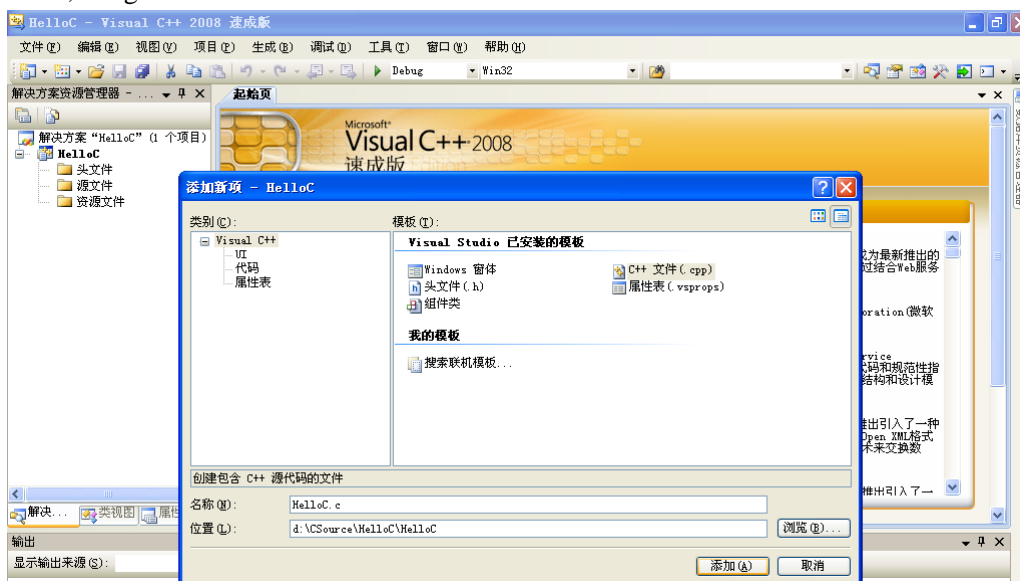
## 4. Make sure Console Application and Empty project are selected under Application Configuration.



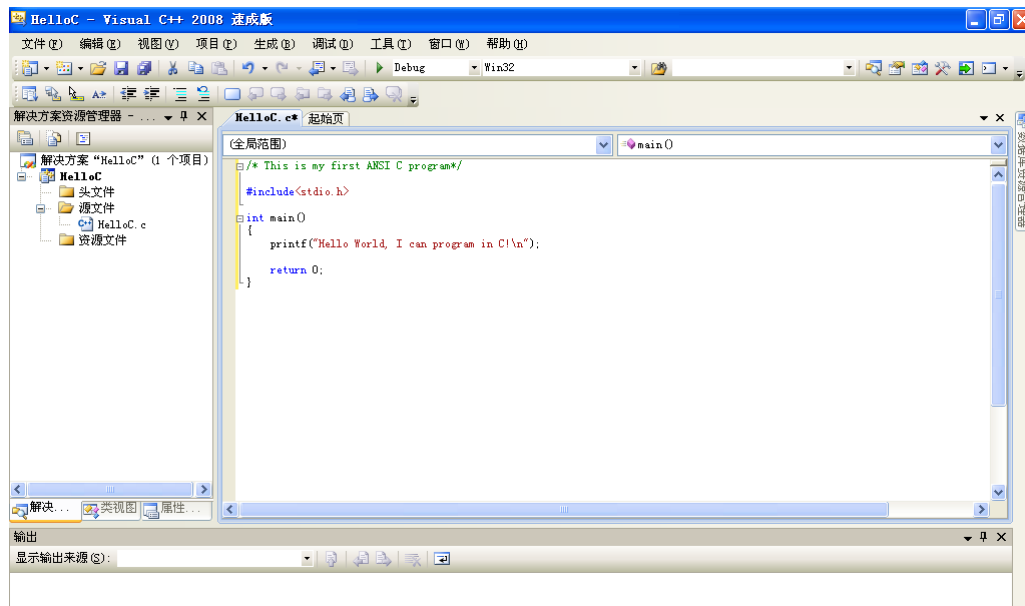
## 5. Once the above step is finished, add a new item from Source/Add/New Item.



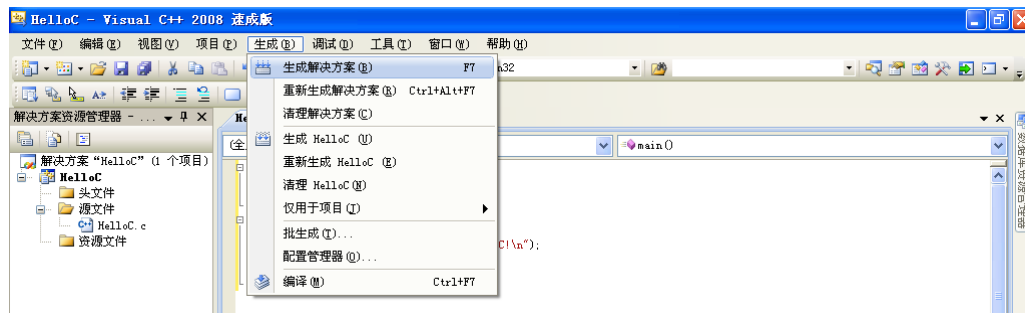
## 6. Select C++ file, but give it a name with .c as extension.



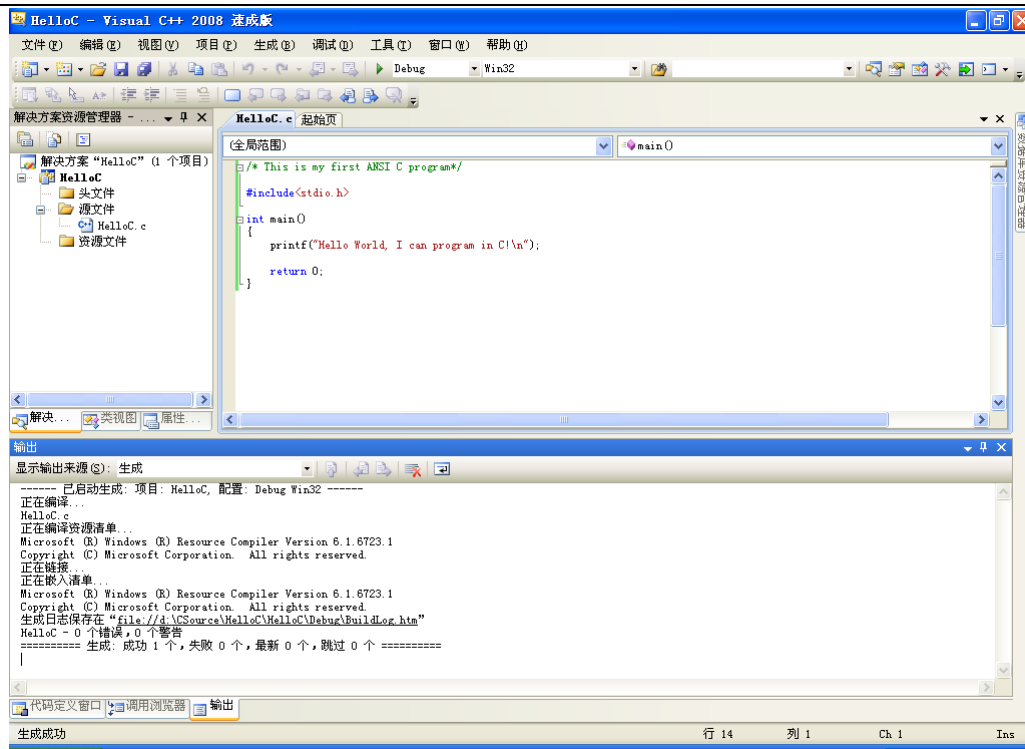
7. Now you can code the ANSI C program.



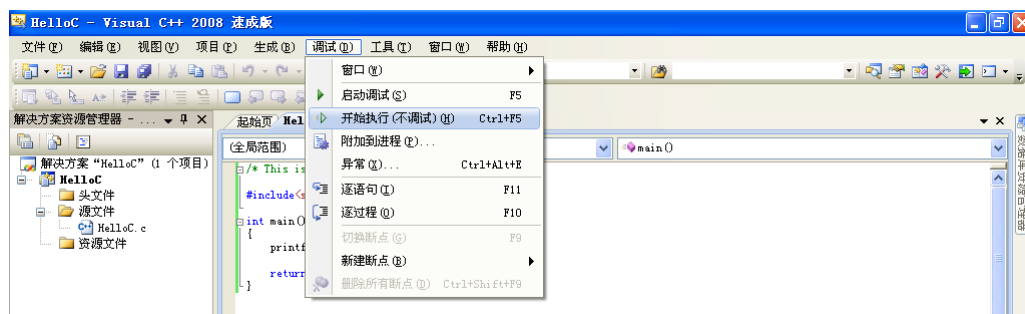
8. Now build the solution from Build/Build Solution or press F7 key.



9. Build information will be displayed.

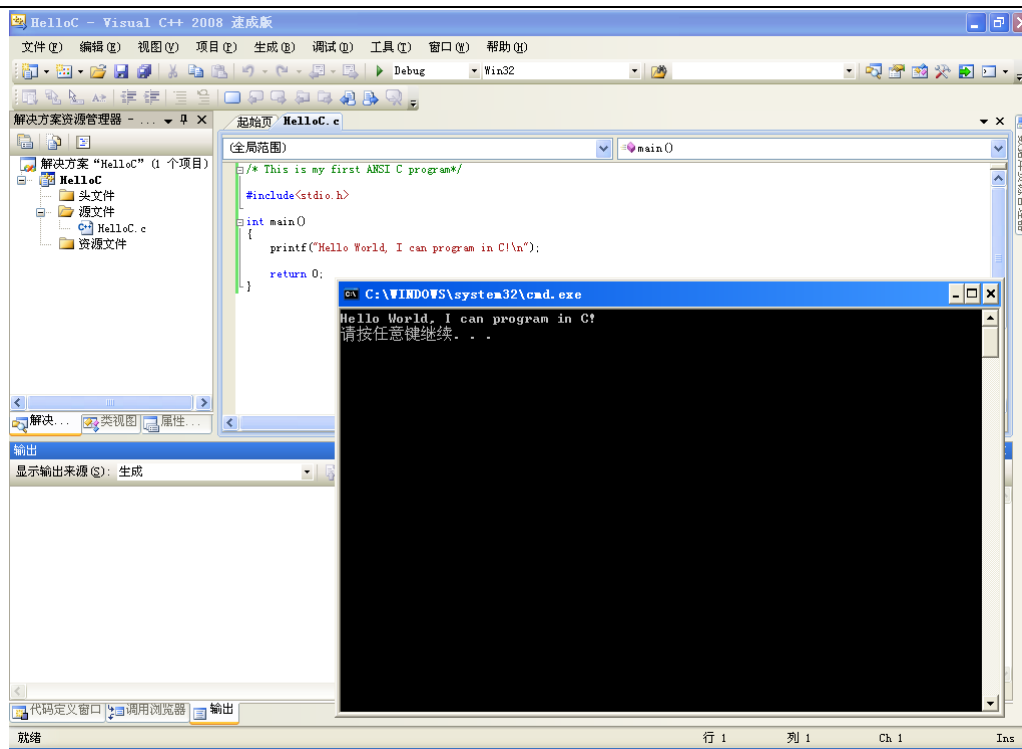


10. Run the program by chose Debug/Run or Ctrl+5. You may not see the console window if you chose Start debug or F5.



11. Here is the result.





Congratulations! You have done great job! Start you programming journey now. Have Fun!

## 第2次实验 (printf() 练习)

主要任务:

1. 熟悉 printf() 显示变量值, 表达式结果的功能
2. 编写一个顺序计算的程序, 要求选写算法, 再写流程图或伪代码, 最终调试完成程序。

知识点应用:

1. printf() 的控制字符串及转义字符
2. 正确运用面向过程的程序设计的方式和方法

练习项目:

### 1. printf() 练习

(1) 应用 printf() 计算出下列表达式的值。

1.  $3+4*6$
2.  $3*4/6+6$
3.  $2*3/12*8/4$
4.  $10*(1+7*3)$
5.  $20-2/6+3$
6.  $20-2/(6+3)$
7.  $(20-2)/6+3$
8.  $(20-2)/(6+3)$
9.  $3.0+4.0*6.0$
10.  $3.0*4.0/6.0+6.0$

11.  $2.0*3.0/12.0*8.0/4.0$
12.  $10.0*(1.0+7.0*3.0)$
13.  $20.0-2.0/6.0+3.0$
14.  $20.0-2.0/(6.0+3.0)$
15.  $(20.0-2.0)/6.0+3.0$
16.  $(20.0-2.0)/(6.0+3.0)$
17. 'm'-5
18. 'm'+5
19. 'G'+6
20. 'G'-6
21. 'b'-'a'
22. 'g'-'a'+1
23. 'G'-'A'+1

要求：1—8 显示为十进制整数；9—16 显示为小数，保留小数 2 位，宽度为 8；17—20 显示为字符；21—23 显示为十进制整数。

报告：

1. 记录各表达式的计算结果
2. 怎样应用控制字符串显示所期望的结果格式
3. 总结字符、整数、小数显示的方式方法
4. 总结表达式计算

评分标准：

正确完成任务：60 分

记录详略得当：30 分（实验步骤算法[包括今后的流程图]占 15 分；有实验结果记录说明占 5 分；总结最好能有自己出错的分析和总结占 10 分）

书写工整：10 分

## 2. 简单程序设计

- (1) 写一个简单的程序，完成教材中的练习：\_\_\_\_\_
- (2) 已知摄氏温度（℃）与华氏温度（°F）的转换公式为： $^{\circ}\text{C} = (5/9) (^{\circ}\text{F} - 32)$ ，试用一个简单的程序计算出 0 摄氏度是多少华氏温度？要求精确至小数点后 1 位。

选做项目：

## 第 3 次实验（scanf() 练习）

主要任务：

1. 了解地址的基本概念
2. 理解 scanf() 的输入
3. 理解并熟悉 scanf() 的使用，数字的输入及字符的输入

知识点应用：

1. 地址的基本概念

## 2. scanf ( ) 输入字符与数字

练习项目：

1. 教材指定练习：\_\_\_\_\_
2. 将第 2 次实验中的“摄氏温度 (°C) 与华氏温度 (°F) 的转换”程序，做改进，使之能根据用户的输入给出相应的结果。

选做项目：

## 第 4 次实验（控制流）

主要任务：

1. 熟悉控制流，熟练掌握控制流的应用
2. 掌握逻辑表达式、关系表达式的概念并能正确应用
3. 熟练掌握分支结构、循环结构，正确应用三种基本结构，完成程序设计

知识点应用：

控制流的概念，关系表达式，逻辑表达式，分支结构，循环结构

练习项目：

1. 输入三个数字，输出最大者和最者；
2. 企业发放的奖金根据利润提成。利润(I)低于或等于 10 万元时，奖金可提 10%；利润高于 10 万元，低于 20 万元时，低于 10 万元的部分按 10%提成，高于 10 万元的部分，可提成 7.5%；20 万到 40 万之间时，高于 20 万元的部分，可提成 5%；40 万到 60 万之间时高于 40 万元的部分，可提成 3%；60 万到 100 万之间时，高于 60 万元的部分，可提成 1.5%，高于 100 万元时，超过 100 万元的部分按 1%提成，从键盘输入当月利润 I，求应发放奖金总数？
3. 百马百担问题：有 100 匹马，驮 100 担货，大马驮 3 担，中马驮 2 担，两匹小马驮 1 担,问有大、中、小马各多少？
4. 相传印度国王要褒奖他的聪明能干的宰相达依尔（国际象棋发明者），问他要什么？达依尔回答：“陛下只要在国际象棋棋盘的第一个格子上放一粒麦子，第二个格子放二粒麦子，以后每一个格子的麦子数都按前一格的两倍计算。如果陛下按此法给我 64 格的麦子，就感激不尽，其它什么也不要了。”国王想，“这还不容易！”让人扛了一袋麦子，但很快用光了，再扛出一袋还是不够，请你为国王算一下共要给达依尔多少小麦？（设 1 立方米小麦约  $1.4 \times 10^8$  粒。）
5. 打印出以下图案。要求可以根据用户的要求调整菱形的大小。如：

最长行为 3 的

```

      *
     * *
    * * *
```

最长行为 7 的

```

      *
     * *
    * * *
   * * * *
  * * * * *
 * * * * * *
```

```
* * * * *  
  * * *  
    *
```

选做项目：

## 第 5/6 次实验（函数）

主要任务：

学会分析和设计函数，用自己设计的函数来组织程序。

知识点应用：

1. 熟练掌握函数的概念
2. 掌握函数的声明、定义及调用
3. 掌握函数设计的基本方法
4. 掌握模块化程序的基本方法

练习项目：

1. 设计一个程序根据用户的选择（1.日期转换成天数；2.指定年份和第几天求出日期）调用设计要的程序给出正确的输出。
2. 练习函数的设计程序，求  $n!$ ？用户给定  $n$  的值计算出  $n!$  的值，显示出来。请分别使用递归和迭代的方式完成设计。
3. 教材指定练习:\_\_\_\_\_

选做项目：

1. 猴子吃桃问题：猴子第一天摘下若干个桃子，当即吃了一半，还不过瘾，又多吃了一个，第二天早上又将剩下的桃子吃掉一半，又多吃了一个。以后每天早上都吃了前一天剩下的一半零一个。到第 10 天早上想再吃时，见只剩下一个桃子了。求第一天共摘了多少？（提示：采取逆向递推的方法，从后往前推算。）
2. 请用给出“汉诺塔”问题的递归实现程序。

## 第 7 次实验（数组）

主要任务：

1. 熟练掌握数据级的基本概念
2. 掌握数据的声明、定义及引用
3. 正确应用数据来设计和解决问题

知识点应用：

1. 数组的概念
2. 数组的声明、定义及引用
3. 数组的综合应用

练习项目：

1. 打印出杨辉三角形（要求打印出 10 行如下图）

```

1
1 1
1 2 1
1 3 3 1
1 4 6 4 1
1 5 10 10 5 1

```

2. 应用数组计算并保存，Fibonacci 数列的前 12 项值。

Fibonacci 数列的提示：有一对兔子，从出生后第 3 个月起每个月都生一对兔子，小兔子长到第三个月后每个月又生一对兔子，假如兔子都不死，问每个月的兔子总数为多少？

3. 数组排序，定义包含一个 10 个整数的数组，设计起泡排序的函数和选择排序的函数，通过调用两个排序函数，显示每趟排序的结果，比较起泡排序与选择排序的差异。
4. 用二分法在一个有序整数序列中，查找用户给定的数字，并给出结果（若找到相同数字给出其在序列中的位置，若未找到相同数字则提示用户）。

选做项目：

## 第 8 次实验（字符串）

主要任务：

1. 熟练掌握字符串的基本概念
2. C 语言中字符串的存储特点
3. 应用字符串解决实际问题

知识点应用：

1. 字符串的声明、定义、和操作
2. 字符串的存储和访问
3. 字符串的应用

练习项目：

1. 将输入的数字转换成二进制字符输出。
2. 试写一个程序统计输入的字符串中，包含的字符数、行数及单词数。提示：这里的单词是指前后均是空白字符的字符序列，如 adsaf asdfdsaf, <sdfsd 中共有 1 行、23 个字符、3 个单词。

选做项目：

试编写程序，统计多行输入的字符串长度，并将统计结果以升序方式列出，以水平柱状图的形式显示出来。

如：

输入：

0

1

22

333

4444  
666666  
55555  
666666  
88888888  
88888888  
88888888  
88888888

输出：

| length | Number | Graphic |
|--------|--------|---------|
| 1      | 2      | ■ ■     |
| 2      | 1      | ■       |
| 3      | 3      | ■       |
| 4      | 1      | ■       |
| 5      | 1      | ■       |
| 6      | 2      | ■ ■     |
| 8      | 4      | ■ ■ ■ ■ |

## 第 9 次实验（文件）

主要任务：

综合应用结构体及文件的知识，掌握对文件操作的基本技能。

知识点应用：

数据文件的声明、打开、读取、写入操作

练习项目：

1. 题目：有五个学生，每个学生有 3 门课的成绩，从键盘输入相关数据（包括：学号，姓名和三门课程的成绩），计算出平均成绩，并将原有的数据和计算出的平均分数存放在磁盘文件"stud.txt"中。

选做项目：

## 第 10 次实验（数组和指针）

主要任务：

1. 熟练掌握数组、地址和指针的概念
2. 掌握数组名、地址和指针的关系
3. 正确应用数组保存数据

知识点应用：

数组、地址和指针

练习项目：

1. 请使用数组元素存取方式，设计实现一个函数可以将任意字符数组元素存放位置逆转的程序。
2. 请使用指针存取方式完成练习 1
3. 用二维数组保存用户输入的字符串，当用户终止输入时，将所有输入中最长的字符串及其长度输出。

选做项目：

## 第 11 次实验（结构体）

主要任务：

1. 掌握结构体的声明、定义、引用及相关操作
2. 熟练掌握结构的基本概念和操作

知识点应用：

结构体的声明、定义、引用

练习项目：

1. 采用结构体变量的方法保存学生成绩，有五个学生，每个学生有 3 门课的成绩，从键盘输入相关数据（包括：学号，姓名和三门课程的成绩），计算出平均成绩。
2. 采用结构体数组的方法保存学生成绩，完成练习 1。
3. 采用链表的方法保存学生成绩，完成练习 1。

选做项目：

1. 创建一个学生成绩的链表，完成对链表的插入、修改、删除等操作。

## 第 12 次实验（其它杂项）

### 本章内容可全部选，给出了部份题目的参考程序。

主要任务：

1. 掌握类型定义的概念
2. 掌握位运算的概念
3. 掌握宏的概念
4. 掌握命令行参数的概念

知识点应用：

类型定义，位运算，宏，命令行参数

练习项目：

1. 将你设计的带有结构体定义的程序，给每一种结构体类型，应用 typedef 取一个新的名称
2. 设计程序，统计用户输入的整数的二进制串中包含多少个“1”
3. 使用宏来，改写你已有程序中的符号常量及简单的表达式；
4. 设计一个程序，将通过命令行参数传递的信息显示出来。
5. 设计一个程序，将通过命令行参数传递的逆波兰表达式的值计算出来。

选做项目：

题目：学习使用按位与  $\&$ 。

1.程序分析： $0\&0=0$ ;  $0\&1=0$ ;  $1\&0=0$ ;  $1\&1=1$

2.程序源代码：

```
#include "stdio.h"
main()
{
int a,b;
a=077;
b=a&3;
printf("\40: The a & b(decimal) is %d \n",b);
b&=7;
printf("\40: The a & b(decimal) is %d \n",b);
}
```

题目：学习使用按位或  $\mid$ 。

1.程序分析： $0\mid0=0$ ;  $0\mid1=1$ ;  $1\mid0=1$ ;  $1\mid1=1$

2.程序源代码：

```
#include "stdio.h"
main()
{
int a,b;
a=077;
b=a|3;
printf("\40: The a & b(decimal) is %d \n",b);
b|=7;
printf("\40: The a & b(decimal) is %d \n",b);
}
```

题目：学习使用按位异或  $\wedge$ 。

1.程序分析： $0\wedge0=0$ ;  $0\wedge1=1$ ;  $1\wedge0=1$ ;  $1\wedge1=0$

2.程序源代码：

```
#include "stdio.h"
main()
{
int a,b;
a=077;
b=a^3;
printf("\40: The a & b(decimal) is %d \n",b);
b^=7;
printf("\40: The a & b(decimal) is %d \n",b);
}
```

题目：取一个整数 a 从右端开始的 4~7 位。



程序分析：可以这样考虑：

- (1)先使 a 右移 4 位。
- (2)设置一个低 4 位全为 1,其余全为 0 的数。可用 $\sim(\sim 0 < 4)$
- (3)将上面二者进行&运算。

2.程序源代码：

```
main()
{
    unsigned a,b,c,d;
    scanf("%o",&a);
    b=a>>4;
    c=~(~0<<4);
    d=b&c;
    printf("%o\n%o\n",a,d);
}
```

=====

题目：学习使用按位取反~。

1.程序分析： $\sim 0=1$ ； $\sim 1=0$ ；

2.程序源代码：

```
#include "stdio.h"
main()
{
    int a,b;
    a=234;
    b=~a;
    printf("\40: The a's 1 complement(decimal) is %d \n",b);
    a=~a;
    printf("\40: The a's 1 complement(hexidecimal) is %x \n",a);
}
```

=====

题目：学习 static 定义静态变量的用法

1.程序分析：

2.程序源代码：

```
#include "stdio.h"
#include "conio.h"
varfunc()
{
    int var=0;
    static int static_var=0;
    printf("\40:var equal %d \n",var);
    printf("\40:static var equal %d \n",static_var);
    printf("\n");
    var++;
    static_var++;
}
void main()
{
```

```

int i;
for(i=0;i<3;i++)
    varfunc();
getch();
}

```

=====

题目：学习使用 auto 定义变量的用法

1.程序分析：

2.程序源代码：

```

#include "stdio.h"
#include "conio.h"
main()
{
    int i,num;
    num=2;
    for(i=0;i<3;i++)
    {
        printf("\40: The num equal %d \n",num);
        num++;
        {
            auto int num=1;
            printf("\40: The internal block num equal %d \n",num);
            num++;
        }
    }
    getch();
}

```

=====

题目：学习使用 static 的另一用法。

1.程序分析：

2.程序源代码：

```

#include "stdio.h"
#include "conio.h"
main()
{
    int i,num;
    num=2;
    for(i=0;i<3;i++)
    {
        printf("\40: The num equal %d \n",num);
        num++;
        {
            static int num=1;
            printf("\40:The internal block num equal %d\n",num);
            num++;
        }
    }
}

```

```

    }
    getch();
}

```

=====

题目：学习使用 `external` 的用法。

1.程序分析：

2.程序源代码：

```

#include "stdio.h"
#include "conio.h"
int a,b,c;
void add()
{
    int a;
    a=3;
    c=a+b;
}
void main()
{
    a=b=4;
    add();
    printf("The value of c is equal to %d\n",c);
    getch();
}

```

=====

题目：学习使用 `register` 定义变量的方法。

1.程序分析：

2.程序源代码：

```

#include "stdio.h"
#include "conio.h"
void main()
{
    register int i;
    int tmp=0;
    for(i=1;i<=100;i++)
        tmp+=i;
    printf("The sum is %d\n",tmp);
    getch();
}

```

=====

题目：宏 `#define` 命令练习(1)

1.程序分析：

2.程序源代码：

```

#include "stdio.h"
#include "conio.h"
#define TRUE 1
#define FALSE 0

```

```

#define SQ(x) (x)*(x)
void main()
{
    int num;
    int again=1;
    printf("\40: Program will stop if input value less than 50.\n");
    while(again)
    {
        printf("\40:Please input number==>");
        scanf("%d",&num);
        printf("\40:The square for this number is %d \n",SQ(num));
        if(num>=50)
            again=TRUE;
        else
            again=FALSE;
    }
    getch();
}

```

=====

题目：宏#define 命令练习(2)

1.程序分析：

2.程序源代码：

```
#include "stdio.h"
```

```
#include "conio.h"
```

```
/*宏定义中允许包含两道衣裳命令的情形，此时必须在最右边加上"\n"*/
```

```
#define exchange(a,b) { \
```

```

        int t;\
        t=a;\
        a=b;\
        b=t;\
    }

```

```
void main(void)
```

```

{
    int x=10;
    int y=20;
    printf("x=%d; y=%d\n",x,y);
    exchange(x,y);
    printf("x=%d; y=%d\n",x,y);
    getch();
}

```

=====

题目：宏#define 命令练习(3)

1.程序分析：

2.程序源代码：

```
#define LAG >
```

```
#define SMA <
```

```

#define EQ ==
#include "stdio.h"
#include "conio.h"
void main()
{
    int i=10;
    int j=20;
    if(i LAG j)
        printf("\40: %d larger than %d \n",i,j);
    else if(i EQ j)
        printf("\40: %d equal to %d \n",i,j);
    else if(i SMA j)
        printf("\40:%d smaller than %d \n",i,j);
    else
        printf("\40: No such value.\n");
    getch();
}

```

=====

题目：#if #ifdef 和#endif 的综合应用。

1. 程序分析：

2.程序源代码：

```

#include "stdio.h"
#include "conio.h"
#define MAX
#define MAXIMUM(x,y) (x>y)?x:y
#define MINIMUM(x,y) (x>y)?y:x
void main()
{
    int a=10,b=20;
#ifdef MAX
    printf("\40: The larger one is %d\n",MAXIMUM(a,b));
#else
    printf("\40: The lower one is %d\n",MINIMUM(a,b));
#endif
#ifndef MIN
    printf("\40: The lower one is %d\n",MINIMUM(a,b));
#else
    printf("\40: The larger one is %d\n",MAXIMUM(a,b));
#endif
#undef MAX
#ifdef MAX
    printf("\40: The larger one is %d\n",MAXIMUM(a,b));
#else
    printf("\40: The lower one is %d\n",MINIMUM(a,b));
#endif
#define MIN

```

```

#ifndef MIN
    printf("\40: The lower one is %d\n",MINIMUM(a,b));
#else
    printf("\40: The larger one is %d\n",MAXIMUM(a,b));
#endif
    getch();
}

```

=====

题目：#include 的应用练习

1.程序分析：

2.程序源代码：

test.h 文件如下：

```
#define LAG >
```

```
#define SMA <
```

```
#define EQ ==
```

主文件如下：

```
#include "test.h" /*一个新文件 50.c， 包含 test.h*/
```

```
#include "stdio.h"
```

```
#include "conio.h"
```

```
void main()
```

```

{
    int i=10;
    int j=20;
    if(i LAG j)
        printf("\40: %d larger than %d \n",i,j);
    else if(i EQ j)
        printf("\40: %d equal to %d \n",i,j);
    else if(i SMA j)
        printf("\40:%d smaller than %d \n",i,j);
    else
        printf("\40: No such value.\n");
    getch();
}

```

## 第 13 次实验（课程设计）

主要任务：

对高级语言程序设计的基本特性及知识要点进行总结，并就大作业相关事宜做安排布置及指导。

知识点应用：

1. 归纳总结课程内容
2. 常量的应用技巧
3. 宏的应用技巧
4. 数组的应用技巧
5. 指针的应用技巧

6. 文件操作的应用技巧
7. 程序设计的基本技巧

### 综合应用技术

#### 练习项目：

根据每个学期的具体要求选题，期中考试以后正式确定题目。

#### 选做项目：

1. 应用 C 语言设计一个学生信息管理系统。（详见大作业要求）
2. 应用 C 语言设计一个商品信息管理系统。（详见大作业要求）
3. 应用 C 语言设计一个分类记事本应用程序。（详见大作业要求）