云南大学数学与统计学实验教学中心

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**：模块化程序设计 | | |
| **实验编号**：NO.6 | **实验日期**：2017年1月6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 |

# 实验目的

1．掌握C语言中定义函数的方法及其调用方法。

2．掌握函数实参与形参的对应关系以及“值传递”与“地址传递”的方式。

3．掌握全局变量和局部变量、动态变量与静态变量的概念和使用方法。

4．掌握函数的嵌套调用与递归调用的方法。

5．掌握通过数组进行数据传递的方法。

6．学会使用宏替换编写程序，弄清“文件包含”的作用。

7．学会全局变量和局部变量、动态变量和静态变量的概念和使用方法。

# 二、实验内容

1．变量的生存期是一个静态的概念，是不变的，作用域是一个动态的概念，随程序的运行而变。从作用域角度分，有**局部变量**和**全局变量**；从变量存在的时间来分，有**动态存贮**和**静态存贮**。

2．C语言中的参数调用为值调用，名调用（地址）通过指针实现的。

3．用递归方式编写程序，一般由三部分组成：

递归条件；

递归化简；

递归出口；

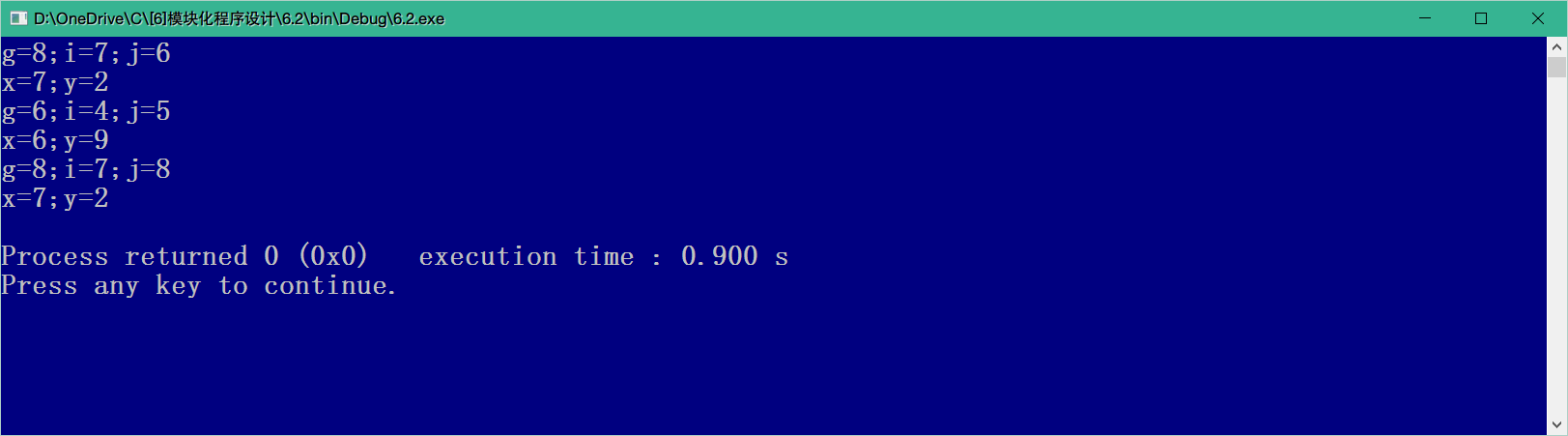
一个递归总是可分为“回推”和“递推”两个阶段，即从要解决问题出发，一步步推到已知条件，然后再由已知条件一步步找到结论。

1通过运行下面程序,熟悉函数的调用方法。

|  |  |
| --- | --- |
| 1  2  3  4  5  6  7  8  9  10  11  12  13  14  15  16 | /\* filename: 6.1 \*/  #include<stdio.h>  void fun**(**int i**,** int j**,** int k**)**  **{**  int t**;**  t**=(**i**+**j**+**k**)/**2**;**  printf**(**"t=%d\n"**,**t**);**  **}**  int main**()**  **{**  int x**,**y**,**z**;**  x**=**4**;**y**=**12**;**z**=**6**;**  fun**(**x**,**y**,**z**);**  printf**(**"x=%d;y=%d;z=%d\n"**,**x**,**y**,**z**);**  **return** 0**;**  **}** |

运行下面程序，写出执行结果。

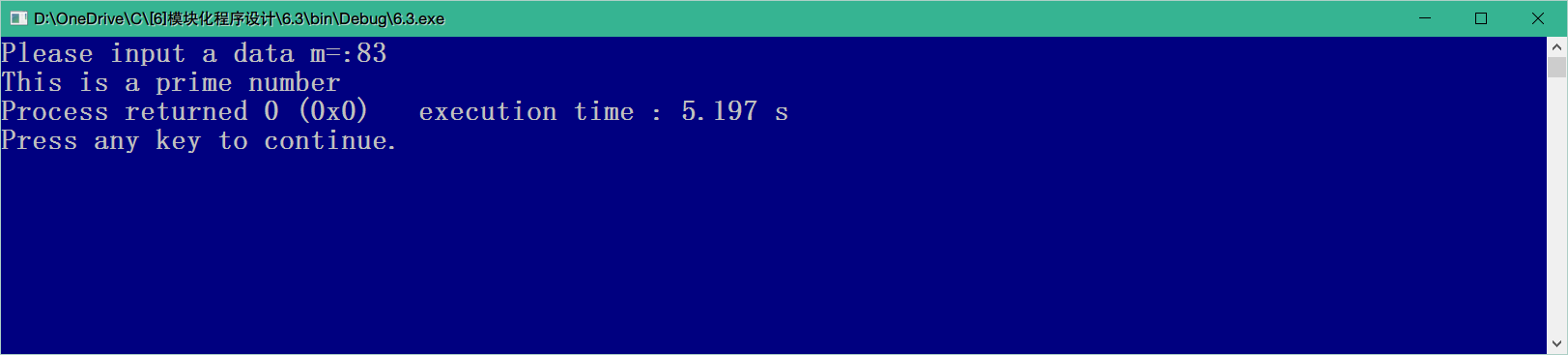
|  |  |
| --- | --- |
| 1  2  3  4  5  6  7  8  9  10  11  12  13  14  15  16  17  18  19 | /\* filename: 6.2 \*/  #include<stdio.h>  void f**(**int i**,**int j**)**  **{**  int x**,**y**,**g**;**  g**=**8**;**x**=**7**;**y**=**2**;**  printf**(**"g=%d;i=%d;j=%d\n"**,**g**,**i**,**j**);**  printf**(**"x=%d;y=%d\n"**,**x**,**y**);**  **}**  int main**()**  **{**  int i**,**j**,**x**,**y**,**n**,**g**;**  i**=**4**;**j**=**5**;**g**=**x**=**6**;**y**=**9**;**n**=**7**;**  f**(**n**,**6**);**  printf**(**"g=%d;i=%d;j=%d\n"**,**g**,**i**,**j**);**  printf**(**"x=%d;y=%d\n"**,**x**,**y**);**  f**(**n**,**8**);**  **return** 0**;**  **}** |



３．编写一个判断素数的函数，在主函数输入一个整数，输出是否是素数的信息。

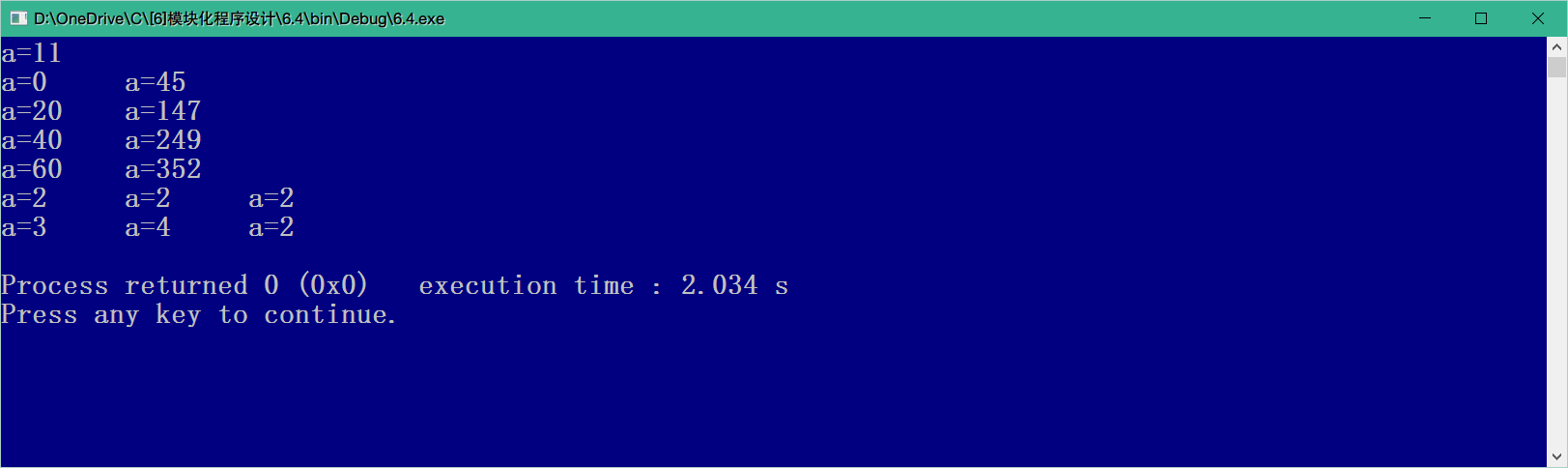
参考程序如下：

|  |  |
| --- | --- |
| 1  2  3  4  5  6  7  8  9  10  11  12  13  14  15  16  17  18  19  20  21  22  23  24  25  26 | /\* filename: 6.3 \*/  #include<stdio.h>  #include<math.h>  void Prime**(**int n**)**  **{**  int i**,**k**;**  k**=**n**/**2**;**  **for(**i**=**2**;**i**<=**k**;**i**++)**  **if** **(**n**%**i**==**0**)** **break;**  **if(**i**>=**k**+**1**)**  **{**  printf**(**"This is a prime number"**);**  **}**  **else**  **{**  printf**(**"This is not a prime number"**);**  **}**  **}**  int main**()**  **{**  int m**;**  printf**(**"Please input a data m=:"**);**  scanf**(**"%d"**,&**m**);**  Prime**(**m**);**  **return** 0**;**  **}** |



先读懂程序，分析出结果，然后上机运行此程序。

|  |  |
| --- | --- |
| 1  2  3  4  5  6  7  8  9  10  11  12  13  14  15  16  17  18  19  20  21  22  23  24  25  26  27  28 | /\* filename: 6.4 \*/  #include<stdio.h>  #define FUE(K) K+3.14159  #define PR(a) printf("a=%d\t", (int)(a))  #define PRINT(a) PR(a) ; putchar('\n')  #define PRINT2(a,b) PR(a); PRINT(b)  #define PRINT3(a,b,c) PR(a); PRINT2(b,c)  #define MAX(a,b) (a<b? b:a)  int main**()**  **{**  **{**  int x**=**2**;**  PRINT**(**x**\***FUE**(**4**));**  **}**  **{**  int f**;**  **for(**f**=**0**;**f**<=**60**;**f**+=**20**)**  **{**  PRINT2**(**f**,**5.12**\***f**+**45**);**  **}**  **}**  **{**  int x**=**1**,**y**=**2**;**  PRINT3**(**MAX**(**x**++,**y**),**x**,**y**);**  PRINT3**(**MAX**(**x**++,**y**),**x**,**y**);**  **}**  **return** 0**;**  **}** |



\*5 编写一函数，由实参传来一个字符串，统计此字符串中字母、数字、空格和其它字符的个数，在主函数中输入字符串，输出上述结果。此程序编写时最好把存放字母、数字、空格和其它字符的变量定义为全程变量，这样在函数中就不必定义了。用其它方法也可以。

写出计算Ackermann函数Ack(m,n)的递归函数，对于m≥0,n≥0,Ack(m,n)定义为：

Ack**(**0**,**n**)=**n**+**1

Ack**(**m**,**0**)=**Ack**(**m**-**1**,**1**)**

Ack**(**m**,**n**)=**Ack**((**m**-**1**),**Ack**(**m**,**n**-**1**))**

对于，，要显示出计算结果。

编程并上机调试。

|  |  |
| --- | --- |
| 1  2  3  4  5  6  7  8  9  10  11  12  13  14  15  16  17  18  19  20  21  22  23  24  25  26  27  28  29  30  31  32  33  34  35  36  37  38  39  40 | #include <stdio.h>  int main**()**  **{**  int mm**,**nn**;**  long int a**;**  printf**(**"Please enter m,n:"**);**  scanf**(**"%d %d"**,&**mm**,&**nn**);**  a**=**ack**(**mm**,**nn**);**  printf**(**"ack(%d,%d)=%ld\n"**,**mm**,**nn**,**a**);**  getchar**();**  **}**  long int ack**(**int m**,**int n**)**  long int value**;**  **{**  **if(**m**<**0**||**n**<**0**)**  **{**  printf**(**"The condiction of caculating is ont exist,EXIT!\n"**);**  exit**();**  **}**  **if(**m**==**0**)**  **{**  value**=**n**+**1**;**  printf**(**"ack(0,%d)=%d\n"**,**n**,**n**+**1**);**  **}**  **else**  **{**  **if(**n**==**0**)**  **{**  value**=**ack**(**m**-**1**,**1**);**  printf**(**"ack(%d,1)=%d\n"**,**m**-**1**,**ack**(**m**-**1**,**1**));**  **}**  **else**  **{**  printf**(**"ack(%d,%d)=ack(%d,ack(%d,%d))\n"**,**m**,**n**,**m**-**1**,**m**,**n**-**1**);**  value**=**ack**((**m**-**1**),**ack**(**m**,**n**-**1**));**  **}**  **}**  **return(**value**);**  **}**  long int ack**(**int m**,**int n**);** |

\*6 写一函数，输入一行字符，将此字符串中最长和单词输出。

编写程序并上机调试运行，记录下程序运行的结果。

提示：我们认为每行字符都是由字母组成的字符串，在程序中设longest函数为实现寻找最长的字符串。设标记flag表示单词是否开始，flag=1表示单词开始，flag=0表示单词未开始。Point代表当前单词的起始位置，place代表最长单词的起始位置。Len代表当前单词已累计字母的个数，length代表先前单词中最长单词的长度。

下面给一个程序段，上实现寻找最长的单词。

|  |  |
| --- | --- |
| 1  2  3  4  5  6  7  8  9  10  11  12  13  14  15  16  17  18  19  20  21  22  23  24  25 | int longest**(**string**)**  char string**[];**  **{**  int len**=**0**,**i**,**length**=**0**,**flag**=**1**,**place**,**point**;**  **for(**i**=**0**;**i**<=**strlen**(**string**);**i**++)**  **if(**alphab**(**string**[**i**]))**  **if(**flag**)**  **{**  point**=**i**;**  flag**=**0**;**  **}**  **else**  len**++;**  **else**  **{**  flag**=**1**;**  **if(**len**>**length**)**  **{**  length**=**len**;**  place**=**point**;**  len**=**0**;**  **}**  **}**  **return(**place**);**  **}** |

1．求两整数的最大公约数和最小公倍数。用一函数求最大公约数，用另一函数调用此函数求出最大公约数，并用求出的最大公约数求最小公倍数。

具体要求如下：

①用全局变量。将最大公约数与最小公倍数设为全局变量，在主函数中输出它们的值。

②不用全局变量。最大公约数和最小公倍数由被调模块返回值。

2．计算并输出



具体要求如下:

①编制一个函数，返回值。

②编制主函数，由键盘输入与，调用(1)中的函数计算下列算式值。



③在主函数中，输入与之前要有提示，并检查输入数据的合理性，对于不合理的输入，应输出出错信息，不再进行计算。在函数*pq*(*n*)中也要检查*n*的合理性，当*n*＜0时输出出错信息，不再进行计算。

④分别输入，，，，运行该程序。

⑤画出模块的流程图。

3．编写程序，要求找出满足下列条件的3位数：它是完全平方数，又有两位数字相同。如：144、676。

要求：设计一函数判断一个三位数是否为完全平方数，设计另一函数判断一个三位数中是否有两位数字相同，再在主函数中调用这两个函数，找出所有的满足这两个条件的三位数。

4．编写函数，将*n*个整数的数列进行重新排放，重新排放后的结果为：前段都是奇数，后段都是偶数，并编写主函数完成：

①输入10个整数；

②调用此函数进行重排；

③输出重排后的结果。

5．输入10个学生4门课的成绩，分别用函数求：

①每个学生的平均成绩；

②每门课的及格率；

③最高分所对应的学生和课程。

# 四 习题

1．下列说法哪些是不正确的

1）C语言的函数必须在主函数的前面定义。（×）

2）函数名前面都必须写数据类型和存贮类型。（×）

3）一个C程序可以包含多个函数，并且必须有最多有一个主函数。（√）

4）函数的外部不允许再说明其它变量。（×）

5）void类函数不能有返回值。（√）

6）函数不一定都有返回值，不管其类型是否为void。（×）

7）引用程序中的函数被调用函数必须在调用函数前面预先定义。（√）

8）在函数内部定义的变量都是局部量。（√）

9）所有外部量，都必须在其它程序中定义过。

10）函数返回的数据类型必须和函数数据类型一致。（√）

11）调用函数的实参与被调用函数的形参必须保证其个数、次序类型完全一致。（√）

12）在C程序中，函数嵌套调用不允许超过18层。（×）

2．下列定义有何错误？

|  |  |
| --- | --- |
| 1  2  3  4  5  6  7  8  9  10  11  12  13  14  15  16  17  18  19  20  21  22  23  24  25  26  27  28  29  30  31  32  33  34  35  36  37  38  39  40  41  42  43  44  45  46  47  48  49  50 | /\*----------------------------------\*/  float f**(**x**,**y**);**  int x**,**y**;**  **{**  **}**  /\*----------------------------------\*/  f**(**int x**,**y**)**  **{**  retuen x**\***y**;**  **}**  /\*----------------------------------\*/  /\*----------------------------------\*/  int max**(**x**,**y**)** /\*定义求最大值的函数\*/  **{** int x**,**y**,**z**;**  z**=**x**>**y**?** x**:**y**;**  **return** z**;**  **}**  （4）  int f**(**x**,**y**,**z**)**  int x**,**y**;**  **{**  int z**;**  **:**  **}**  /\*----------------------------------\*/  int f**(**int n**)** /\*定义求n!的函数\*/  **{**  **if** n**<=**0 **return** 1**;**  **else** f**=**n**\***f**(**n**-**1**);**  **}**  /\*----------------------------------\*/  int main**(**x**,**y**)**  int x**,**y**;**  **{**  printf**(**“x**+**y**=%**d\n”**,**x**+**y**);**  **}**  int x**,**y**;**  f**(**int x**)**  **{**  y**=**x**\***x**;** **return** y **;}**  int f1**(**x**)**  int x**;**  **{**  int f2**(**y**)**  int y**;**  **{** …  **}**  **}** |

3．定义一个函数，调用程序通过f(n,x)的形式就可能计算，，等样式的表达式的值。

|  |  |
| --- | --- |
| 1  2  3  4  5  6  7  8  9  10  11  12  13  14  15  16  17  18  19  20  21  22  23 | #include <stdio.h>  #include <math.h>  void f**(**int n**,**double x**)**  **{**  **switch(**n**)**  **{**  **case** 1**:**printf**(**"%lf"**,(**pow**(**x**,**3**)+**x**-**1**));break;**  **case** 2**:**printf**(**"%lf"**,**pow**((**x**+**5**),**3**)+(**x**+**5**)-**1**);break;**  **case** 3**:**printf**(**"%lf"**,**pow**(**sin**(**x**),**3**)+**sin**(**x**)-**1**);break;**  **default:**printf**(**"Error!"**);**  **}**  **}**  int main**()**  **{**  printf**(**"Please input:\n"**);**  int n**;**  double x**;**  scanf**(**"%d,%lf"**,&**n**,&**x**);**  printf**(**"The answer is: "**);**  f**(**n**,**x**);**  **return** 0**;**  **}** |

通过n进行switch语句判断。

4．写一递归函数，计算



5．写一递归函数，将读入的整数按位分开后以相反顺序输出。

|  |  |
| --- | --- |
| 1  2  3  4  5  6  7  8  9  10  11  12  13  14  15  16  17  18  19  20  21  22  23  24  25  26 | #include <stdio.h>  int f**(**int n**)**  **{**  int z**;**  **if(**n**<**10**)**  **{**  printf**(**"%d"**,**n**);**  **return** 0**;**  **}**  **else**  **{**  z**=**n**%**10**;**  printf**(**"%d"**,**z**);**  n**=(**n**-**n**%**10**)/**10**;**  **return** f**(**n**);**  **}**  **}**  int main**()**  **{**  int n**;**  printf**(**"Please input:\n"**);**  scanf**(**"%d"**,&**n**);**  f**(**n**);**  **return** 0**;**  **}** |

6．写一函数，它将回送整数*m*从右边开始的第k个数字的值，例如,.

|  |  |
| --- | --- |
| 1  2  3  4  5  6  7  8  9  10  11  12  13  14  15  16  17  18  19  20  21 | #include <stdio.h>  int digh**(**int n**,**int k**)**  **{**  int y**;**  int t**;**  **for(**t**=**1**;**t**<=**k**;**t**++)**  **{**  y**=**n**%**10**;**  n**=(**n**-**y**)/**10**;**  **}**  **return** y**;**  **}**  int main**()**  **{**  int n**,**k**;**  printf**(**"Please input:\n"**);**  scanf**(**"%d,%d"**,&**n**,&**k**);**  printf**(**"%d"**,**digh**(**n**,**k**));**  **return** 0**;**  **}** |

7．写两个函数，分别求两个整数的最大公约数和最小公倍数。

|  |  |
| --- | --- |
| 1  2  3  4  5  6  7  8  9  10  11  12  13  14  15  16  17  18  19  20  21  22  23  24  25  26  27  28  29  30  31  32  33 | #include <stdio.h>  int main**()**  **{**  int a**,**b**;**  printf**(**"----------------------\n"**);**  printf**(**"Please input a and b:\n"**);**  scanf**(**"%d,%d"**,&**a**,&**b**);**  int memo**=**a**\***b**;**/\*储存最初的数值，为计算LCM做准备\*/  printf**(**"----------------------\n"**);**  **if(**a**!=**b**)**  **{**  **for(;**a**!=**b**;)**  **{**  **if(**a**>**b**)**  **{**  a**=**a**-**b**;**  **}**  **else**  **{**  b**=**b**-**a**;**  **}**  **}**  printf**(**"GCD(a,b)=%d\n"**,**a**);**  printf**(**"LCM(a,b)=%d\n"**,**memo**/**a**);**  **}**  **else**  **{**  printf**(**"GCD(a,b)=%d\n"**,**a**);**  printf**(**"LCM(a,b)=%d\n"**,**memo**/**a**);**  **}**  printf**(**"----------------------\n"**);**  **return** 0**;**  **}** |

8．定义一个宏 *swap*(*x*, *y*),完成对两个整数*x*, *y*的交换。

|  |  |
| --- | --- |
| 1  2  3  4  5  6  7  8  9  10  11  12  13  14  15  16 | #include <stdio.h>  int X**,**Y**;**  void swap**(**int x**,**int y**)**  **{**  Y**=**x**;**  X**=**y**;**  **}**  int main**()**  **{**  printf**(**"Please input:\n"**);**  scanf**(**"%d,%d"**,&**X**,&**Y**);**  swap**(**X**,**Y**);**  printf**(**"%d,%d"**,**X**,**Y**);**  **return** 0**;**  **}** |

9．定义一个宏*max*(*x*, *y*, *z*)从三个数*x*, *y*, *z*中找出最大数。

|  |  |
| --- | --- |
| 1  2  3  4  5  6  7  8  9  10  11  12  13  14  15  16  17  18  19 | #include <stdio.h>  int max**(**int a**,**int b**)**  **{**  **return** a**>**b**?** a**:**b**;**  **}**  int max2**(**int a**,**int b**,**int c**)**  **{**  **return** max**(**a**,**b**)>**c**?** max**(**a**,**b**):**c**;**  **}**  int main**()**  **{**  int x**,**y**,**z**;**  printf**(**"Please input the values of x,y and z:\n"**);**  scanf**(**"%d,%d,%d"**,&**x**,&**y**,&**z**);**  printf**(**"max(x,y,z)=%d"**,**max2**(**x**,**y**,**z**));**  **return** 0**;**  **}** |

1．先静态阅读以下程序，然后上机运行程序，查看运行结果是否与你阅读的结果一致？不一致的原因何在？

|  |  |
| --- | --- |
| 1  2  3  4  5  6  7  8  9  10  11  12  13  14  15 | #include<stdio.h>  int main**(** **)**  **{**  int a**=**4**,**b**,**c**,**k**;**  k**=**f**(**a**+**2**,**b**+**1**);**  printf**(**"\n %d,%d,%d,%d"**,**a**,**b**,**c**,**k**);**  getchar**();**  **return** 0**;**  **}**  int a**=**3**,**b**=**5**,**c**=**2**;**  int f**(**int a**,**int b**)**  **{**  a**++;**b**--;**c**=**a**+**b**;**  **return(**a**+**b**+**c**);**  **}** |

2．静态分析以下程序的执行结果，然后上机运行程序，将分析结果与运行结果加以对比，从中领会静态局部变量的含义及用法。

|  |  |
| --- | --- |
| 1  2  3  4  5  6  7  8  9  10  11  12  13  14  15  16  17  18 | #include<stdio.h>  int f**(**int x**)**  **{**  static int f**=**0**,**y**=**0**;**  **if(**f**==**0**)** y**+=**2**\***x**;**  **else**  **if(**f**==**1**)** y**+**3**=**3**\***x**;**  **else** y**+=**4**\***x**;**  f**++;**  **return** y**;**  **}**  int main**(** **)**  **{**  printf**(**"\n %d"**,**f**(**2**));**  printf**(**"\n %d"**,**f**(**2**));**  printf**(**"\n %d"**,**f**(**2**));**  getchar**();**  **}** |

# 三、实验环境

Windows10 1703 Enterprise中文版操作系统；

Turbo C 2.0与Code::Blocks 16.01集成开发环境。

# 四、实验过程

1题：

这个题要观察到最小公倍数与最大公约数之间的关系。

|  |  |
| --- | --- |
| 1  2  3  4  5  6  7  8  9  10  11  12  13  14  15  16  17  18  19  20  21  22  23  24  25  26  27  28  29  30  31  32  33  34 | /\*用模块化编程设计一个求最大公约数的函数\*/  #include <stdio.h>  int main**()**  **{**  /\*函数声明\*/  int a**,**b**;**/\*要输入的数据\*/  int Max**;**/\*用来存放两个数的最大值\*/  int GCD**,**LCM**;**  printf**(**"Please input the numbers:\n"**);**  scanf**(**"%d,%d"**,&**a**,&**b**);**  Max**=**max**(**a**,**b**);**  **for(**LCM**=**Max**;**LCM**<=**a**\***b**;**LCM**++)**  **{**  **if(**LCM**%**a**==**0 **&&** LCM**%**b**==**0**)**  **{**  printf**(**"The LCM of these two is %d\n"**,**LCM**);**  printf**(**"The GCD of these two is %d"**,**a**\***b**/**LCM**);**  **}**  **}**  **return** 0**;**  **}**  int max**(**int a**,**int b**)**  **{**  int z**;**  **if(**a**>**b**)**  **{**  z**=**a**;**  **}**  **else**  **{**  z**=**b**;**  **}**  **return(**z**);**  **}** |

2题：

|  |  |
| --- | --- |
| 1  2  3  4  5  6  7  8  9  10  11  12  13  14  15  16  17  18  19  20  21  22  23  24  25  26  27  28  29  30  31  32  33  34  35  36  37  38 | /\*计算阶乘表达式\*/  #include <stdio.h>  #include <math.h>  int main**()**  **{**  int pq**(**int a**);**  int m**,**n**;**/\*定义表达式中的变量\*/  printf**(**"Please input your numbers with order:\n"**);**  scanf**(**"%d,%d"**,&**m**,&**n**);**  int a**,**b**,**c**;**/\*计算数值\*/  int ans**;**/\*输出答案\*/  **if(**m**<**0 **||** n**<**0**)**  **{**  printf**(**"ERROE INPUT!"**);**  **}**  **else**  **{**  a**=**pq**(**n**);**  b**=**pq**(**abs**(**m**-**n**));**  c**=**pq**(**m**);**  ans**=**c**/(**a**\***b**);**  printf**(**"The answer of the cal is%5d"**,**ans**);**  **}**  **return** 0**;**  **}**  int pq**(**int a**)**  **{**  int z**;**  **if(**a**==**1 **||** a**==**0 **||** a**<**0**)**  **{**  z**=**1**;**  **}**  **else**  **{**  z**=**a**\***pq**(**a**-**1**);**  **}**  **return** z**;**  **}** |

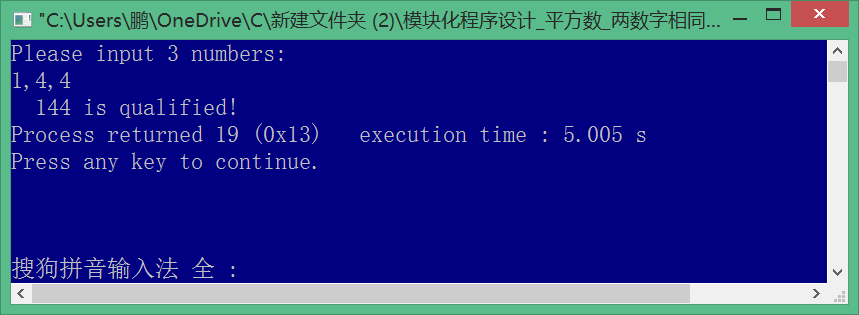


1

3题：

这个题在考察模块化编程的同时，还注重continue与break的用法。及时中断循环是减少CPU运算的重要策略，也能减少Debug时候耗费的时间。

|  |  |
| --- | --- |
| 1  2  3  4  5  6  7  8  9  10  11  12  13  14  15  16  17  18  19  20  21  22  23  24  25  26  27  28  29  30  31  32  33  34  35  36  37  38  39  40  41  42  43  44  45 | /\*判断输入的数字是不是三位完全平方数并且有且仅有两位数相同\*/  #include <stdio.h>  #include <math.h>  int main**()**  **{**  int judge**(**int a**,**int b**,**int c**);**  int compare**(**a**,**b**,**c**);**/\*声明函数\*/  int a**,**b**,**c**;**  printf**(**"Please input 3 numbers:\n"**);**  scanf**(**"%d,%d,%d"**,&**a**,&**b**,&**c**);**  **if(**judge**(**a**,**b**,**c**)==**1 **&&** compare**(**a**,**b**,**c**)==**1**)**  **{**  printf**(**"%5d is qualified!"**,**100**\***a**+**10**\***b**+**c**);**  **}**  **else**  **{**  printf**(**"Sorry,it isn't."**);**  **}**  **}**  int judge**(**int a**,**int b**,**int c**)**/\*判断是否是平方数\*/  **{**  int BOOL**;**  float z**;**/\*z是十进制下a,b,c所代表的数值\*/  float k**;**/\*k是z的平方根\*/  float j**;**/\*j是循环变量\*/  z**=**100**\***a**+**10**\***b**+**c**;**  k**=**sqrt**(**z**);**  **for(**j**=**1.0**;**j**<**z**;**j**++)**  **{**  **if(**j**-**k**==**0**)**  BOOL**=**1**;continue;**  **}**  **return** BOOL**;**  **}**  int compare**(**int a**,**int b**,**int c**)**  **{**  int BOOL**;**  **if((**a**==**b**&&**a**!=**c**)||(**a**==**c**&&**a**!=**b**)||(**b**==**c**&&**a**!=**b**))**  **{**  BOOL**=**1**;**  **}**  **return** BOOL**;**  **}** |



4题：

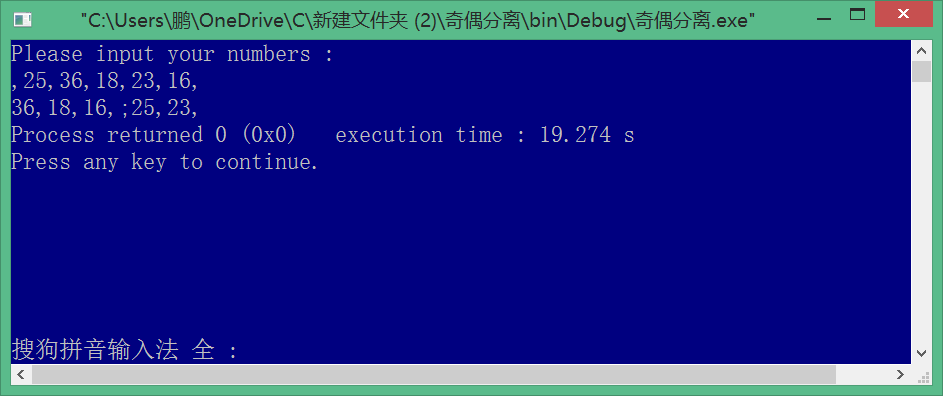
这个题的难点就是输入任意个数字然后进行排序，并且要求是一次性全部输入。

从用户角度看，这是人性化设计的一个方面，然而这就导致了我们不能使用数组进行循环输入。我的解决方案是首先定义一个很大的字符型数组，然后用一定的格式进行输入。在这里采用了英文的逗号进行分隔，然后根据逗号进行定位，判断之前的那一个数字是不是偶数。

我们知道，只要个位数字是偶数，即可判定是偶数。奇数也是这样的道理。

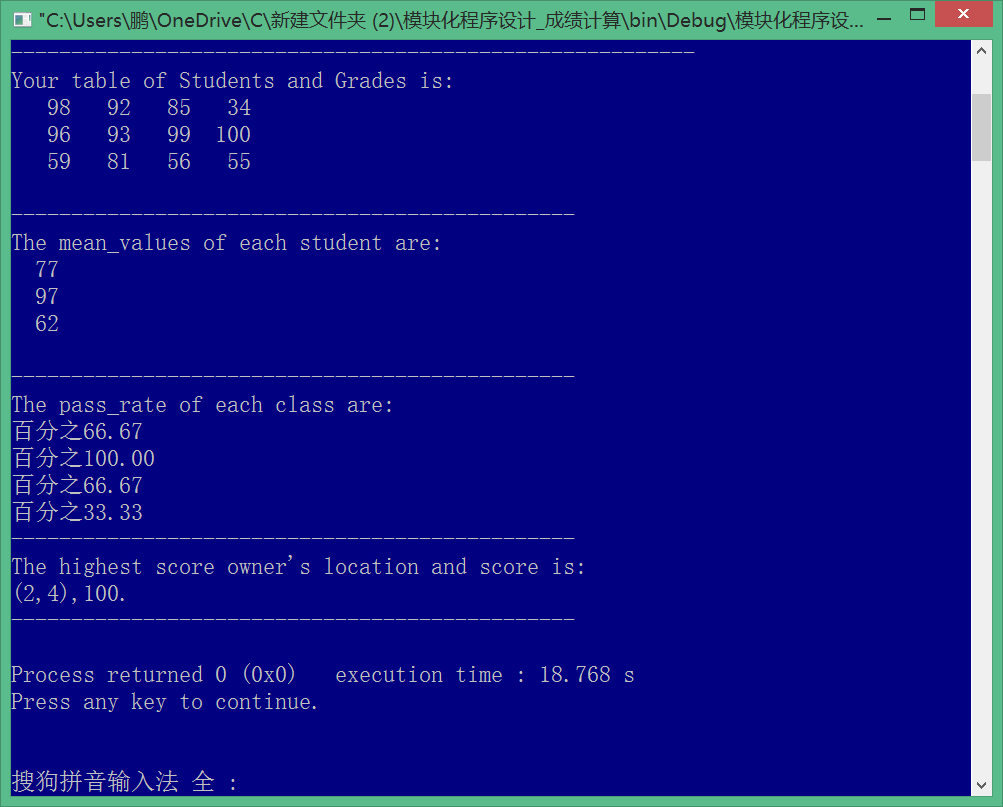
遗憾的是，这种定义大数组的方式会导致内存的浪费，并且在某些数据量特别大的情况下无法使用该程序。目前还没有好的解决办法。

|  |  |
| --- | --- |
| 1  2  3  4  5  6  7  8  9  10  11  12  13  14  15  16  17  18  19  20  21  22  23  24  25  26  27  28  29  30  31  32  33  34  35  36  37  38  39  40  41  42  43  44  45  46  47  48  49  50  51  52 | /\*首先用一个很大的字符型数组来存放我们的数据，用逗号分隔\*/  /\*编写程序中要注意中断循环\*/  #include<stdio.h>  #include<string.h>  int main**()**  **{**  char str**[**99**];**  char a**[**100**];**/\*置换数组必须尽可能大，即使有可能就是置换一个数字\*/  int i**,**j**,**k**;**/\*i,j是循环变量，k是赋值数组的循环变量\*/  printf**(**"Please input your numbers :\n"**);**  gets**(**str**);**  **for(**i**=**1**;**i**<**99**;**i**++)**/\*从第二项开始，所以输入的第一项是逗号\*/  **{**  **if((**str**[**i**]==**','**)&&(**str**[**i**-**1**]%**2**==**0**))**  **{**  **for(**k**=**1**;**str**[**i**-**k**]!=**','**;**k**++)**  **{**  a**[**k**]=**str**[**i**-**k**];**  **}**  **for(**j**=**k**-**1**;**j**>=**1**;**j**--)**  **{**  printf**(**"%c"**,**a**[**j**]);**  **}**  printf**(**","**);**  **}**  **if(**str**[**i**]==**'\0'**)**  **{**  **break;**  **}**  **}**  printf**(**";"**);**/\*之后输奇数\*/  **for(**i**=**1**;**i**<**99**;**i**++)**/\*从第二项开始，所以输入的第一项是逗号\*/  **{**  **if((**str**[**i**]==**','**)&&(**str**[**i**-**1**]%**2**!=**0**))**  **{**  **for(**k**=**1**;**str**[**i**-**k**]!=**','**;**k**++)**  **{**  a**[**k**]=**str**[**i**-**k**];**  **}**  **for(**j**=**k**-**1**;**j**>=**1**;**j**--)**  **{**  printf**(**"%c"**,**a**[**j**]);**  **}**  printf**(**","**);**  **}**  **if(**str**[**i**]==**'\0'**)**  **{**  **break;**  **}**  **}**  **return** 0**;**  **}** |



5题:

|  |  |
| --- | --- |
| 1  2  3  4  5  6  7  8  9  10  11  12  13  14  15  16  17  18  19  20  21  22  23  24  25  26  27  28  29  30  31  32  33  34  35  36  37  38  39  40  41  42  43  44  45  46  47  48  49  50  51  52  53  54  55  56  57  58  59  60  61  62  63  64  65  66  67  68  69  70  71  72  73  74  75  76  77  78  79  80  81  82  83  84  85  86  87  88  89  90  91  92  93  94  95  96  97  98  99  100  101  102  103  104  105  106  107  108  109  110  111  112  113  114  115  116  117 | /\*输入10个学生4门课的成绩，分别用函数求：\*/  /\*每个学生的平均成绩；\*/  /\*每门课的及格率；\*/  /\*最高分所对应的学生和课程。\*/  #include <stdio.h>  #define ROW 3  #define COLUMN 4  int main**()**  **{**  void mean**(**int a**[**ROW**][**COLUMN**]);**  void passrate**(**int a**[**ROW**][**COLUMN**]);**  void max**(**int a**[**ROW**][**COLUMN**]);**  int i**,**j**;**  int a**[**ROW**][**COLUMN**];**  printf**(**"Please input your table one by one with [Enter] to apart:\n"**);**  printf**(**"--------------------------------------------------------\n"**);**  **for(**i**=**0**;**i**<**ROW**;**i**++)**  **{**  **for(**j**=**0**;**j**<**COLUMN**;**j**++)**  **{**  scanf**(**"%d"**,&**a**[**i**][**j**]);**  **}**  **}**  printf**(**"---------------------------------------------------------\n"**);**  printf**(**"Your table of Students and Grades is:\n"**);**  **for(**i**=**0**;**i**<**ROW**;**i**++)**  **{**  **for(**j**=**0**;**j**<**COLUMN**;**j**++)**  **{**  printf**(**"%5d"**,**a**[**i**][**j**]);**  **if(**j**==**COLUMN**-**1**)**  **{**  printf**(**"\n"**);**  **}**  **}**  **}**  printf**(**"\n"**);**  printf**(**"-----------------------------------------------\n"**);**  printf**(**"The mean\_values of each student are:\n"**);**  mean**(**a**);**  printf**(**"-----------------------------------------------\n"**);**  printf**(**"The pass\_rate of each class are:\n"**);**  passrate**(**a**);**  printf**(**"-----------------------------------------------\n"**);**  printf**(**"The highest score owner's location and score is:\n"**);**  max**(**a**);**  printf**(**"-----------------------------------------------\n"**);**  **return** 0**;**  **}**  /\*-----------------------------------------------\*/  void mean**(**int a**[**ROW**][**COLUMN**])**  **{**/\*求平均值\*/  int sum**[**ROW**];**  int avg**[**ROW**];**  int i**,**j**;**  **for(**j**=**0**;**j**<**ROW**;**j**++)**  **{**/\*行循环\*/  sum**[**j**]=**0**;**/\*初始化设在大循环里面\*/  **for(**i**=**0**;**i**<**COLUMN**;**i**++)**  **{**/\*列循环\*/  sum**[**j**]=**sum**[**j**]+**a**[**j**][**i**];**  **}**  avg**[**j**]=**sum**[**j**]/**COLUMN**;**  **}**  **for(**j**=**0**;**j**<**ROW**;**j**++)**  **{**  printf**(**"%4d\n"**,**avg**[**j**]);**  **}**  printf**(**"\n"**);**  **}**  /\*-----------------------------------------------\*/  void passrate**(**int a**[**ROW**][**COLUMN**])**  **{**/\*求及格比例\*/  float count**[**COLUMN**];**  int i**,**j**;**  **for(**j**=**0**;**j**<**COLUMN**;**j**++)**  **{**  **for(**i**=**0**;**i**<**ROW**;**i**++)**  **{**  **if(**a**[**i**][**j**]>=**60**)**  **{**  count**[**j**]++;**  **}**  **}**  **}**  **for(**i**=**0**;**i**<**COLUMN**;**i**++)**  **{**  printf**(**"百分之%2.2f\n"**,(**count**[**i**]/**ROW**)\***100.0**);**  **}**  **}**  /\*-----------------------------------------------\*/  int max2**(**int a**,**int b**)**  **{**  **return(**a**>**b**?** a**:**b**);**  **}**  /\*-----------------------------------------------\*/  void max**(**int a**[**ROW**][**COLUMN**])**  **{**  int Max**;**  Max**=**a**[**0**][**0**];**  int column**,**row**;**/\*最大值的行列坐标\*/  int i**,**j**;**  **for(**i**=**0**;**i**<**ROW**;**i**++)**  **{**  **for(**j**=**0**;**j**<**COLUMN**;**j**++)**  **{**  **if(**Max**<**a**[**i**][**j**])**  **{**  Max**=**a**[**i**][**j**];**  row**=**i**;**  column**=**j**;**  **}**  **}**  **}**  printf**(**"(%d,%d),%d.\n"**,**row**+**1**,**column**+**1**,**Max**);**  **}** |



# 五、实验总结

问题：

1. 数组作为自定义函数的引用对象，当在main函数里面进行调用的时候，只需要引用数组的名称。在网络论坛以及同学中问了很多，终于知道了该怎么去做。

这次实验报告算是比较费心的一次。

我学会了用code::blocks进行debug，诚实说来，以前写程序都是把代码大体写出来，然后去排错。甚至有的时候连自己的思路都搞不清楚就去摁[F9]，导致了系统提示很多warning与error。

现在学聪明了一点，知道去 [Shift] + [F7] 进行step into，然后 [F7] next line 进行排错与检验运算过程。

上课老师说的编程的三个步骤：**流程**，**实例**，**动态调试**；当初确实没有搞懂，也没有学会编译环境的具体使用。

# 六、参考文献

[1]谭浩强，C 程序设计[M] (第四版)．北京：清华大学出版社，2010年6月（中国高等院校计算机基础教育课程体系规划教材）

[2]谭浩强，C程序设计( 第四版 )学习辅导，北京：清华大学出版社，2010年7月（中国高等院校计算机基础教育课程体系规划教材）

# 七、教师评语