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：循环结构程序设计 |  |  |
| 实验编号：No.04 | 实验日期：2017年8月9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. 进一步练习选择结构的程序设计。
2. 练习并掌握实现循环结构的三种方法。
3. 练习并掌握选择结构与循环结构的嵌套。
4. 掌握多重循环的应用。
5. 学会单步跟踪的操作方法。

# 实验环境

Windows10 Pro Workstation 17096；

Code::Blocks 16.01 GCC集成开发环境；

Cygwin GCC编译器。

# 实验内容

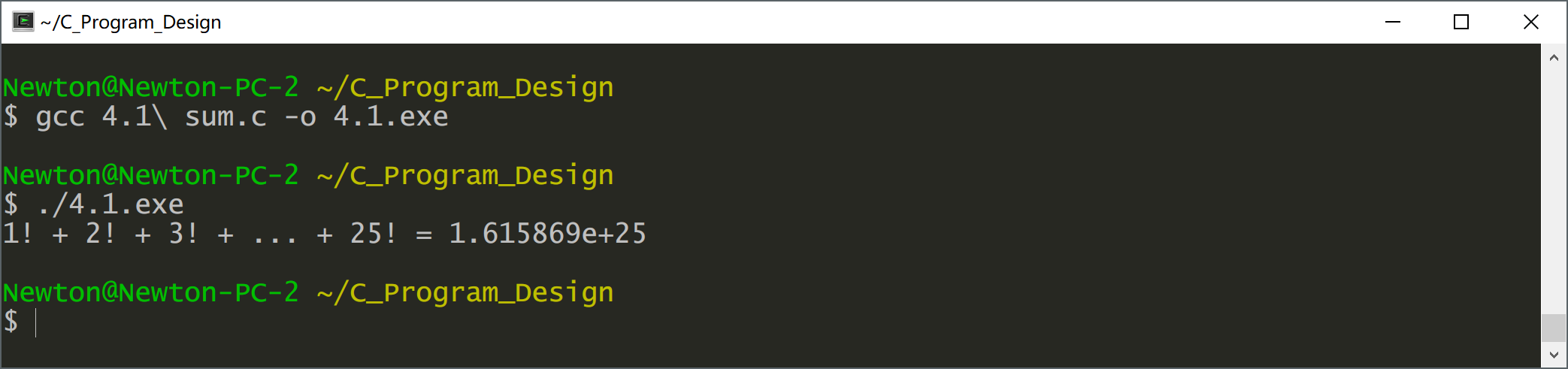
## 2题

求

### 程序代码：

|  |  |
| --- | --- |
| 1  2  3  4  5  6  7  8  9  10  11  12  13  14  15  16 | /\*  \* filename: 4.1 sum.c  \* property: exercise  \*/  #include <stdio.h>  int main**()** **{**  float n**,** s **=** 0**,** t **=** 1**;**  **for(**n **=** 1**;** n **<=** 25**;** n**++)** **{**  t **\*=** n**;**  s **+=** t**;**  **}**  printf**(**"1! + 2! + 3! + ... + 25! = %e\n"**,** s**);**  **return** 0**;**  **}** |

上机运行，并记录下结果。然后用另外两种循环语句实现上述功能。



## 3题

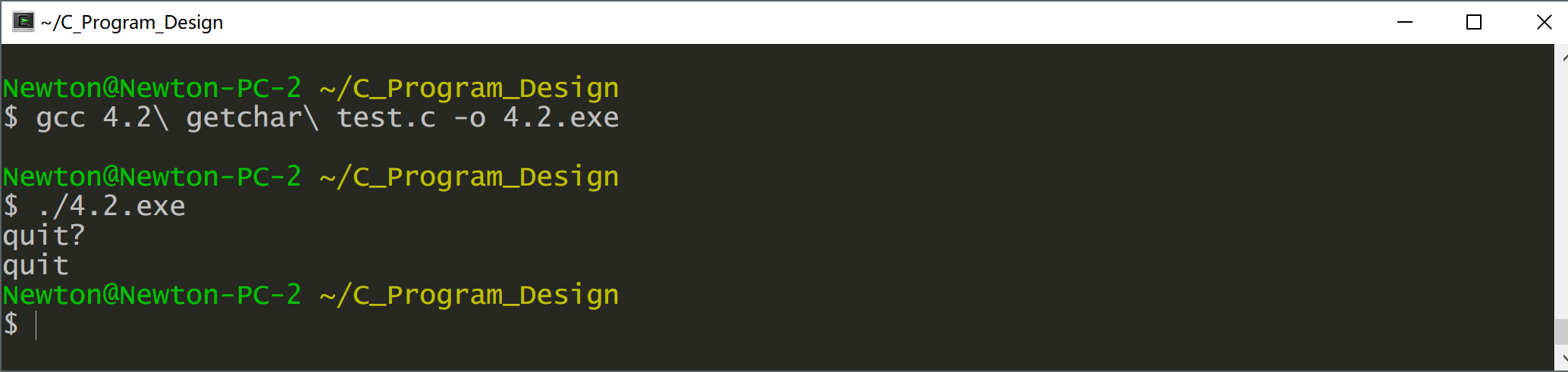
指出下面三个程序的功能，当输入“quit?”时，它们的执行结果是什么？

### 程序1

#### 程序代码

|  |  |
| --- | --- |
| 1  2  3  4  5  6  7  8  9  10  11  12  13  14  15  16 | /\*  \* filename: 4.2 getchar test.c  \* property: test  \*/  #include <stdio.h>  int main**()** **{**  char c**;**  c **=** getchar**();**  **while(**c **!=** '?'**)** **{**  putchar**(**c**);**  c **=** getchar**();**  **}**  **return** 0**;**  **}** |

#### 运行结果

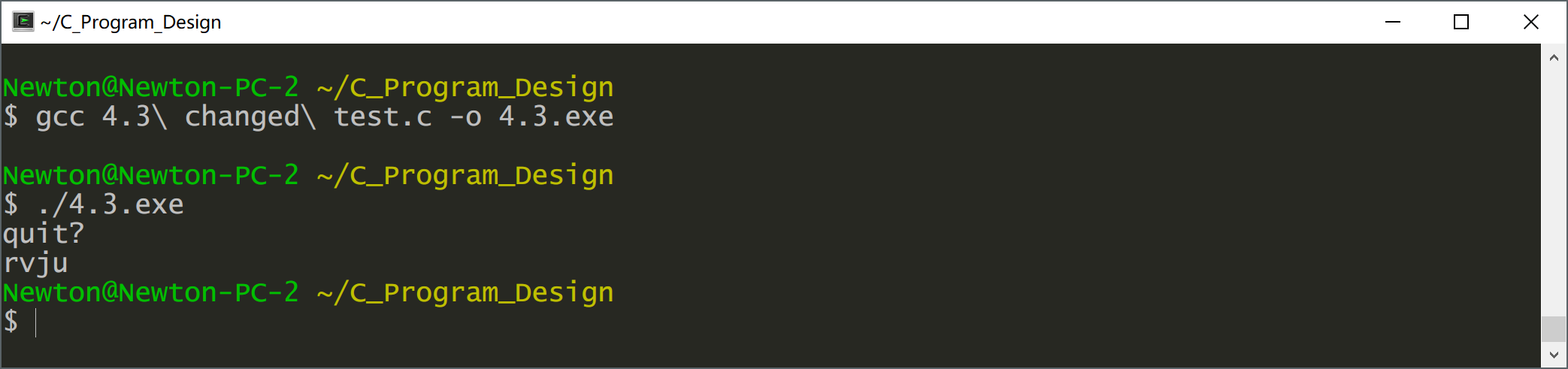


### 程序2

#### 程序代码

|  |  |
| --- | --- |
| 1  2  3  4  5  6  7  8  9  10  11  12  13  14 | /\*  \* filename: 4.3 changed test.c  \* property: test  \*/  #include <stdio.h>  int main**()** **{**  char c**;**  **while((**c**=**getchar**())** **!=** '?'**)** **{**  putchar**(++**c**);**  **}**  **return** 0**;**  **}** |

#### 运行结果

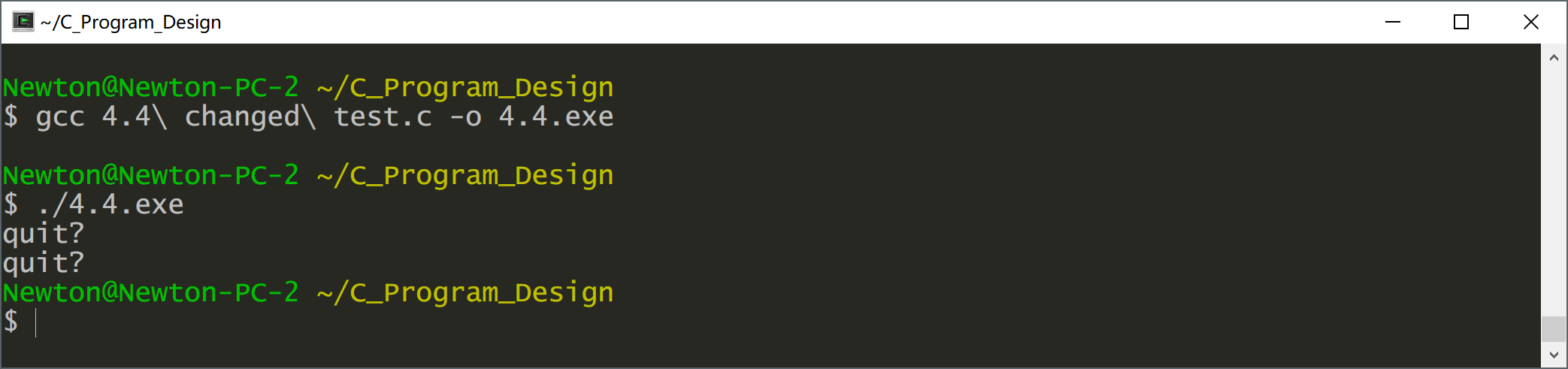


### 程序3

#### 程序代码

|  |  |
| --- | --- |
| 1  2  3  4  5  6  7  8  9  10  11  12  13  14 | /\*  \* filename: 4.4 changed test.c  \* property: test  \*/  #include <stdio.h>  int main**()** **{**  char c**;**  **while(**putchar**(**getchar**())** **!=** '?'**)** **{**  putchar**(++**c**);**  **}**  **return** 0**;**  **}** |

#### 运行结果



分析输出的三种不同结果，在实验报告中写出为什么。

原因：

（1）输入了’?’就结束循环.

（2）当输入的不是’?’时候，就输出该字符的下一个字符，当输入’?’后，结束循环。

（3）直接输出，但是在这里，变量c没有被改动过，++c都是乱码；程序的意义，就是输出从键盘得到的字符，之后判断输出的是不是’?’，之后输出++c。

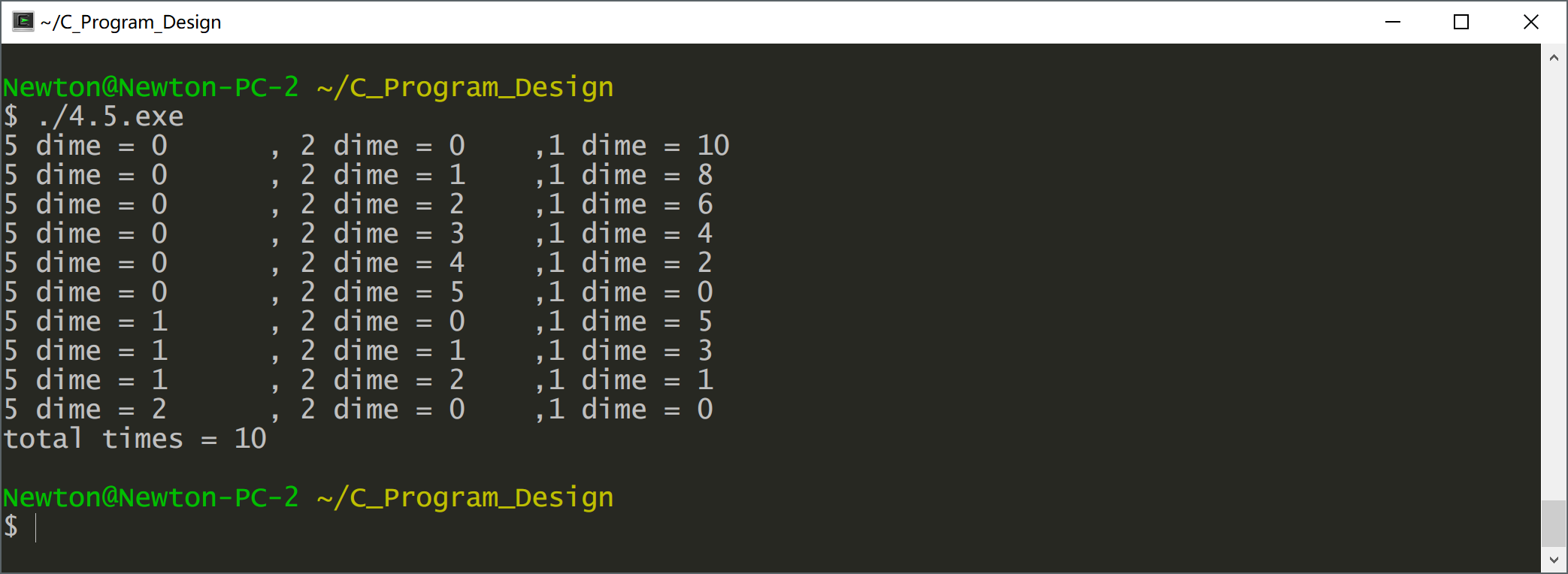
## 换硬币

换零钱。把一元钱全兑换成硬币，有多少种兑换方法？有五角、两角和一分的硬币。

### 程序代码

|  |  |
| --- | --- |
| 1  2  3  4  5  6  7  8  9  10  11  12  13  14  15  16  17  18  19 | /\*  \* filename: 4.5 coin.c  \* property: exercise  \*/  #include <stdio.h>  int main**()** **{**  int i**,** j**,** k**,** n**;**  n **=** 10**,** k **=** 0**;**  **for(**i **=** 0**;** i **<=** n **/** 5**;** i**++)** **{**  **for(**j **=** 0**;** j **<=** **(**n**-**5**\***i**)/**2**;** j**++)** **{**  printf**(**"5 dime = %d\t, 2 dime = %d\t,1 dime = %d\n"**,**i**,**j**,**n**-**i**\***5**-**j**\***2**);**  k**++;**  **}**  **}**  printf**(**"total times = %d\n"**,** k**);**  **return** 0**;**  **}** |

### 运行结果



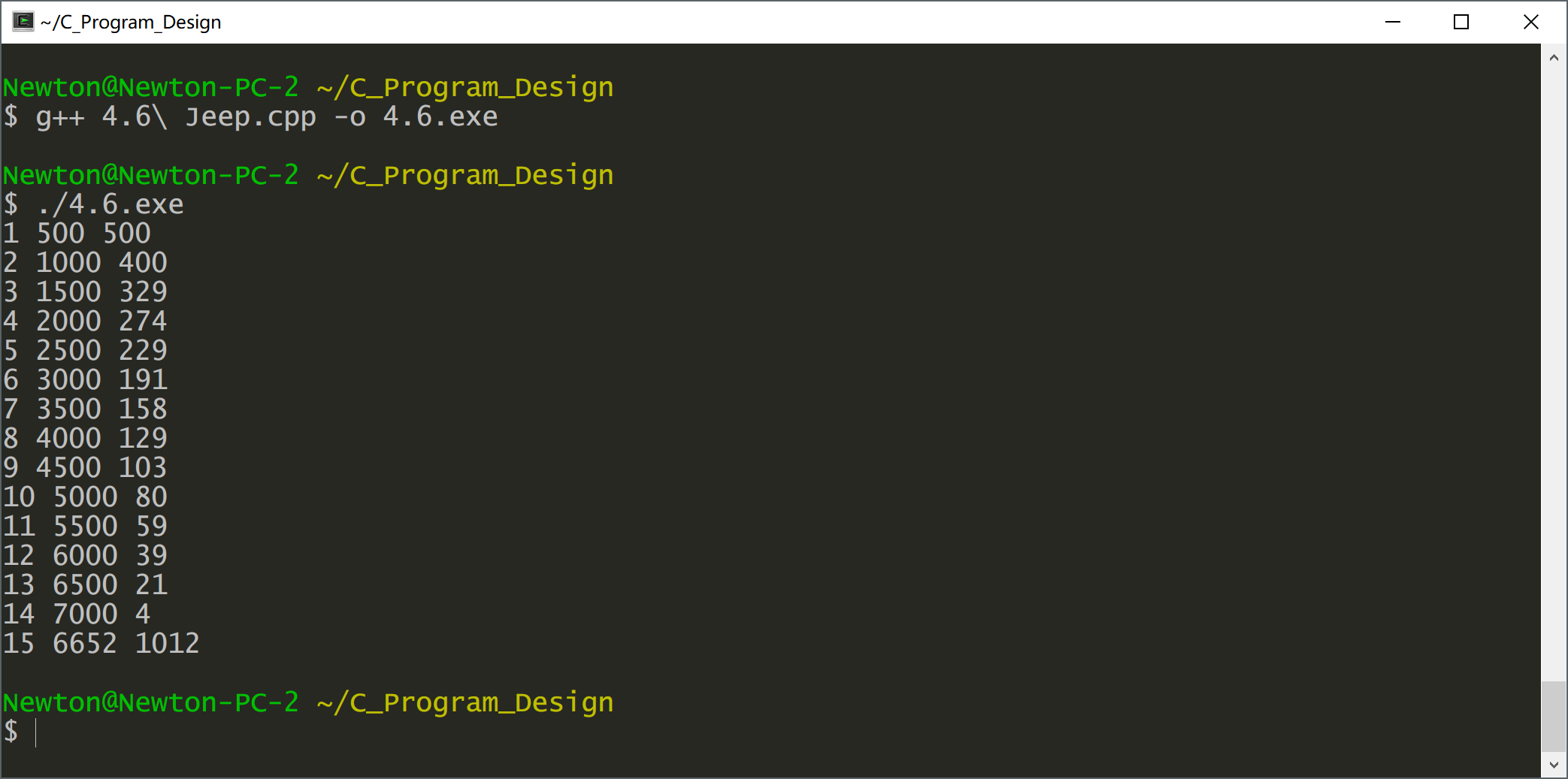
## \*5题

穿越沙漠。用一辆吉普车穿越1000公里的沙漠。吉普车的总装油量为500加仑，耗油量为1加仑/公里。由于沙漠中没有油库，必须先用车在沙漠中建立临时加油站，该吉普车要以最少的油耗穿越沙漠，应在什么地方建立临时油库，以及在什么地方安放多少油最好？

### 参考程序：

|  |  |
| --- | --- |
| 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 | /\*  \* filename: 4.6 Jeep.cpp  \* property: difficulty  \*/  #include <iostream>  #include <stdio.h>  **using** **namespace** std**;**  int main**()** **{**  int dis**,** k**,** oil**;**  dis **=** 500**;**  k **=** 1**;**  oil **=** 500**;**  **while(**dis**<**1000**)** **{**  printf**(**"%d %d %d\n"**,** k**,** oil**,** 1000 **-** dis**);**  k **=** k **+** 1**;**  dis **=** dis **+** 500 **/** **(**2**\***k **+** 1**);**  oil **=** 500 **\*** k**;**  **}**  oil **=** 500 **\*** **(**k **-** 1**)** **+** **(**1000 **-** dis**)** **\*** **(**2**\***k **-** 1**);**  printf**(**"%d %d %d\n"**,** k**,** oil**,** dis**);**  **return** 0**;**  **}** |

### 运行结果



## 动态调试练习

①上机运行程序，分析运行结果。

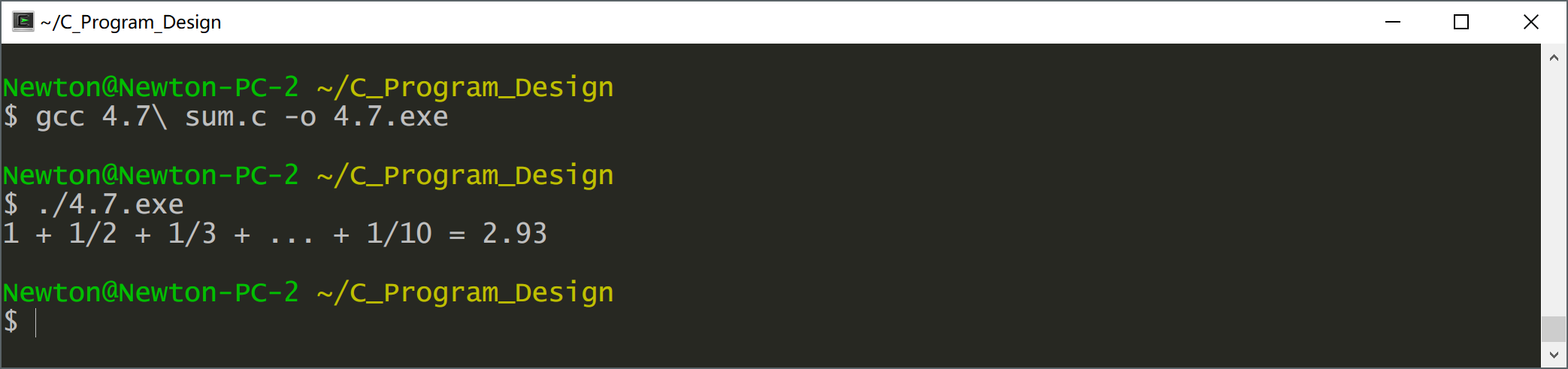
②用单步跟踪观察while语句的执行过程：连续按三次F8键，再用两次Ctrl-F7操作分别将i和sum的值显示出来，然后不按断F8键，每次按F8后，观察绿条的变化和变量值的变化情况，以此来分析并弄清while语句的执行过程。

③修改程序，实现

### 程序代码

|  |  |
| --- | --- |
| 1  2  3  4  5  6  7  8  9  10  11  12  13  14  15  16  17 | /\*  \* filename: 4.7 sum.c  \* property: test  \*/  #include <stdio.h>  int main**()** **{**  float i **=** 1**;**  float s **=** 0**;**  **for** **(;** i **<=** 10**;** i**++)** **{**  s **+=** 1 **/** i**;**  **}**  printf**(**"1 + 1/2 + 1/3 + ... + 1/10 = %3.2f\n"**,** s**);**  **return** 0**;**  **}** |

### 运行结果



## 计算级数

直到最后一项的绝对值小于为止（注意：是）。

具体要求如下：

①画出流程图。

②除了要输出级数和外，同时要求输出总的项数。输出形式为：

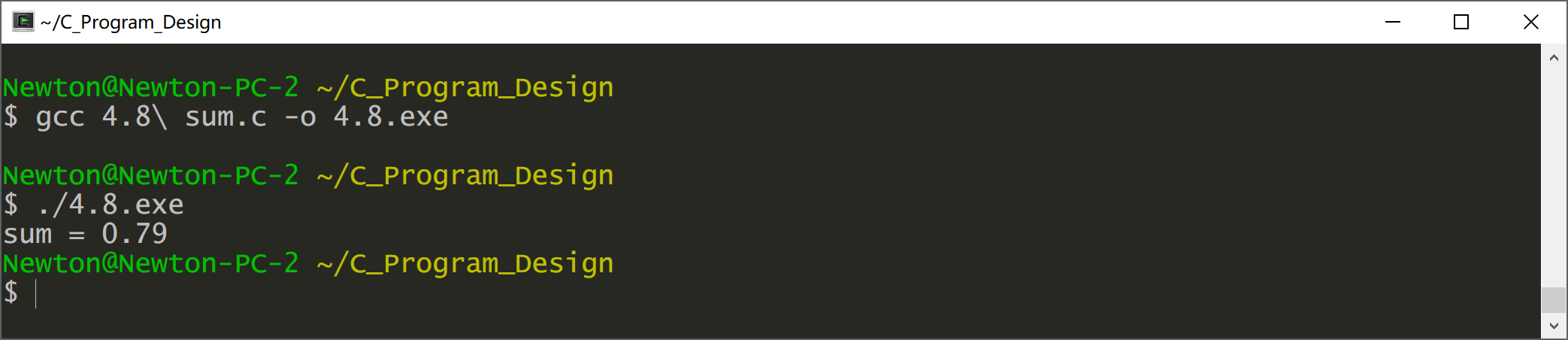
=具体值，=具体值



### 程序代码

|  |  |
| --- | --- |
| 1  2  3  4  5  6  7  8  9  10  11  12  13  14  15  16  17  18  19  20  21  22 | /\*  \* filename: 4.8 sum.c  \* property: exercise  \*/  #include <stdio.h>  #include <math.h>  int main**()** **{**  float i **=** 1**;**  float s **=** 1**;**  float test**;**  **for** **(;** **(**test **=** 1 **/** **(**2**\***i**+**1**))** **>=** 1e-4**;** i**++)** **{**  s **+=** pow**(-**1**,** i**)** **\*** test**;**  // printf("test = %2.5f\n", test);  **}**  printf**(**"sum = %3.2f"**,** s**);**  **return** 0**;**  **}** |

### 运行结果



## 完数之和

如果一个数恰好等于它的所有因子(包括但不包括自身)之和，例如：的因子为，，，且，因此是一个“完数”。计算并输出以内的所有“完数”之和。

具体要求如下:

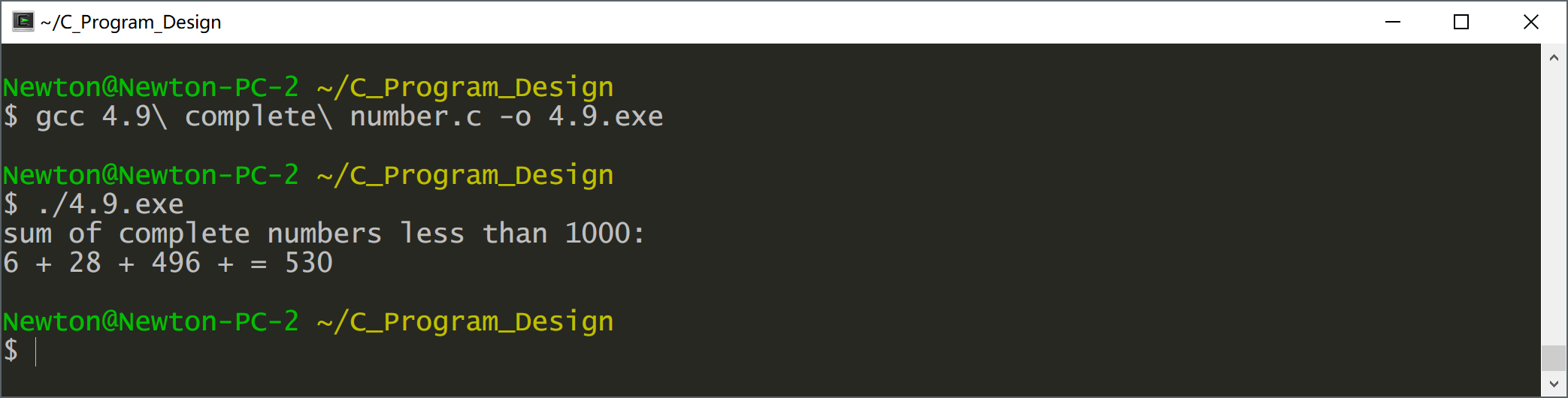
①所有循环均用for循环。

②输出要有文字说明，并同时输出各“完数”。输出形式为：完数1 + 完数2 + … =和值

### 程序代码

|  |  |
| --- | --- |
| 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 | /\*  \* filename: 4.9 complete number.c  \* property: exercise  \*/  #include <stdio.h>  int main**()** **{**  int i**,** j**,** sum\_1**,** sum\_2 **=** 0**;**  printf**(**"sum of complete numbers less than 1000:\n"**);**  **for(**i **=** 2**;**i **<=** 1000**;** i**++)** **{**  sum\_1 **=** 0**;**  **for(**j **=** 1**;**j **<=** i**-**1**;** j**++)** **{**  **if(**i **%** j**==**0**)** **{**  sum\_1 **=** sum\_1 **+** j**;**  **}**  **}**  **if(**sum\_1 **==** i**)** **{**  printf**(**"%d + "**,** i**);**  sum\_2 **=** sum\_2 **+** i**;**  **}**  **}**  printf**(**"= %d \n"**,** sum\_2**);**  **return** 0**;**  **}** |

### 运行结果



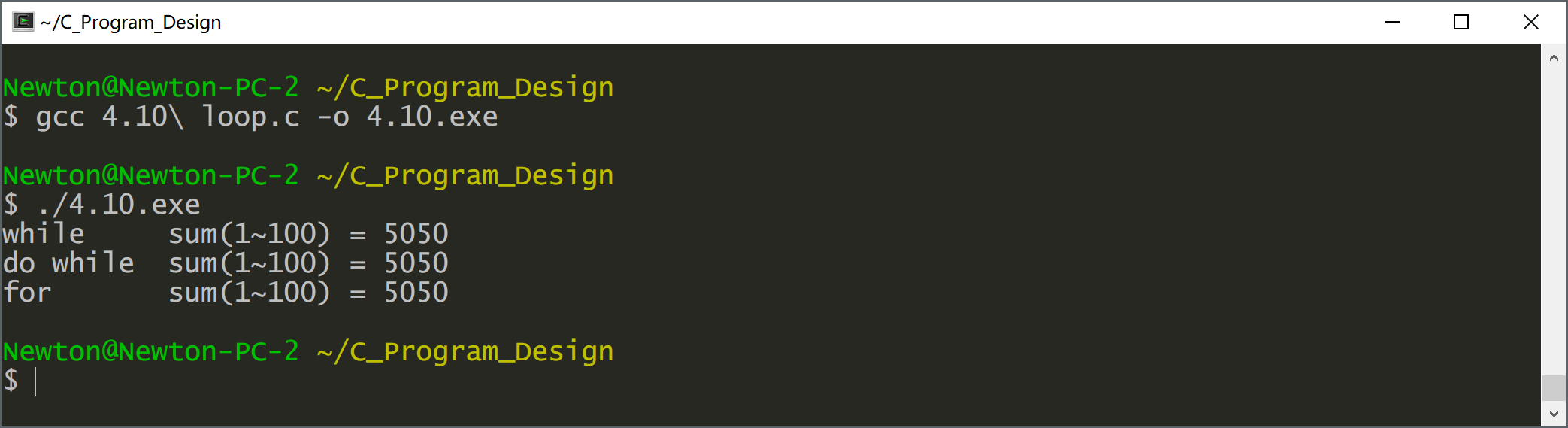
## 1题

分别用三种循环语句（while语句、do-while语句、for语句），实现求1～100的累加和。编程上机调试，总结出三种循环语句哪种实现起来方便、灵活。

### 程序代码

|  |  |
| --- | --- |
| 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 | /\*  \* filename: 4.10 loop.c  \* property: test  \*/  #include <stdio.h>  int main**()** **{**  int i **=** 1**;**  int s **=** 0**;**    // while  **while** **(**i **<=** 100**)** **{**  s **+=** i**++;**  **}**  printf**(**"while sum(1~100) = %d\n"**,** s**);**  i **=** 1**;**  s **=** 0**;**  // do while  **do** **{**  s **+=** i**++;**    **}** **while(**i **<=** 100**);**  printf**(**"do while sum(1~100) = %d\n"**,** s**);**    i **=** 0**;**  s **=** 0**;**  // for  **for(;** i **<=** 100**;** s **+=** i**++);**  printf**(**"for sum(1~100) = %d\n"**,** s**);**    **return** 0**;**  **}** |

### 运行结果



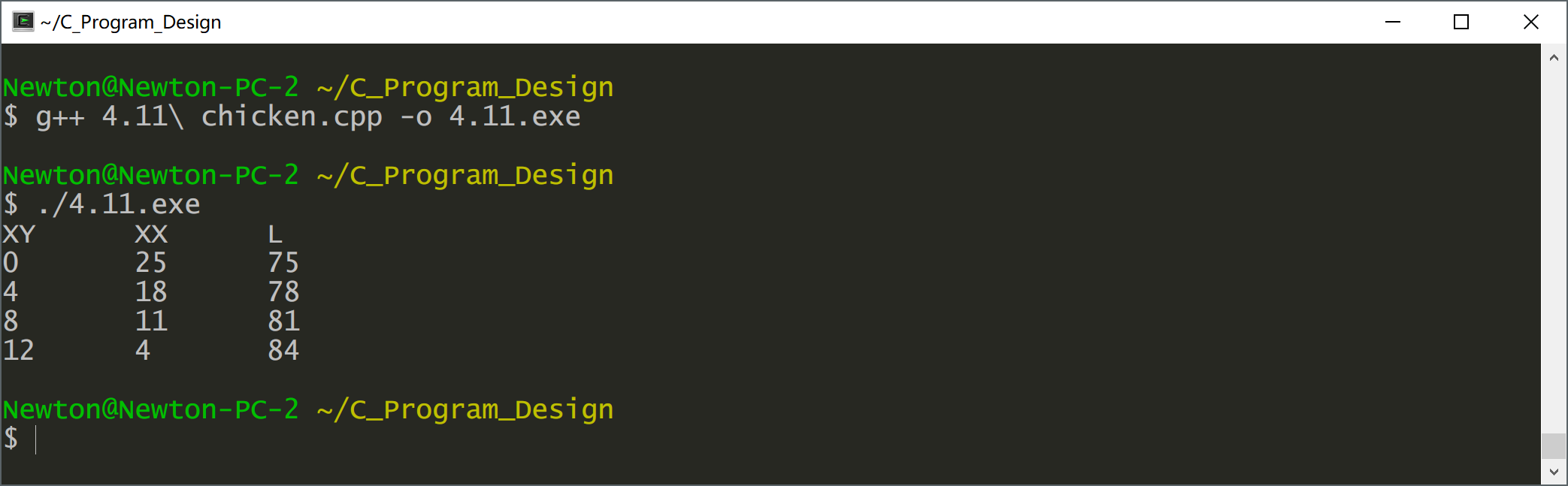
## 百元买鸡问题

已知公鸡每只5元，母鸡每只3元，小鸡1元3只，要求100元钱正好买100只鸡，则应买公鸡、母鸡的小鸡各多少只？

### 程序代码

|  |  |
| --- | --- |
| 1  2  3  4  5  6  7  8  9  10  11  12  13  14  15  16  17  18  19  20  21 | /\*  \* filename: 4.11 chicken.cpp  \* property: test  \*/  #include <iostream>  **using** **namespace** std**;**  int main**()** **{**  cout **<<** "XY\t" **<<** "XX\t" **<<** "L" **<<** endl**;**  int x**,** y**,** z**;**  **for(**x **=** 0**;** x **<=** 20**;** x**++)** **{**  **for(**y **=** 0**;** y **<=** 33**;** y**++)** **{**  z **=** 100 **-** x **-** y**;**  **if(**5**\***x **+** 3**\***y **+** z**/**3 **==** 100 **&&** z **%** 3 **==** 0**)**  cout **<<** x **<<** "\t" **<<** y **<<** "\t" **<<** z **<<** endl**;**  **}**  **}**  **return** 0**;**  **}** |

### 运行结果



## 成绩排序

某班有学生人，从终端读入及个人学生的成绩，试编程实现以下功能：

①印出前3个最好成绩及取得每个最好成绩的人数；

②若90分以上计为A，75－89分计为B，60－74分计为C，60分以下计为D，试统计各档成绩所占百分率。

### 程序代码

一般而言，这种动态数据类型都需要用指针来实现，对于初学者，往往用一个很大的数组，按照限制在其一个子块上进行操作就可以了。但是这种操作被认为是有害的，因为无法估计内存的情况，所以这个问题的实现放到后面的练习性实验报告里完成。到时候会涉及数据结构与算法的一些东西。

当然，这里也可以直接使用C++ STL的高级操作，但是作为初级报告，不再展开。

## 习题

* + - 1. 下列论题哪些是错误的？

C语言没有goto语句。（×）

While表达式语句的作用是：当表达式的值为0时重复执行循环体语句。（×）

do（语句）while（表达式）的作用是：重复执行循环体（“语句”），直到表达式成立（其值为真）。（×）

“do…while”语句中，写在do后面、While前面的若干语句，不必用花括号括起来。（×）

break语句用于退出条件语句和循环语句的判断。（√）

contiune语句表示将循环继续下去。

凡是while语句能解决的问题也能用do…while语句解决。（√）

凡是用while语句能解决的问题都可以用for语句实现。（√）

凡是用for语句能解决的问题都可以用while语句实现。（×）

造成“死循环“的主要原因是循环变量的值没有得到必要的修改。（√）

## 程序排错

### 排错1

#### 程序代码

|  |  |
| --- | --- |
| 1  2  3  4  5  6  7  8  9  10  11 | #include <stdio.h>  int main**()** **{**  int n**,** sum**;**  n **=** 1**;**  **while(**n**<**100**)** **{**  sum **+=** n**;**  n**++;**  **}**  printf**(**"sum=%f\n"**,** sum**);**  **return** 0**;**  **}** |

错误原因：数据类型设置错误。sum是浮点型。

### 排错2

从键盘输入若干学生的成绩（输入负分结束），输出平均成绩和最高分。

|  |  |
| --- | --- |
| 1  2  3  4  5  6  7  8  9  10  11  12  13  14  15  16  17  18 | #include <stdio.h>  int main**()** **{**  int n **=** 0**;**  float grade**,** sum**,** max **=** 0**;**  scanf**(**"%f"**,** **&**grade**);**  **while(**grade **>=** 0**)** **{**  **if(**grade **>** max**)** **{**  max **=** grade**;**  **}**  sum **=** sum **+** grade**;**  n **=** n **+** 1**;**  scanf**(**"%f"**,** **&**grade**);**  **}**  grade **=** sum **/** n**;**  printf**(**"max = %3.2f, a = %3.2f\n"**,** max**,** grade**);**  **return** 0**;**  **}** |

### 排错3

计算并输出超过1000的第一个值。

#### 程序代码

|  |  |
| --- | --- |
| 1  2  3  4  5  6  7  8  9  10  11 | #include <stdio.h>  int main**()** **{**  int n **=** 1**,** sum **=** 0**;**  **for(;** **;** n**++)**  sum **=** sum **+** **(**2 **\*** n **+** 1**);**  **if(**sum**>**2000**)**  **break;**  printf**(**"n = %d, sum = %d\n"**,** n**,** sum**);**  **return** 0**;**  **}** |

for循环的语句超过了1，所以要添加花括号

### 排错4

求2~1000之间的全部素数（每行显示10个数）。

#### 程序代码

|  |  |
| --- | --- |
| 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>  int main**()** **{**  int m **=** 3**,** k**,** i**,** n **=** 1**;**  printf**(**"%7d"**,** 2**);**  **do** **{**  **if(**n **%** 10 **=** 0**)** **{**  printf**(**"\n"**);**  k**=**sqrt**(**m**);**  **}**  **for(**i **=** 2**;** i **<=** k**;** i**++)**  **if(**m **%** i **==** 0**)**  **continue;**  **if(**i**>=**k**+**1**)** **{**  printf**(**"%8d"**,** m**);**  n**++;**  **}**  **}**  **while(**m **>** 1000**);**  printf**(**"\n"**);**  **return** 0**;**  **}** |

错误：

（1）没有使得循环变量更改的语句；

（2）循环条件设置错误。第23行中应该是m<1000.

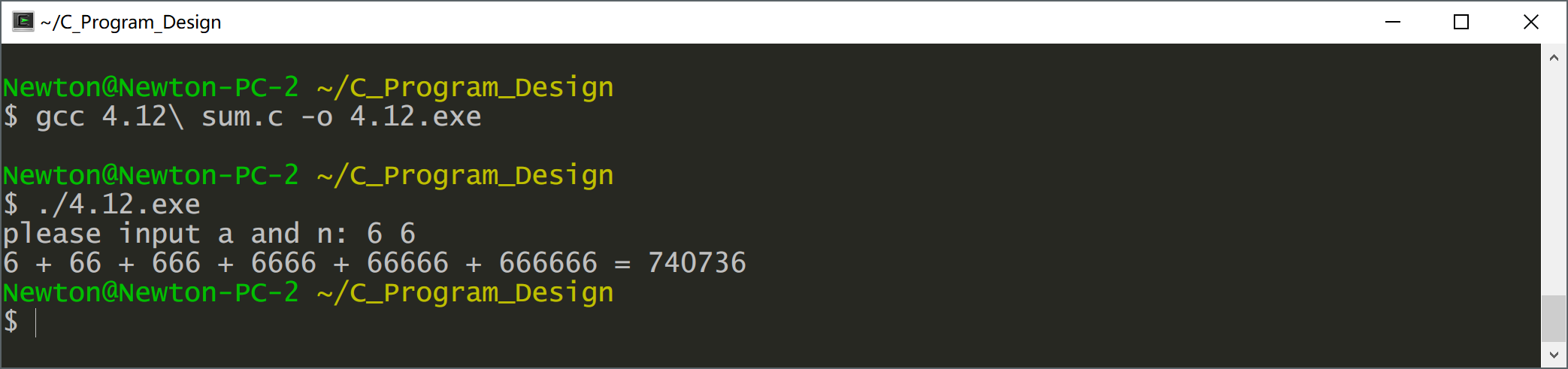
## 条件求和

，其中是1~9中的一个数字。为一正整数，和均从键盘输入。（例如输入为，为，）。

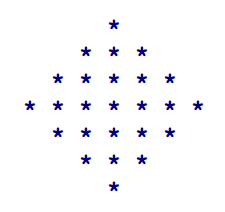
### 程序代码

|  |  |
| --- | --- |
| 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 | /\*  \* filename: 4.12 sum.c  \* property: exercise  \*/  #include <stdio.h>  #include <math.h>  int main**()** **{**  int a**,** n**;**  float i**,** j**,** count **=** 0**;**  printf**(**"please input a and n: "**);**  scanf**(**"%d %d"**,&**a**,&**n**);**  **for(**i **=** 1**;** i **<=** n**;** i**++)** **{**  **for(**j **=** 1**;** j **<=** i**;** j**++)** **{**  printf**(**"%d"**,** a**);**  count **=** count **+** pow**(**10**,** j**-**1**)** **\*** a**;**  **if((**j **==** i**)** **&&** **(**j**!=**n**))** **{**  printf**(**" + "**);**  **}**  **}**  **}**  printf**(**" = %5.0f"**,** count**);**  **return** 0**;**  **}** |

### 运行结果



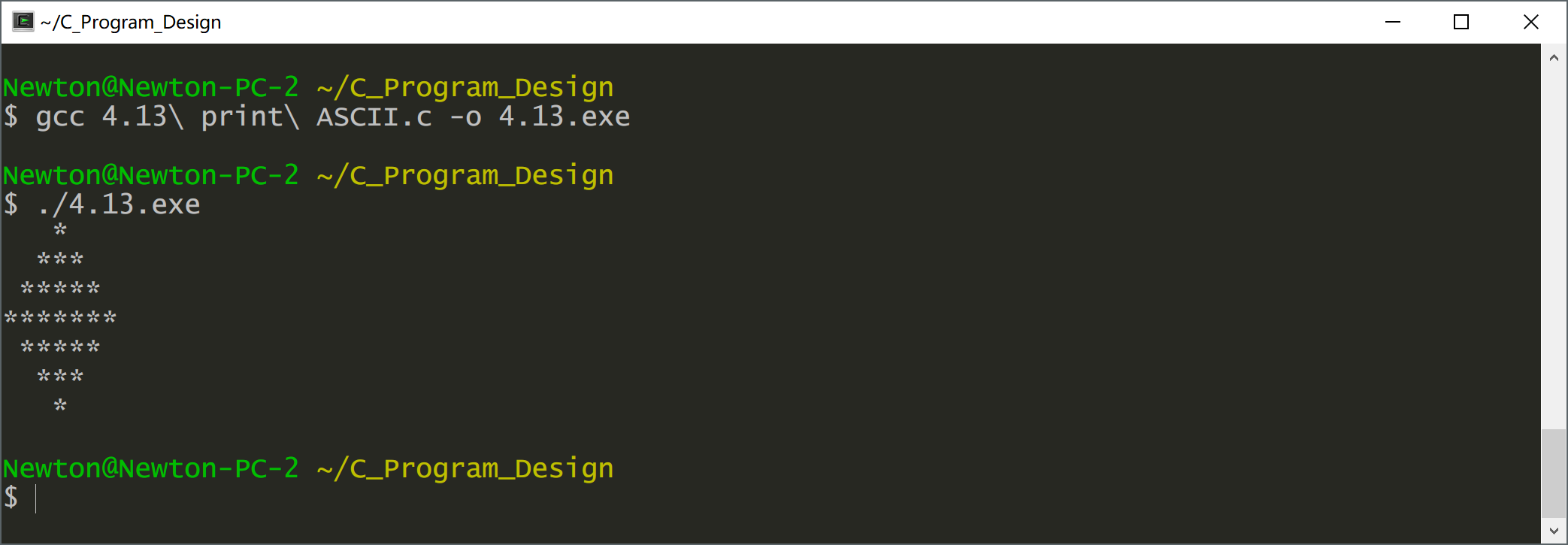
## 打印菱形图案



### 程序代码

|  |  |
| --- | --- |
| 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 | /\*  \* filename: 4.12 sum.c  \* property: exercise  \*/  #include <stdio.h>  #include <math.h>  int main**()** **{**  int a**,** n**;**  float i**,** j**,** count **=** 0**;**  printf**(**"please input a and n: "**);**  scanf**(**"%d %d"**,&**a**,&**n**);**  **for(**i **=** 1**;** i **<=** n**;** i**++)** **{**  **for(**j **=** 1**;** j **<=** i**;** j**++)** **{**  printf**(**"%d"**,** a**);**  count **=** count **+** pow**(**10**,** j**-**1**)** **\*** a**;**  **if((**j **==** i**)** **&&** **(**j**!=**n**))** **{**  printf**(**" + "**);**  **}**  **}**  **}**  printf**(**" = %5.0f"**,** count**);**  **return** 0**;**  **}** |

### 运行结果



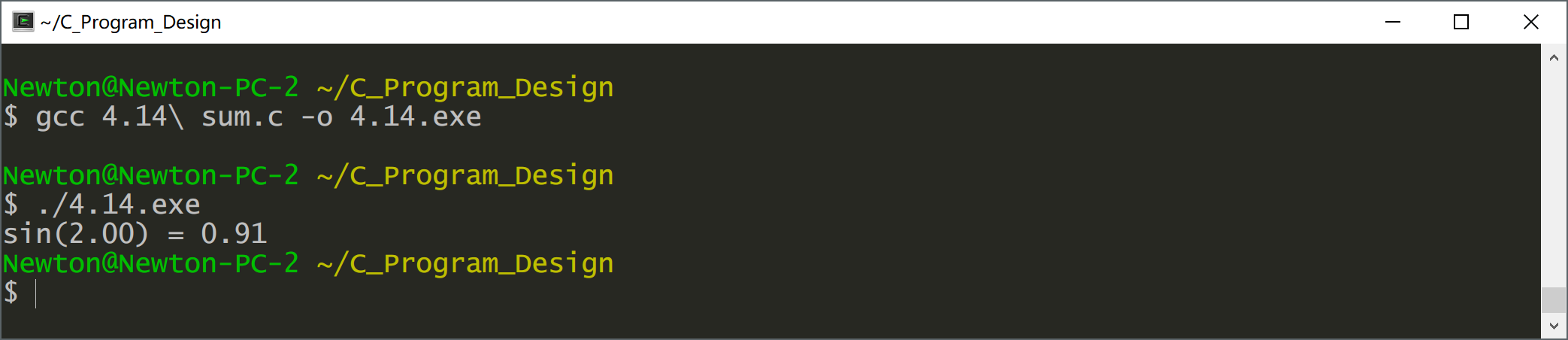
## 求和

的近似值，精确到。

### 程序代码

|  |  |
| --- | --- |
| 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 | /\*  \* filename: 4.14 sum.c  \* property: exercise  \*/  #include <stdio.h>  #include <math.h>  #define TYPE\_ERROR -1  double factorial**(**double in**)** **{**  **if(**floor**(**in**)** **!=** in**)** **{**  **return** **-**1**;**  **}**  double ans **=** 1**;**  **while(**in **!=** 1**)** **{**  ans **\*=** in**;**  in **-=** 1**;**  **}**  **return** ans**;**  **}**  int main**()** **{**  double i **=** 0**;**  double s **=** 0**;**  const double x **=** 2**;** // sin(\pi) = 0  double test**;**  double low **=** 1**;**  **for** **(;** **(**test **=** pow**(**x**,** 2**\***i**+**1**)/**factorial**(**2**\***i**+**1**))** **>=** 1e-6**;** i**++)** **{**  s **+=** pow**(-**1**,** i**)** **\*** test**;**  //printf("test = %3.4f\n", test);  **}**  printf**(**"sin(%3.2f) = %3.2f"**,** x**,** s**);**  **return** 0**;**  **}** |

### 运行结果



# 实验总结

循环问题，重在循环体的安排与设计。[1]

# 参考文献

1. Prata, S., *C++ Primer Plus*. 6th ed. 2012.

# 教师评语