第一章 程序设计和 C 语言 【第 15 页】

1-5

```
#include <stdio.h>
int main ( )
{    printf ("**************************\n\n");
     printf("              Very    Good!\n\n");
     printf ("**************************\n");
     return 0;
}
```

1-6

```
#include <stdio.h>
int main()
{int a,b,c,max;
  printf("please input a,b,c:\n");
  scanf("%d,%d,%d",&a,&b,&c);
  max=a;
  if (max<b)
     max=b;
  if (max<c)
     max=c;
  printf("The largest number is %d\n",max);
  return 0;
}
```

第 2 章算法——程序的灵魂 【第 36 页】暂无答案

第 3 章最简单的 C 程序设计——顺序程序设计 【第 82 页】

3-1

```
#include <stdio.h>
#include <math.h>
int main()
{float p,r,n;
  r=0.1;
  n=10;
  p=pow(1+r,n);
  printf("p=%f\n",p);
  return 0;
}
```

3-2-1

```
#include <stdio.h>
#include <math.h>
int main()
{float r5,r3,r2,r1,r0,p,p1,p2,p3,p4,p5;
  p=1000;
  r5=0.0585;
  r3=0.054;
  r2=0.0468;
  r1=0.0414;
  r0=0.0072;

  p1=p*((1+r5)*5);                  // 一次存 5 年期
  p2=p*(1+2*r2)*(1+3*r3);           // 先存 2 年期，到期后将本息再存 3 年期
  p3=p*(1+3*r3)*(1+2*r2);           // 先存 3 年期，到期后将本息再存 2 年期
  p4=p*pow(1+r1,5);                  // 存 1 年期，到期后将本息存再存 1 年期，连续存 5 次
  p5=p*pow(1+r0/4,4*5);             // 存活期存款。活期利息每一季度结算一次
  printf("p1=%f\n",p1);             // 输出按第 1 方案得到的本息和
  printf("p2=%f\n",p2);             // 输出按第 2 方案得到的本息和
  printf("p3=%f\n",p3);             // 输出按第 3 方案得到的本息和
  printf("p4=%f\n",p4);             // 输出按第 4 方案得到的本息和
  printf("p5=%f\n",p5);             // 输出按第 5 方案得到的本息和
  return 0;
}
```

3-2-2

```
#include <stdio.h>
#include <math.h>
int main()
{double r5,r3,r2,r1,r0,p,p1,p2,p3,p4,p5;
  p=1000;
  r5=0.0585;
  r3=0.054;
  r2=0.0468;
  r1=0.0414;
  r0=0.0072;

  p1=p*((1+r5)*5);                  // 一次存 5 年期
  p2=p*(1+2*r2)*(1+3*r3);           // 先存 2 年期，到期后将本息再存 3 年期
  p3=p*(1+3*r3)*(1+2*r2);           // 先存 3 年期，到期后将本息再存 2 年期
  p4=p*pow(1+r1,5);                  // 存 1 年期，到期后将本息存再存 1 年期，连续存 5 次
  p5=p*pow(1+r0/4,4*5);             // 存活期存款。活期利息每一季度结算一次
  printf("p1=%f\n",p1);             // 输出按第 1 方案得到的本息和
```

```
  printf("p2=%f\n",p2);          // 输出按第 2 方案得到的本息和
  printf("p3=%f\n",p3);          // 输出按第 3 方案得到的本息和
  printf("p4=%f\n",p4);          // 输出按第 4 方案得到的本息和
  printf("p5=%f\n",p5);          // 输出按第 5 方案得到的本息和
  return 0;
}
```

3-2-3

```
#include <stdio.h>
#include <math.h>
int main()
{float r5,r3,r2,r1,r0,p,p1,p2,p3,p4,p5;
  p=1000;
  r5=0.0585;
  r3=0.054;
  r2=0.0468;
  r1=0.0414;
  r0=0.0072;

  p1=p*((1+r5)*5);                  // 一次存 5 年期
  p2=p*(1+2*r2)*(1+3*r3);           // 先存 2 年期，到期后将本息再存 3 年期
  p3=p*(1+3*r3)*(1+2*r2);           // 先存 3 年期，到期后将本息再存 2 年期
  p4=p*pow(1+r1,5);                 // 存 1 年期，到期后将本息存再存 1 年期，连续存 5 次
  p5=p*pow(1+r0/4,4*5);             // 存活期存款。活期利息每一季度结算一次
  printf("p1=%10.2f\n",p1);         // 输出按第 1 方案得到的本息和
  printf("p2=%10.2f\n",p2);         // 输出按第 2 方案得到的本息和
  printf("p3=%10.2f\n",p3);         // 输出按第 3 方案得到的本息和
  printf("p4=%10.2f\n",p4);         // 输出按第 4 方案得到的本息和
  printf("p5=%10.2f\n",p5);         // 输出按第 5 方案得到的本息和
  return 0;
}
```

3-3.

```
#include <stdio.h>
#include <math.h>
int main()
{float d=300000,p=6000,r=0.01,m;
  m=log10(p/(p-d*r))/log10(1+r);
  printf("m=%6.2f\n",m);
  return 0;
}
```

3-4

```
#include <stdio.h>
int main()
{int c1,c2;
  c1=197;
  c2=198;
  printf("c1=%c,c2=%c\n",c1,c2);
  printf("c1=%d，c2=%d\n",c1,c2);
  return 0;
}
```

3-5

```
#include <stdio.h>
int main()
{int a,b;
  float x,y;
  char c1,c2;
  scanf("a=%d b=%d",&a,&b);
  scanf("%f %e",&x,&y);
  scanf("%c%c",&c1,&c2);
  printf("a=%d,b=%d,x=%f,y=%f,c1=%c,c2=%c\n",a,b,x,y,c1,c2);
  return 0;
}
```

3-6

```
#include <stdio.h>
int main()
{char c1='C',c2='h',c3='i',c4='n',c5='a';
  c1=c1+4;
  c2=c2+4;
  c3=c3+4;
  c4=c4+4;
  c5=c5+4;
  printf("passwor is %c%c%c%c%c\n",c1,c2,c3,c4,c5);
  return 0;
}
```

3-7

```
#include <stdio.h>
int main ()
{float h,r,l,s,sq,vq,vz;
 float pi=3.141526;
 printf("请输入圆半径 r，圆柱高 h：");
 scanf("%f,%f",&r,&h);                    //要求输入圆半径 r 和圆柱高 h
 l=2*pi*r;                                //计算圆周长 l
 s=r*r*pi;                                //计算圆面积 s
 sq=4*pi*r*r;                             //计算圆球表面积 sq
 vq=3.0/4.0*pi*r*r*r;                     //计算圆球体积 vq
 vz=pi*r*r*h;                             //计算圆柱体积 vz
 printf("圆周长为:        l=%6.2f\n",l);
 printf("圆面积为:        s=%6.2f\n",s);
 printf("圆球表面积为:    sq=%6.2f\n",sq);
 printf("圆球体积为:      v=%6.2f\n",vq);
 printf("圆柱体积为:      vz=%6.2f\n",vz);
 return 0;
 }
```

3-8-1

```
#include <stdio.h>
int main()
{
   int c1,c2;                               //整型定义
   printf("请输入两个整数 c1,c2:");
   scanf("%d,%d",&c1,&c2);
   printf("按字符输出结果:\n");
   printf("%c,%c\n",c1,c2);
   printf("按 ASCII 码输出结果为:\n");
   printf("%d,%d\n",c1,c2);
   return 0;
}
```

3-8-2

```
#include <stdio.h>
int main()
{
   char c1,c2;                              //定义字符型变量
   int i1,i2;                               //定义整型变量
   printf("请输入两个字符 c1,c2:");
   scanf("%c,%c",&c1,&c2);
   i1=c1;                                   //赋值给整型变量
   i2=c2;
```

```
    printf("按字符输出结果:\n");
    printf("%c,%c\n",i1,i2);
    printf("按整数输出结果:\n");
    printf("%d,%d\n",c1,c2);
    return 0;
}
```

3-8-3

```
#include <stdio.h>
int main()
{
    char c1,c2;                                          //定义为字符型
    int i1,i2;                                           //定义为整型
    printf("请输入两个整数 i1,i2:");
    scanf("%d,%d",&i1,&i2);
    c1=i1;                                               //将整数赋值给字符变量
    c2=i2;
    printf("按字符输出结果:\n");
    printf("%c,%c\n",c1,c2);
    printf("按整数输出结果:\n");
    printf("%d,%d\n",c1,c2);
    return 0;
}
```

3-8

```
#include <stdio.h>
int main()
{
char c1,c2;
printf("请输入两个字符 c1,c2:");
c1=getchar();
c2=getchar();
printf("用 putchar 语句输出结果为:");
putchar(c1);
putchar(c2);
printf("\n");
printf("用 printf 语句输出结果为:");
printf("%c %c\n",c1,c2);
return 0;
}
```

第 4 章选择结构程序设计【第 111 页】

4-4-1

```
#include <stdio.h>
```

```
int main()
{
   int a,b,c;
   printf("请输入三个整数:");
   scanf("%d,%d,%d",&a,&b,&c);
   if (a<b)
      if (b<c)
         printf("max=%d\n",c);
      else
         printf("max=%d\n",b);
   else if (a<c)
         printf("max=%d\n",c);
       else
         printf("max=%d\n",a);
   return 0;
}
```

4-4-2

```
#include <stdio.h>
int main()
{ int a,b,c,temp,max;
   printf("请输入三个整数:");
   scanf("%d,%d,%d",&a,&b,&c);
   temp=(a>b)?a:b;                          /*将 a 和 b 中的大者存入 temp 中*/
   max=(temp>c)?temp:c;                     /*将 a 和 b 中的大者与 c 比较,取最大者*/
   printf("三个整数的最大数是%d\n",max);
   return 0;
}
```

4-5-2

```
#include <stdio.h>
#include <math.h>
#define M 1000
int main()
{
   int i,k;
   printf("请输入一个小于%d 的整数 i:",M);
   scanf("%d",&i);
   while (i>M)
   {printf("输入的数不符合要求，请重新输入一个小于%d 的整数 i:",M);
    scanf("%d",&i);
   }
   k=sqrt(i);
```

```
   printf("%d 的平方根的整数部分是：%d\n",i,k);
   return 0;
}
```

4-5

```
#include <stdio.h>
#include <math.h>
#define M 1000
int main()
{
   int i,k;
   printf("请输入一个小于%d 的整数 i:",M);
   scanf("%d",&i);
   if (i>M)
   {printf("输入的数不符合要求，请重新输入一个小于%d 的整数 i:",M);
    scanf("%d",&i);
   }
   k=sqrt(i);
   printf("%d 的平方根的整数部分是：%d\n",i,k);
   return 0;
}
```

4-6.

```
#include <stdio.h>
int main()
{ int x,y;
   printf("输入 x:");
   scanf("%d",&x);
   if(x<1)                          /* x<1 */
   { y=x;
      printf("x=%3d,     y=x=%d\n" ,x,y);
      }
   else    if(x<10)              /* 1=<x<10 */
      { y=2*x-1;
        printf("x=%d,    y=2*x-1=%d\n",x,y);
      }
      else                      /* x>=10    */
      { y=3*x-11;
        printf("x=%d,    y=3*x-11=%d\n",x,y);
      }
   return 0;
}
```

4-7-1

```
#include <stdio.h>
int main()
{
   int x,y;
   printf("enter x:");
   scanf("%d",&x);
   y=-1;
   if(x!=0)
      if(x>0)
         y=1;
   else
      y=0;
  printf("x=%d,y=%d\n",x,y);
  return 0;
}
```

4-7-2

```
#include <stdio.h>
int main()
{
   int x,y;
   printf("please enter x:");
   scanf("%d",&x);
   y=0;
   if(x>=0)
      if(x>0) y=1;
   else    y=-1;
  printf("x=%d,y=%d\n",x,y);
  return 0;
}
```

4-8

```
#include <stdio.h>
int main()
   { float score;
      char grade;
      printf("请输入学生成绩:");
      scanf("%f",&score);
      while (score>100||score<0)
```

```
    {printf("\n 输入有误,请重输");
    scanf("%f",&score);
    }
    switch((int)(score/10))
        {case 10:
    case 9: grade='A';break;
    case 8: grade='B';break;
    case 7: grade='C';break;
    case 6: grade='D';break;
    case 5:
    case 4:
    case 3:
    case 2:
    case 1:
    case 0: grade='E';
    }
    printf("成绩是 %5.1f,相应的等级是%c\n ",score,grade);
    return 0;
}
```

4-9

```
#include <stdio.h>
#include <math.h>
int main()
{
   int  num,indiv,ten,hundred,thousand,ten_thousand,place;          //分别代表个位,十位,百位,千位,万位和位数
   printf("请输入一个整数(0-99999):");
   scanf("%d",&num);
   if (num>9999)
           place=5;
   else    if (num>999)
           place=4;
   else    if (num>99)
           place=3;
   else    if (num>9)
           place=2;
   else place=1;
   printf("位数:%d\n",place);
   printf("每位数字为:");
   ten_thousand=num/10000;
   thousand=(int)(num-ten_thousand*10000)/1000;
   hundred=(int)(num-ten_thousand*10000-thousand*1000)/100;
```

```
   ten=(int)(num-ten_thousand*10000-thousand*1000-hundred*100)/10;
   indiv=(int)(num-ten_thousand*10000-thousand*1000-hundred*100-ten*10);
   switch(place)
      {case 5:printf("%d,%d,%d,%d,%d",ten_thousand,thousand,hundred,ten,indiv);
            printf("\n 反序数字为:");
            printf("%d%d%d%d%d\n",indiv,ten,hundred,thousand,ten_thousand);
            break;
       case 4:printf("%d,%d,%d,%d",thousand,hundred,ten,indiv);
            printf("\n 反序数字为:");
            printf("%d%d%d%d\n",indiv,ten,hundred,thousand);
            break;
       case 3:printf("%d,%d,%d",hundred,ten,indiv);
            printf("\n 反序数字为:");
            printf("%d%d%d\n",indiv,ten,hundred);
            break;
       case 2:printf("%d,%d",ten,indiv);
            printf("\n 反序数字为:");
            printf("%d%d\n",indiv,ten);
            break;
       case 1:printf("%d",indiv);
            printf("\n 反序数字为:");
            printf("%d\n",indiv);
            break;
        }
   return 0;
  }
```

4-10-1

```
#include <stdio.h>
int main()
{
   int i;
   double bonus,bon1,bon2,bon4,bon6,bon10;
   bon1=100000*0.1;
   bon2=bon1+100000*0.075;
   bon4=bon2+100000*0.05;
   bon6=bon4+100000*0.03;
   bon10=bon6+400000*0.015;
   printf("请输入利润 i:");
   scanf("%d",&i);
   if (i<=100000)
       bonus=i*0.1;
   else if (i<=200000)
```

```
        bonus=bon1+(i-100000)*0.075;
    else if (i<=400000)
        bonus=bon2+(i-200000)*0.05;
    else if (i<=600000)
        bonus=bon4+(i-400000)*0.03;
    else if (i<=1000000)
        bonus=bon6+(i-600000)*0.015;
    else
        bonus=bon10+(i-1000000)*0.01;
    printf("奖金是: %10.2f\n",bonus);
    return 0;
  }
```

4-10-2

```
#include <stdio.h>
int main()
{
    int i;
    double    bonus,bon1,bon2,bon4,bon6,bon10;
    int    branch;
    bon1=100000*0.1;
    bon2=bon1+100000*0.075;
    bon4=bon2+200000*0.05;
    bon6=bon4+200000*0.03;
    bon10=bon6+400000*0.015;
    printf("请输入利润 i:");
    scanf("%d",&i);
    branch=i/100000;
    if (branch>10)    branch=10;
    switch(branch)
    {    case 0:bonus=i*0.1;break;
         case 1:bonus=bon1+(i-100000)*0.075;break;
         case 2:
         case 3: bonus=bon2+(i-200000)*0.05;break;
         case 4:
         case 5: bonus=bon4+(i-400000)*0.03;break;
         case 6:
         case 7:
         case 8:
         case 9: bonus=bon6+(i-600000)*0.015;break;
         case 10: bonus=bon10+(i-1000000)*0.01;
    }
      printf("奖金是  %10.2f\n",bonus);
```

```
    return 0;
  }
```

4-11

```
#include <stdio.h>
int main()
  {int    t,a,b,c,d;
   printf("请输入四个数:");
   scanf("%d,%d,%d,%d",&a,&b,&c,&d);
   printf("a=%d,b=%d,c=%d,d=%d\n",a,b,c,d);
   if (a>b)
      { t=a;a=b;b=t;}
   if (a>c)
      { t=a;a=c;c=t;}
   if (a>d)
      { t=a;a=d;d=t;}
   if (b>c)
      { t=b;b=c;c=t;}
   if (b>d)
      { t=b;b=d;d=t;}
   if (c>d)
      { t=c;c=d;d=t;}
   printf("排序结果如下: \n");
   printf("%d   %d   %d   %d   \n"     ,a,b,c,d);
   return 0;
  }
```

4-12

```
#include <stdio.h>
int main()
  {
   int    h=10;
   float x1=2,y1=2,x2=-2,y2=2,x3=-2,y3=-2,x4=2,y4=-2,x,y,d1,d2,d3,d4;
   printf("请输入一个点(x,y):");
   scanf("%f,%f",&x,&y);
   d1=(x-x4)*(x-x4)+(y-y4)*(y-y4);                    /*求该点到各中心点距离*/
   d2=(x-x1)*(x-x1)+(y-y1)*(y-y1);
   d3=(x-x2)*(x-x2)+(y-y2)*(y-y2);
   d4=(x-x3)*(x-x3)+(y-y3)*(y-y3);
   if (d1>1 && d2>1 && d3>1 && d4>1)      h=0;    /*判断该点是否在塔外*/
   printf("该点高度为  %d\n",h);
   return 0;
```

```
}
```

第 5 章循环结构程序设计【第 140 页】

5-2

```
#include <stdio.h>
#include <math.h>                          // 程序中用到数学函数 fabs，应包含头文件 math.n
int main()
{
   int sign=1,count=0;                     // sign 用来表示数值的符号,count 用来统计循环次数
   double pi=0.0,n=1.0,term=1.0;           // pi 开始代表多项式的值，最后代表π的值, n 代表分母，
term 代表当前项的值
   while(fabs(term)>=1e-8)                 // 检查当前项 term 的绝对值是否大于或等于 10 的(-6)
次方
   {
     pi=pi+term;                           // 把当前项 term 累加到 pi 中
     n=n+2;                                // n+2 是下一项的分母
     sign=-sign;                           // sign 代表符号，下一项的符号与上一项符号相反
     term=sign/n;                          // 求出下一项的值 term
     count++;                              // count 累加 1
   }
   pi=pi*4;                                // 多项式的和 pi 乘以 4，才是π的近似值
   printf("pi=%10.8f\n",pi);               // 输出π的近似值
   printf("count=%d\n",count);             // 输出循环次数
   return 0;
}
```

5-3

```
#include <stdio.h>
int main()
 {
   int    p,r,n,m,temp;
   printf("请输入两个正整数 n,m:");
   scanf("%d,%d,",&n,&m);
   if (n<m)
     {
       temp=n;
       n=m;
       m=temp;
     }
   p=n*m;
   while(m!=0)
     {
       r=n%m;
       n=m;
```

```
      m=r;
    }
  printf("它们的最大公约数为:%d\n",n);
  printf("它们的最小公约数为:%d\n",p/n);
  return 0;
 }
```

5-4

```
#include <stdio.h>
int main()
 {
  char c;
  int letters=0,space=0,digit=0,other=0;
  printf("请输入一行字符:\n");
  while((c=getchar())!='\n')
   {
     if (c>='a' && c<='z' || c>='A' && c<='Z')
         letters++;
     else if (c==' ')
         space++;
     else if (c>='0' && c<='9')
         digit++;
     else
         other++;
    }
   printf("字母数:%d\n 空格数:%d\n 数字数:%d\n 其它字符数:%d\n",letters,space,digit,other);
   return 0;
  }
```

5-5

```
#include <stdio.h>
int main()
 {
  int   a,n,i=1,sn=0,tn=0;
  printf("a,n=:");
  scanf("%d,%d",&a,&n);
  while (i<=n)
  {
  tn=tn+a;   /*赋值后的 tn 为 i 个  a 组成数的值*/
  sn=sn+tn; /*赋值后的 sn 为多项式前 i 项之和*/
  a=a*10;
  ++i;
  }
  printf("a+aa+aaa+...=%d\n",sn);
```

```
   return 0;
   }
```

5-6

```
#include <stdio.h>
int main()

 {double s=0,t=1;
   int n;
   for (n=1;n<=20;n++)
   {
     t=t*n;
     s=s+t;
   }
   printf("1!+2!+...+20!=%22.15e\n",s);
   return 0;
}
```

5-7

```
#include <stdio.h>
int main()
 {
   int n1=100,n2=50,n3=10;
   double k,s1=0,s2=0,s3=0;
   for (k=1;k<=n1;k++)    /*计算 1 到 100 的和*/
      {s1=s1+k;}
   for (k=1;k<=n2;k++)    /*计算 1 到 50 各数的平方和*/
      {s2=s2+k*k;}
   for (k=1;k<=n3;k++)    /*计算 1 到 10 的各倒数和*/
      {s3=s3+1/k;}
   printf("sum=%15.6f\n",s1+s2+s3);
   return 0;
 }
```

5-8

```
#include <stdio.h>
int main()
 {
   int i,j,k,n;
   printf("parcissus numbers are ");
   for (n=100;n<1000;n++)
     {
```

```
    i=n/100;
    j=n/10-i*10;
    k=n%10;
    if (n==i*i*i + j*j*j + k*k*k)
       printf("%d ",n);
  }
 printf("\n");
 return 0;
 }
```

5-9-1

```
#define M 1000                    /*定义寻找范围*/
#include <stdio.h>
int main()
 {
  int k1,k2,k3,k4,k5,k6,k7,k8,k9,k10;
  int i,a,n,s;
  for (a=2;a<=M;a++)          /* a 是 2-1000 之间的整数，检查它是否完数 */
   {n=0;                       /* n 用来累计 a 的因子的个数 */
    s=a;                       /* s 用来存放尚未求出的因子之和，开始时等于 a */
     for (i=1;i<a;i++)       /* 检查 i 是否 a 的因子 */
        if (a%i==0)            /* 如果 i 是 a 的因子 */
    {n++;                      /* n 加 1，表示新找到一个因子 */
     s=s-i;                    /* s 减去已找到的因子，s 的新值是尚未求出的因子之和 */
     switch(n)                 /* 将找到的因子赋给 k1...k9，或 k10 */
      {case 1:
           k1=i;   break;     /* 找出的第 1 个因子赋给 k1 */
       case 2:
           k2=i;   break;     /* 找出的第 2 个因子赋给 k2 */
       case 3:
           k3=i;   break;     /* 找出的第 3 个因子赋给 k3 */
       case 4:
           k4=i;   break;     /* 找出的第 4 个因子赋给 k4 */
       case 5:
           k5=i;   break;     /* 找出的第 5 个因子赋给 k5 */
       case 6:
           k6=i;   break;      /* 找出的第 6 个因子赋给 k6 */
       case 7:
           k7=i;   break;     /* 找出的第 7 个因子赋给 k7 */
       case 8:
           k8=i;   break;      /* 找出的第 8 个因子赋给 k8 */
       case 9:
           k9=i;   break;     /*找出的第 9 个因子赋给 k9 */
```

```
            case 10:
                k10=i;   break;     /*  找出的第 10 个因子赋给 k10 */
           }
       }
       if (s==0)
       {
        printf("%d ,Its factors are ",a);
        if (n>1)    printf("%d,%d",k1,k2);          /* n>1 表示 a 至少有 2 个因子  */
        if (n>2)    printf(",%d",k3);               /* n>2 表示至少有 3 个因子，故应再输出一个
因子  */
        if (n>3)    printf(",%d",k4);               /* n>3 表示至少有 4 个因子，故应再输出一个
因子  */
        if (n>4)    printf(",%d",k5);               /*   以下类似  */
        if (n>5)    printf(",%d",k6);
        if (n>6)    printf(",%d",k7);
        if (n>7)    printf(",%d",k8);
        if (n>8)    printf(",%d",k9);
        if (n>9)    printf(",%d",k10);
        printf("\n");
       }
     }
     return 0;
  }
```

5-9-2

```
#include <stdio.h>
int main()
  {int m,s,i;
    for (m=2;m<1000;m++)
       {s=0;
        for (i=1;i<m;i++)
           if ((m%i)==0) s=s+i;
        if(s==m)
          {printf("%d,its factors are ",m);
           for (i=1;i<m;i++)
        if (m%i==0)    printf("%d ",i);
           printf("\n");
           }
       }
    return 0;
  }
```

5-10

```
#include <stdio.h>
int main()
  {
    int i,n=20;
    double a=2,b=1,s=0,t;
    for (i=1;i<=n;i++)
      {
      s=s+a/b;
      t=a,
      a=a+b,
      b=t;
      }
      printf("sum=%16.10f\n",s);
      return 0;
      }
```

5-11

```
#include <stdio.h>
int main()
  {
    double sn=100,hn=sn/2;
    int n;
    for (n=2;n<=10;n++)
      {
      sn=sn+2*hn;      /*第 n 次落地时共经过的米数*/
      hn=hn/2;         /*第 n 次反跳高度*/
      }
      printf("第 10 次落地时共经过%f 米\n",sn);
      printf("第 10 次反弹%f 米\n",hn);
      return 0;
      }
```

5-12

```
#include <stdio.h>
int main()
  {
    int day,x1,x2;
    day=9;
    x2=1;
    while(day>0)
      {x1=(x2+1)*2;       /*第 1 天的桃子数是第 2 天桃子数加 1 后的 2 倍.*/
        x2=x1;
        day--;
```

```
    }
   printf("total=%d\n",x1);
   return 0;
  }
```

5-13

```
#include <stdio.h>
 #include <math.h>
int main()
 {
   float a,x0,x1;
   printf("enter a positive number:");
   scanf("%f",&a);
   x0=a/2;
   x1=(x0+a/x0)/2;
   do
     {x0=x1;
      x1=(x0+a/x0)/2;
     }while(fabs(x0-x1)>=1e-5);
   printf("The square root of %5.2f    is %8.5f\n",a,x1);
   return 0;
 }
```

5-14

```
#include <stdio.h>
 #include <math.h>
int    main()
 {double x1,x0,f,f1;
   x1=1.5;
   do
     {x0=x1;
      f=((2*x0-4)*x0+3)*x0-6;
      f1=(6*x0-8)*x0+3;
      x1=x0-f/f1;
     }while(fabs(x1-x0)>=1e-5);
   printf("The root of equation is %5.2f\n",x1);
   return 0;
 }
```

5-15

```
#include <stdio.h>
 #include <math.h>
```

```
int main()
  {float x0,x1,x2,fx0,fx1,fx2;
    do
      {printf("enter x1 & x2:");
        scanf("%f,%f",&x1,&x2);
        fx1=x1*((2*x1-4)*x1+3)-6;
        fx2=x2*((2*x2-4)*x2+3)-6;
      }while(fx1*fx2>0);
    do
      {x0=(x1+x2)/2;
        fx0=x0*((2*x0-4)*x0+3)-6;
        if ((fx0*fx1)<0)
          {x2=x0;
            fx2=fx0;
          }
      else
      {x1=x0;
          fx1=fx0;
        }
      }while(fabs (fx0)>=1e-5);
    printf("x=%6.2f\n",x0);
    return 0;
  }
```

5-16

```
#include <stdio.h>
int main()
  {int i,j,k;
    for (i=0;i<=3;i++)
      {for (j=0;j<=2-i;j++)
           printf(" ");
        for (k=0;k<=2*i;k++)
           printf("*");
        printf("\n");
      }
    for (i=0;i<=2;i++)
      {for (j=0;j<=i;j++)
            printf(" ");
        for (k=0;k<=4-2*i;k++)
            printf("*");
        printf("\n");
      }
      return 0;
```

```
   }
```

5-17

```
#include <stdio.h>
int main()
 {
   char i,j,k;                    /*是 a 的对手;j 是 b 的对手;k 是 c 的对手*/
   for (i='x';i<='z';i++)
      for (j='x';j<='z';j++)
        if (i!=j)
           for (k='x';k<='z';k++)
      if (i!=k && j!=k)
         if (i!='x' && k!='x' && k!='z')
            printf("A--%c\nB--%c\nC--%c\n",i,j,k);
     return 0;

   }
```

第 6 章利用数组处理批量数据 【第 168 页】

6-1

```
#include <stdio.h>
#include <math.h>
int main()
{int i,j,n,a[101];
   for (i=1;i<=100;i++)
         a[i]=i;
   a[1]=0;
   for (i=2;i<sqrt(100);i++)
      for (j=i+1;j<=100;j++)
          {if(a[i]!=0 && a[j]!=0)
              if (a[j]%a[i]==0)
                 a[j]=0;
          }
   printf("\n");
   for (i=2,n=0;i<=100;i++)
      { if(a[i]!=0)
          {printf("%5d",a[i]);
            n++;
          }
        if(n==10)
          {printf("\n");
            n=0;
          }
```

```
        }
    printf("\n");
    return 0;
}
```

6-2

```
#include <stdio.h>
int main()
{int i,j,min,temp,a[11];
    printf("enter data:\n");
    for (i=1;i<=10;i++)
      {printf("a[%d]=",i);
        scanf("%d",&a[i]);
      }
    printf("\n");
    printf("The orginal numbers:\n");
    for (i=1;i<=10;i++)
        printf("%5d",a[i]);
    printf("\n");
    for (i=1;i<=9;i++)
        {min=i;
          for (j=i+1;j<=10;j++)
        if (a[min]>a[j]) min=j;
          temp=a[i];
          a[i]=a[min];
          a[min]=temp;
        }
    printf("\nThe sorted numbers:\n");
    for (i=1;i<=10;i++)
        printf("%5d",a[i]);
    printf("\n");
    return 0;
  }
```

6-3

```
#include <stdio.h>
int main()
{
int a[3][3],sum=0;
int i,j;
    printf("enter data:\n");
    for (i=0;i<3;i++)
        for (j=0;j<3;j++)
```

```
        scanf("%3d",&a[i][j]);
    for (i=0;i<3;i++)
       sum=sum+a[i][i];
    printf("sum=%6d\n",sum);
    return 0;
}
```

6-4

```
#include <stdio.h>
int main()
{ int a[11]={1,4,6,9,13,16,19,28,40,100};
    int temp1,temp2,number,end,i,j;
    printf("array a:\n");
    for (i=0;i<10;i++)
       printf("%5d",a[i]);
    printf("\n");
    printf("insert data:");
    scanf("%d",&number);
    end=a[9];
    if (number>end)
       a[10]=number;
    else
      {for (i=0;i<10;i++)
       {if (a[i]>number)
            {temp1=a[i];
       a[i]=number;
       for (j=i+1;j<11;j++)
          {temp2=a[j];
           a[j]=temp1;
           temp1=temp2;
          }
          break;
           }
       }
    }
    printf("Now array a:\n");
    for (i=0;i<11;i++)
       printf("%5d",a[i]);
    printf("\n");
    return 0;
  }
```

6-5

```
#include <stdio.h>
```

```
#define N 5
int main()
{ int a[N],i,temp;
   printf("enter array a:\n");
   for (i=0;i<N;i++)
      scanf("%d",&a[i]);
   printf("array a:\n");
   for (i=0;i<N;i++)
      printf("%4d",a[i]);
   for (i=0;i<N/2;i++)                    //循环的作用是将对称的元素的值互换
      { temp=a[i];
        a[i]=a[N-i-1];
        a[N-i-1]=temp;
       }
   printf("\nNow,array a:\n");
   for (i=0;i<N;i++)
      printf("%4d",a[i]);
   printf("\n");
   return 0;
 }
```

6-6

```
#include <stdio.h>
#define N    10
int main()
{ int i,j,a[N][N];
   for (i=0;i<N;i++)
        {a[i][i]=1;
         a[i][0]=1;
        }
   for (i=2;i<N;i++)
      for (j=1;j<=i-1;j++)
          a[i][j]=a[i-1][j-1]+a[i-1][j];
   for (i=0;i<N;i++)
      {for (j=0;j<=i;j++)
          printf("%6d",a[i][j]);
       printf("\n");
      }
   printf("\n");
   return 0;
}
```

6-7

```
#include <stdio.h>
int main()
{ int a[15][15],i,j,k,p,n;
   p=1;
   while(p==1)
      {printf("enter n(n=1--15):");
       scanf("%d",&n);
       if ((n!=0) && (n<=15) && (n%2!=0))
          p=0;
      }
   for (i=1;i<=n;i++)
       for (j=1;j<=n;j++)
          a[i][j]=0;
   j=n/2+1;
   a[1][j]=1;
   for (k=2;k<=n*n;k++)
      {i=i-1;
       j=j+1;
       if ((i<1) && (j>n))
          {i=i+2;
           j=j-1;
          }
       else
          {if (i<1) i=n;
           if (j>n) j=1;
          }
       if (a[i][j]==0)
          a[i][j]=k;
       else
          {i=i+2;
           j=j-1;
           a[i][j]=k;
          }
      }
   for (i=1;i<=n;i++)
      {for (j=1;j<=n;j++)
          printf("%5d",a[i][j]);
       printf("\n");
      }
   return 0;
 }
```

6-8

```
#include <stdio.h>
#define N 4
#define M 5                              /*  数组为 4 行 5 列  */
int main()
{
   int i,j,k,a[N][M],max,maxj,flag;
   printf("please input matrix:\n");
   for (i=0;i<N;i++)                    /*  输入数组  */
        for (j=0;j<M;j++)
         scanf("%d",&a[i][j]);
   for (i=0;i<N;i++)
    {max=a[i][0];                        /*  开始时假设 a[i][0]最大  */
     maxj=0;                             /*  将列号 0 赋给 maxj 保存  */
     for (j=0;j<M;j++)                  /*  找出第 i 行中的最大数  */
        if (a[i][j]>max)
           {max=a[i][j];                /*  将本行的最大数存放在 max 中  */
            maxj=j;                     /*  将最大数所在的列号存放在 maxj 中  */
           }
     flag=1;                            /*  先假设是鞍点，以 flag 为 1 代表  */
     for (k=0;k<N;k++)
        if (max>a[k][maxj])             /*  将最大数和其同列元素相比  */
            {flag=0;                    /*  如果 max 不是同列最小，表示不是鞍点令 flag1 为 0 */
             continue;}
     if(flag)                           /*  如果 flag1 为 1 表示是鞍点  */
     {printf("a[%d][%d]=%d\n",i,maxj,max);     /*  输出鞍点的值和所在行列号  */
      break;
     }
   }
   if(!flag)                                /*  如果 flag 为 0 表示鞍点不存在  */
      printf("It is not exist!\n");
   return 0;
 }
```

6-9

```
#include <stdio.h>
#define    N 15
int main()
{ int i,number,top,bott,mid,loca,a[N],flag=1,sign;
   char c;
   printf("enter data:\n");
```

```
scanf("%d",&a[0]);
i=1;
while(i<N)
  {scanf("%d",&a[i]);
   if (a[i]>=a[i-1])
      i++;
   else
      printf("enter this data again:\n");
  }
printf("\n");
for (i=0;i<N;i++)
   printf("%5d",a[i]);
printf("\n");
while(flag)
   {printf("input number to look for:");
    scanf("%d",&number);
    sign=0;
    top=0;                    //top 是查找区间的起始位置
    bott=N-1;                 //bott 是查找区间的最末位置
    if ((number<a[0])||(number>a[N-1]))   //要查的数不在查找区间内
       loca=-1;               //  表示找不到
    while ((!sign) && (top<=bott))
       {mid=(bott+top)/2;
        if (number==a[mid])
          {loca=mid;
           printf("Has found %d, its position is %d\n",number,loca+1);
           sign=1;
          }
        else if (number<a[mid])
          bott=mid-1;
        else
        top=mid+1;
       }
    if(!sign||loca==-1)
       printf("cannot find %d.\n",number);;
    printf("continu or not(Y/N)?");
    scanf(" %c",&c);
    if (c=='N'||c=='n')
       flag=0;
   }
return 0;
  }
```

6-10

```
#include <stdio.h>
int main()
  {int i,j,upp,low,dig,spa,oth;
    char text[3][80];
    upp=low=dig=spa=oth=0;
    for (i=0;i<3;i++)
      { printf("please input line %d:\n",i+1);
         gets(text[i]);
         for (j=0;j<80 && text[i][j]!='\0';j++)
           {if (text[i][j]>='A'&& text[i][j]<='Z')
               upp++;
            else if (text[i][j]>='a' && text[i][j]<='z')
               low++;
            else if (text[i][j]>='0' && text[i][j]<='9')
               dig++;
            else if (text[i][j]==' ')
               spa++;
            else
               oth++;
         }
      }
         printf("\nupper case: %d\n",upp);
         printf("lower case: %d\n",low);
         printf("digit          : %d\n",dig);
         printf("space          : %d\n",spa);
         printf("other          : %d\n",oth);
  return 0;
}
```

6-11

```
#include <stdio.h>
int main()
{ char a[5]={'*','*','*','*','*'};
    int i,j,k;
    char space=' ';
    for (i=0;i<5;i++)
      { printf("\n");
         printf("        ");
         for (j=1;j<=i;j++)
            printf("%c",space);
         for (k=0;k<5;k++)
            printf("%c",a[k]);
```

```
    }
  printf("\n");
  return 0;
}
```

6-12a-c

```
#include <stdio.h>
int main()
{ int j,n;
  char ch[80],tran[80];
  printf("input cipher code:");
  gets(ch);
  printf("\ncipher code    :%s",ch);
  j=0;
  while (ch[j]!='\0')
  { if ((ch[j]>='A') && (ch[j]<='Z'))
      tran[j]=155-ch[j];
    else if ((ch[j]>='a') && (ch[j]<='z'))
      tran[j]=219-ch[j];
    else
      tran[j]=ch[j];
    j++;
  }
  n=j;
  printf("\noriginal text:");
  for (j=0;j<n;j++)
    putchar(tran[j]);
  printf("\n");
  return 0;
 }
```

6-12b

```
#include <stdio.h>
int main()
 {int j,n;
  char ch[80];
  printf("input cipher code:\n");
  gets(ch);
  printf("\ncipher code:%s\n",ch);
  j=0;
  while (ch[j]!='\0')
  { if ((ch[j]>='A') && (ch[j]<='Z'))
```

```
            ch[j]=155-ch[j];
        else if ((ch[j]>='a') && (ch[j]<='z'))
            ch[j]=219-ch[j];
        else
            ch[j]=ch[j];
        j++;
    }
    n=j;
    printf("original text:");
    for (j=0;j<n;j++)
        putchar(ch[j]);
    printf("\n");
    return 0;
  }
```

6-13

```
#include <stdio.h>
int main()
{ char s1[80],s2[40];
    int i=0,j=0;
    printf("input string1:");
    scanf("%s",s1);
    printf("input string2:");
    scanf("%s",s2);
    while (s1[i]!='\0')
        i++;
    while(s2[j]!='\0')
        s1[i++]=s2[j++];
    s1[i]='\0';
    printf("\nThe new string is:%s\n",s1);
    return 0;
  }
```

6-14

```
#include <stdio.h>
int main()
{ int i,resu;
    char s1[100],s2[100];
    printf("input string1:");
    gets(s1);
    printf("\ninput string2:");
```

```
    gets(s2);
    i=0;
    while ((s1[i]==s2[i]) && (s1[i]!='\0'))i++;
    if (s1[i]=='\0' && s2[i]=='\0')
          resu=0;
    else
          resu=s1[i]-s2[i];
    printf("\nresult:%d.\n",resu);
    return 0;
  }
```

6-15

```
#include <stdio.h>
#include <string.h>
int main()
{ char s1[80],s2[80];
    int i;
    printf("input s2:");
    scanf("%s",s2);
    for (i=0;i<=strlen(s2);i++)
        s1[i]=s2[i];
    printf("s1:%s\n",s1);
    return 0;
  }
```

第 7 章用函数实现模块化程序设计 【第 218 页】

7-1-1

```
#include <stdio.h>
int main()
  {int hcf(int,int);
    int lcd(int,int,int);
    int u,v,h,l;
    scanf("%d,%d",&u,&v);
    h=hcf(u,v);
    printf("H.C.F=%d\n",h);
    l=lcd(u,v,h);
    printf("L.C.D=%d\n",l);
    return 0;
  }

int hcf(int u,int v)
{int t,r;
```

```
  if (v>u)
     {t=u;u=v;v=t;}
  while ((r=u%v)!=0)
     {u=v;
      v=r;}
  return(v);
}

int lcd(int u,int v,int h)
   {
     return(u*v/h);
   }
```

7-1-2

```
#include <stdio.h>
int Hcf,Lcd;
int main()
  {void hcf(int,int);
   void lcd(int,int);
   int u,v;
   scanf("%d,%d",&u,&v);
   hcf(u,v);
   lcd(u,v);
   printf("H.C.F=%d\n",Hcf);
   printf("L.C.D=%d\n",Lcd);
   return 0;
  }

void hcf(int u,int v)
{int t,r;
  if (v>u)
     {t=u;u=v;v=t;}
  while ((r=u%v)!=0)
     {u=v;
      v=r;
     }
  Hcf=v;
}

void lcd(int u,int v)
   {
     Lcd=u*v/Hcf;
   }
```

7-2

```
#include <stdio.h>
#include <math.h>
float x1,x2,disc,p,q;
int main()
{void greater_than_zero(float,float);
  void equal_to_zero(float,float);
  void smaller_than_zero(float,float);
  float a,b,c;
  printf("input a,b,c:");
  scanf("%f,%f,%f",&a,&b,&c);
  printf("equation: %5.2f*x*x+%5.2f*x+%5.2f=0\n",a,b,c);
  disc=b*b-4*a*c;
  printf("root:\n");
  if (disc>0)
    {
      greater_than_zero(a,b);
      printf("x1=%f\t\tx2=%f\n",x1,x2);
    }
  else if (disc==0)
    {equal_to_zero(a,b);
      printf("x1=%f\t\tx2=%f\n",x1,x2);
    }
  else
    {smaller_than_zero(a,b);
      printf("x1=%f+%fi\tx2=%f-%fi\n",p,q,p,q);
    }
  return 0;
}

void greater_than_zero(float a,float b)
  {x1=(-b+sqrt(disc))/(2*a);
    x2=(-b-sqrt(disc))/(2*a);
  }

void equal_to_zero(float a,float b)
  {
    x1=x2=(-b)/(2*a);
  }

void smaller_than_zero(float a,float b)
  {
```

```
   p=-b/(2*a);
   q=sqrt(-disc)/(2*a);
  }
```

7-3

```
#include <stdio.h>
int main()
  {int prime(int);
   int n;
   printf("input an integer:");
   scanf("%d",&n);
   if (prime(n))
      printf("%d is a prime.\n",n);
   else
      printf("%d is not a prime.\n",n);
   return 0;
  }

  int prime(int n)
   {int flag=1,i;
    for (i=2;i<n/2 && flag==1;i++)
       if (n%i==0)
          flag=0;
    return(flag);
   }
```

7-4

```
#include <stdio.h>
#define N 3
int array[N][N];
int main()
{ void convert(int array[][3]);
int i,j;
  printf("input array:\n");
  for (i=0;i<N;i++)
     for (j=0;j<N;j++)
        scanf("%d",&array[i][j]);
  printf("\noriginal array :\n");
  for (i=0;i<N;i++)
   {for (j=0;j<N;j++)
```

```
        printf("%5d",array[i][j]);
      printf("\n");
    }
convert(array);
printf("convert array:\n");
  for (i=0;i<N;i++)
    {for (j=0;j<N;j++)
         printf("%5d",array[i][j]);
      printf("\n");
    }
  return 0;
  }

void convert(int array[][3])
{int i,j,t;
  for (i=0;i<N;i++)
      for (j=i+1;j<N;j++)
        {t=array[i][j];
         array[i][j]=array[j][i];
         array[j][i]=t;
        }
}


#include <stdio.h>
#include <string.h>
int main()
{void inverse(char str[]);
  char str[100];
  printf("input string:");
  scanf("%s",str);
  inverse(str);
  printf("inverse string:%s\n",str);
  return 0;
}

void inverse(char str[])
  {char t;
    int i,j;
    for (i=0,j=strlen(str);i<(strlen(str)/2);i++,j--)
      {t=str[i];
        str[i]=str[j-1];
        str[j-1]=t;
      }
```

```
  }
```

7-6

```
#include <stdio.h>
int main()
{void concatenate(char string1[],char string2[],char string[]);
  char s1[100],s2[100],s[100];
  printf("input string1:");
  scanf("%s",s1);
  printf("input string2:");
  scanf("%s",s2);
  concatenate(s1,s2,s);
  printf("\nThe new string is %s\n",s);
  return 0;
  }

void concatenate(char string1[],char string2[],char string[])
{int i,j;
  for (i=0;string1[i]!='\0';i++)
     string[i]=string1[i];
  for(j=0;string2[j]!='\0';j++)
     string[i+j]=string2[j];
  string[i+j]='\0';
}
```

7-7

```
#include <stdio.h>
int main()
{void cpy(char [],char []);
  char str[80],c[80];
  printf("input string:");
  gets(str);
  cpy(str,c);
  printf("The vowel letters are:%s\n",c);
  return 0;
  }

  void cpy(char s[],char c[])
  { int i,j;
     for (i=0,j=0;s[i]!='\0';i++)
```

```
        if (s[i]=='a'||s[i]=='A'||s[i]=='e'||s[i]=='E'||s[i]=='i'||
        s[i]=='I'||s[i]=='o'||s[i]=='O'||s[i]=='u'||s[i]=='U')
           {c[j]=s[i];
            j++;
           }
        c[j]='\0';
  }
```

7-8

```
#include <stdio.h>
#include <string.h>
int main()
{char str[80];
  void insert(char []);
  printf("input four digits:");
  scanf("%s",str);
  insert(str);
  return 0;
}

void insert(char str[])
{int i;
  for (i=strlen(str);i>0;i--)
    {str[2*i]=str[i];
     str[2*i-1]=' ';
    }
  printf("output:\n%s\n",str);
}
```

7-9

```
#include <stdio.h>
int letter,digit,space,others;
int main()
{void count(char []);
  char text[80];
  printf("input string:\n");
  gets(text);
  printf("string:");
  puts(text);
  letter=0;
  digit=0;
  space=0;
  others=0;
```

```
  count(text);
  printf("\nletter:%d\ndigit:%d\nspace:%d\nothers:%d\n",letter,digit,space,others);
  return 0;
}

  void count(char str[])
{int i;
  for (i=0;str[i]!='\0';i++)
  if ((str[i]>='a'&& str[i]<='z')||(str[i]>='A' && str[i]<='Z'))
       letter++;
  else if (str[i]>='0' && str [i]<='9')
       digit++;
  else if (str[i]==32)
       space++;
  else
       others++;
}
```

7-10

```
#include <stdio.h>
#include <string.h>
int main()
{int alphabetic(char);
  int longest(char []);
  int i;
  char line[100];
  printf("input one line:\n");
  gets(line);
  printf("The longest word is :");
  for (i=longest(line);alphabetic(line[i]);i++)
     printf("%c",line[i]);
  printf("\n");
  return 0;
}

int alphabetic(char c)
{if ((c>='a' && c<='z')||(c>='A'&&c<='z'))
    return(1);
  else
    return(0);
}

int longest(char string[])
{int len=0,i,length=0,flag=1,place=0,point;
```

```
  for (i=0;i<=strlen(string);i++)
     if (alphabetic(string[i]))
        if (flag)
          {point=i;
           flag=0;
          }
        else
           len++;
     else
        {flag=1;
         if (len>=length)
       {length=len;
        place=point;
        len=0;
       }
        }
  return(place);
}
```

7-11

```
#include <stdio.h>
#include <string.h>
#define N 10
char str[N];
int main()
{void sort(char []);
  int i,flag;
  for (flag=1;flag==1;)
    {printf("input string:\n");
     scanf("%s",&str);
     if (strlen(str)>N)
        printf("string too long,input again!");
     else
        flag=0;
    }
  sort(str);
  printf("string sorted:\n");
  for (i=0;i<N;i++)
    printf("%c",str[i]);
  printf("\n");
  return 0;
}
```

```
void sort(char str[])
{int i,j;
  char t;
  for(j=1;j<N;j++)
     for (i=0;(i<N-j)&&(str[i]!='\0');i++)
        if(str[i]>str[i+1])
           {t=str[i];
            str[i]=str[i+1];
            str[i+1]=t;
           }
}
```

7-12

```
#include <stdio.h>
#include <math.h>
int main()
{float solut(float a,float b,float c,float d);
  float a,b,c,d;
  printf("input a,b,c,d:");
  scanf("%f,%f,%f,%f",&a,&b,&c,&d);
  printf("x=%10.7f\n",solut(a,b,c,d));
  return 0;
}

float solut(float a,float b,float c,float d)
{float x=1,x0,f,f1;
  do
     {x0=x;
      f=((a*x0+b)*x0+c)*x0+d;
      f1=(3*a*x0+2*b)*x0+c;
      x=x0-f/f1;
     }
  while(fabs(x-x0)>=1e-3);
  return(x);
}
```

7-13

```
#include <stdio.h>
#define N 10
#define M 5
float score[N][M];
float a_stu[N],a_cour[M];
```

```
int r,c;

int main()
{ int i,j;
   float h;
   float s_var(void);
   float highest();
   void input_stu(void);
   void aver_stu(void);
   void aver_cour(void);
   input_stu();
   aver_stu();
   aver_cour();
   printf("\n   NO.      cour1    cour2    cour3    cour4    cour5    aver\n");
   for(i=0;i<N;i++)
     {printf("\n NO %2d ",i+1);
      for(j=0;j<M;j++)
         printf("%8.2f",score[i][j]);
      printf("%8.2f\n",a_stu[i]);
     }
   printf("\naverage:");
   for (j=0;j<M;j++)
      printf("%8.2f",a_cour[j]);
   printf("\n");
   h=highest();
   printf("highest:%7.2f    NO. %2d    course %2d\n",h,r,c);
   printf("variance %8.2f\n",s_var());
   return 0;
}

void input_stu(void)
  {int i,j;
   for (i=0;i<N;i++)
     {printf("\ninput score of student%2d:\n",i+1);
      for (j=0;j<M;j++)
         scanf("%f",&score[i][j]);
     }
  }


void aver_stu(void)
  {int i,j;
   float s;
   for (i=0;i<N;i++)
```

```
    {for (j=0,s=0;j<M;j++)
        s+=score[i][j];
      a_stu[i]=s/5.0;
    }
 }

void aver_cour(void)
 {int i,j;
  float s;
  for (j=0;j<M;j++)
     {s=0;
      for (i=0;i<N;i++)
         s+=score[i][j];
      a_cour[j]=s/(float)N;
     }
 }

float highest()
 {float high;
  int i,j;
  high=score[0][0];
  for (i=0;i<N;i++)
     for (j=0;j<M;j++)
        if (score[i][j]>high)
     {high=score[i][j];
      r=i+1;
      c=j+1;
     }
  return(high);
 }

float s_var(void)
 {int i;
  float sumx,sumxn;
  sumx=0.0;
  sumxn=0.0;
  for (i=0;i<N;i++)
     {sumx+=a_stu[i]*a_stu[i];
      sumxn+=a_stu[i];
     }
  return(sumx/N-(sumxn/N)*(sumxn/N));
 }
```

7-14

```
#include <stdio.h>
#define N 10
#define M 5
float score[N][M];
float a_stu[N],a_cour[M];
int r,c;

int main()
{ int i,j;
   float h;
   float s_var(void);
   float highest();
   void input_stu(void);
   void aver_stu(void);
   void aver_cour(void);
   input_stu();
   aver_stu();
   aver_cour();
   printf("\n NO.      cour1    cour2    cour3    cour4    cour5    aver\n");
   for(i=0;i<N;i++)
     {printf("\n NO %2d ",i+1);
      for(j=0;j<M;j++)
         printf("%8.2f",score[i][j]);
      printf("%8.2f\n",a_stu[i]);
     }
   printf("\naverage:");
   for (j=0;j<M;j++)
      printf("%8.2f",a_cour[j]);
   printf("\n");
   h=highest();
   printf("highest:%7.2f    NO. %2d    course %2d\n",h,r,c);
   printf("variance %8.2f\n",s_var());
   return 0;
}

void input_stu(void)
  {int i,j;
   for (i=0;i<N;i++)
     {printf("\ninput score of student%2d:\n",i+1);
      for (j=0;j<M;j++)
         scanf("%f",&score[i][j]);
     }
```

```
}


void aver_stu(void)
 {int i,j;
  float s;
  for (i=0;i<N;i++)
   {for (j=0,s=0;j<M;j++)
       s+=score[i][j];
    a_stu[i]=s/5.0;
   }
 }

void aver_cour(void)
 {int i,j;
  float s;
  for (j=0;j<M;j++)
     {s=0;
      for (i=0;i<N;i++)
         s+=score[i][j];
      a_cour[j]=s/(float)N;
     }
 }

float highest()
 {float high;
  int i,j;
  high=score[0][0];
  for (i=0;i<N;i++)
     for (j=0;j<M;j++)
        if (score[i][j]>high)
     {high=score[i][j];
      r=i+1;
      c=j+1;
     }
  return(high);
 }

float s_var(void)
 {int i;
  float sumx,sumxn;
  sumx=0.0;
  sumxn=0.0;
  for (i=0;i<N;i++)
```

```
        {sumx+=a_stu[i]*a_stu[i];
         sumxn+=a_stu[i];
        }
    return(sumx/N-(sumxn/N)*(sumxn/N));
  }
```

7-15

```
#include <stdio.h>
#include <string.h>
#define N 10
int main()
      {void input(int [],char name[][8]);
       void sort(int [],char name[][8]);
       void search(int ,int [],char name[][8]);
       int num[N],number,flag=1,c;
       char name[N][8];
       input(num,name);
       sort(num,name);
       while (flag==1)
            {printf("\ninput number to look for:");
             scanf("%d",&number);
             search(number,num,name);
             printf("continue ot not(Y/N)?");
             getchar();
             c=getchar();
             if (c=='N'||c=='n')
                 flag=0;
            }
       return 0;
      }

void input(int num[],char name[N][8])
  {int i;
    for (i=0;i<N;i++)
      {printf("input NO.: ");
       scanf("%d",&num[i]);
       printf("input name: ");
       getchar();
       gets(name[i]);
      }
  }

void sort(int num[],char name[N][8])
  { int i,j,min,templ;
```

```
    char temp2[8];
    for (i=0;i<N-1;i++)
      {min=i;
       for (j=i;j<N;j++)
          if (num[min]>num[j])    min=j;
       templ=num[i];
       strcpy(temp2,name[i]);
       num[i]=num[min];
       strcpy (name[i],name[min]);
       num[min]=templ;
       strcpy(name[min],temp2);
       }
      printf("\n result:\n");
      for (i=0;i<N;i++)
          printf("\n %5d%10s",num[i],name[i]);
 }

void search(int n,int num[],char name[N][8])
   {int top,bott,mid,loca,sign;
    top=0;
    bott=N-1;
    loca=0;
    sign=1;
    if ((n<num[0])||(n>num[N-1]))
       loca=-1;
    while((sign==1) && (top<=bott))
      {mid=(bott+top)/2;
       if (n==num[mid])
          {loca=mid;
           printf("NO. %d , his name is %s.\n",n,name[loca]);
           sign=-1;
          }
       else if (n<num[mid])
           bott=mid-1;
       else
           top=mid+1;
      }
    if (sign==1 || loca==-1)
          printf("%d not been found.\n",n);
   }
```

7-16

```
#include <stdio.h>
#define MAX 1000
int main()
{ int htoi(char s[]);
   int c,i,flag,flag1;
   char t[MAX];
   i=0;
   flag=0;
   flag1=1;
   printf("input a HEX number:");
   while((c=getchar())!='\0' && i<MAX&& flag1)
    {if (c>='0' && c<='9'||c>='a' && c<='f'||c>='A' && c<='F')
       {flag=1;
        t[i++]=c;
       }
      else if (flag)
       {t[i]='\0';
        printf("decimal    number %d\n",htoi(t));
        printf("continue or not?");
        c=getchar();
        if (c=='N'||c=='n')
           flag1=0;
        else
           {flag=0;
            i=0;
            printf("\ninput a HEX number:");
           }
       }
    }
   return 0;
   }

   int htoi(char s[])
   { int i,n;
      n=0;
      for (i=0;s[i]!='\0';i++)
       {if (s[i]>='0'&& s[i]<='9')
           n=n*16+s[i]-'0';
        if (s[i]>='a' && s[i]<='f')
           n=n*16+s[i]-'a'+10;
        if (s[i]>='A' && s[i]<='F')
           n=n*16+s[i]-'A'+10;
       }
      return(n);
```

```
    }
```

7-17

```
#include <stdio.h>
int main()
{ void convert(int n);
    int number;
    printf("input an integer: ");
    scanf("%d",&number);
    printf("output: ");
    if (number<0)
       {putchar('-');putchar(' ');     /* 先输出一个‘-’号和空格 */
         number=-number;
       }
    convert(number);
    printf("\n");
    return 0;
}

void convert(int n)
{ int i;
    if ((i=n/10)!=0)
       convert(i);
    putchar(n%10+'0');
    putchar(32);
}
```

7-18

```
#include <stdio.h>
int main()
{int sum_day(int month,int day);
  int leap(int year);
  int year,month,day,days;
  printf("input date(year,month,day):");
  scanf("%d,%d,%d",&year,&month,&day);
  printf("%d/%d/%d ",year,month,day);
  days=sum_day(month,day);                    /* 调用函数 sum_day */
  if(leap(year)&&month>=3)                    /* 调用函数 leap */
     days=days+1;
  printf("is the %dth day in this year.\n",days);
  return 0;
}
```

```
int sum_day(int month,int day)                /*  函数 sum_day:计算日期  */
   {int day_tab[13]={0,31,28,31,30,31,30,31,31,30,31,30,31};
    int i;
    for (i=1;i<month;i++)
        day+=day_tab[i];          /*  累加所在月之前天数  */
    return(day);
   }                                          /*  函数 leap:判断是否为闰年  */

int leap(int year)
  {int leap;
   leap=year%4==0&&year%100!=0||year%400==0;
   return(leap);
  }
```

第 8 章善于利用指针 【第 291 页】

8-1

```
#include <stdio.h>
int main()
{ void swap(int *p1,int *p2);
  int n1,n2,n3;
  int *p1,*p2,*p3;
  printf("input three integer n1,n2,n3:");
  scanf("%d,%d,%d",&n1,&n2,&n3);
  p1=&n1;
  p2=&n2;
  p3=&n3;
  if(n1>n2) swap(p1,p2);
  if(n1>n3) swap(p1,p3);
  if(n2>n3) swap(p2,p3);
  printf("Now,the order is:%d,%d,%d\n",n1,n2,n3);
  return 0;
  }

  void swap(int *p1,int *p2)
    {int p;
     p=*p1; *p1=*p2; *p2=p;
    }


#include <stdio.h>
#include <string.h>
int main()
```

```
{void swap(char *,char *);
  char str1[20],str2[20],str3[20];
  printf("input three line:\n");
  gets(str1);
  gets(str2);
  gets(str3);
  if(strcmp(str1,str2)>0)    swap(str1,str2);
  if(strcmp(str1,str3)>0)    swap(str1,str3);
  if(strcmp(str2,str3)>0)    swap(str2,str3);
  printf("Now,the order is:\n");
  printf("%s\n%s\n%s\n",str1,str2,str3);
  return 0;
  }

  void swap(char *p1,char *p2)
  {char p[20];
   strcpy(p,p1);strcpy(p1,p2);strcpy(p2,p);
  }
```

8-3

```
#include <stdio.h>
int main()
  { void input(int *);
     void max_min_value(int *);
     void output(int *);
     int number[10];
     input(number);
     max_min_value(number);
     output(number);
     return 0;
  }

  void input(int *number)
  {int i;
   printf("input 10 numbers:");
   for (i=0;i<10;i++)
      scanf("%d",&number[i]);
   }

  void max_min_value(int *number)
  { int *max,*min,*p,temp;
     max=min=number;
     for (p=number+1;p<number+10;p++)
```

```
        if (*p>*max) max=p;
        else if (*p<*min) min=p;
     temp=number[0];number[0]=*min;*min=temp;
     if(max==number) max=min;
     temp=number[9];number[9]=*max;*max=temp;
   }

void output(int *number)
   {int *p;
     printf("Now,they are:        ");
     for (p=number;p<number+10;p++)
          printf("%d ",*p);
     printf("\n");
     }
```

8-4

```
#include <stdio.h>
int main()
{void move(int [20],int,int);
  int number[20],n,m,i;
  printf("how many numbers?");
  scanf("%d",&n);
  printf("input %d numbers:\n",n);
  for (i=0;i<n;i++)
     scanf("%d",&number[i]);
  printf("how many place you want move?");
  scanf("%d",&m);
  move(number,n,m);
  printf("Now,they are:\n");
  for (i=0;i<n;i++)
     printf("%d    ",number[i]);
  printf("\n");
  return 0;
}

void move(int array[20],int n,int m)
  {int *p,array_end;
    array_end=*(array+n-1);
    for (p=array+n-1;p>array;p--)
       *p=*(p-1);
    *array=array_end;
```

```
   m--;
   if (m>0) move(array,n,m);
  }
```

8-5

```
#include <stdio.h>
int main()
{int i,k,m,n,num[50],*p;
  printf("\ninput number of person: n=");
  scanf("%d",&n);
  p=num;
  for (i=0;i<n;i++)
     *(p+i)=i+1;
  i=0;
  k=0;
  m=0;
  while (m<n-1)
    {if (*(p+i)!=0)    k++;
     if (k==3)
         {*(p+i)=0;
           k=0;
           m++;
         }
     i++;
     if (i==n) i=0;
     }
  while(*p==0) p++;
  printf("The last one is NO.%d\n",*p);
  return 0;
}
```

8-6

```
#include <stdio.h>
int main()
{int length(char *p);
int len;
char str[20];
printf("input string:    ");
scanf("%s",str);
len=length(str);
printf("The length of string is %d.\n",len);
return 0;
```

```
}

int length(char *p)
{int n;
  n=0;
  while (*p!='\0')
    {n++;
      p++;
    }
  return(n);
}
```

8-7

```
#include <stdio.h>
#include <string.h>
int main()
{void copystr(char *,char *,int);
  int m;
  char str1[20],str2[20];
  printf("input string:");
  gets(str1);
  printf("which character that begin to copy?");
  scanf("%d",&m);
  if (strlen(str1)<m)
     printf("input error!");
  else
     {copystr(str1,str2,m);
       printf("result:%s\n",str2);
     }
  return 0;
}

void copystr(char *p1,char *p2,int m)
{int n;
  n=0;
  while (n<m-1)
    {n++;
      p1++;
    }
  while (*p1!='\0')
     {*p2=*p1;
       p1++;
       p2++;
```

```
    }
  *p2='\0';
}
```

8-8

```
#include <stdio.h>
int main()
{int upper=0,lower=0,digit=0,space=0,other=0,i=0;
char *p,s[20];
printf("input string:    ");
while ((s[i]=getchar())!='\n') i++;
p=&s[0];
while (*p!='\n')
   {if (('A'<=*p) && (*p<='Z'))
       ++upper;
     else if (('a'<=*p) && (*p<='z'))
       ++lower;
     else if (*p==' ')
       ++space;
     else if ((*p<='9') && (*p>='0'))
       ++digit;
     else
       ++other;
     p++;
   }
printf("upper case:%d          lower case:%d",upper,lower);
printf("          space:%d          digit:%d          other:%d\n",space,digit,other);
return 0;
}
```

8-9

```
#include <stdio.h>
int main()
{void move(int *pointer);
  int a[3][3],*p,i;
  printf("input matrix:\n");
  for (i=0;i<3;i++)
     scanf("%d %d %d",&a[i][0],&a[i][1],&a[i][2]);
  p=&a[0][0];
  move(p);
  printf("Now,matrix:\n");
  for (i=0;i<3;i++)
     printf("%d %d %d\n",a[i][0],a[i][1],a[i][2]);
```

```
  return 0;
   }

  void move(int *pointer)
   {int i,j,t;
    for (i=0;i<3;i++)
        for (j=i;j<3;j++)
           {t=*(pointer+3*i+j);
            *(pointer+3*i+j)=*(pointer+3*j+i);
            *(pointer+3*j+i)=t;
           }
   }
```

8-10-1

```
#include <stdio.h>
int main()
{void change(int *p);
  int a[5][5],*p,i,j;
  printf("input matrix:\n");
  for (i=0;i<5;i++)
     for (j=0;j<5;j++)
        scanf("%d",&a[i][j]);
  p=&a[0][0];
  change(p);
  printf("Now,matrix:\n");
  for (i=0;i<5;i++)
    {for (j=0;j<5;j++)
        printf("%d ",a[i][j]);
     printf("\n");
    }
  return 0;
}

void change(int *p)
  {int i,j,temp;
    int *pmax,*pmin;
    pmax=p;
    pmin=p;
    for (i=0;i<5;i++)
       for (j=i;j<5;j++)
         {if (*pmax<*(p+5*i+j)) pmax=p+5*i+j;
          if (*pmin>*(p+5*i+j)) pmin=p+5*i+j;
         }
```

```
   temp=*(p+12);
   *(p+12)=*pmax;
   *pmax=temp;
   temp=*p;
   *p=*pmin;
   *pmin=temp;
   pmin=p+1;
   for (i=0;i<5;i++)
        for (j=0;j<5;j++)
           if (((p+5*i+j)!=p) && (*pmin>*(p+5*i+j))) pmin=p+5*i+j;
   temp=*pmin;
   *pmin=*(p+4);
   *(p+4)=temp;
   pmin=p+1;
   for (i=0;i<5;i++)
      for (j=0;j<5;j++)
         if (((p+5*i+j)!=(p+4))&&((p+5*i+j)!=p)&&(*pmin>*(p+5*i+j)))pmin=p+5*i+j;
   temp=*pmin;
   *pmin=*(p+20);
   *(p+20)=temp;
   pmin=p+1;
   for (i=0;i<5;i++)
        for (j=0;j<5;j++)
           if (((p+5*i+j)!=p) && ((p+5*i+j)!=(p+4)) && ((p+5*i+j)!=(p+20)) &&
(*pmin>*(p+5*i+j)))
        pmin=p+5*i+j;
   temp=*pmin;
   *pmin=*(p+24);
   *(p+24)=temp;
  }
```

8-10-2

```
#include <stdio.h>
int main()
{void change(int *p);
  int a[5][5],*p,i,j;
  printf("input matrix:\n");
  for (i=0;i<5;i++)
     for (j=0;j<5;j++)
        scanf("%d",&a[i][j]);
  p=&a[0][0];
  change(p);
  printf("Now,matrix:\n");
```

```
  for (i=0;i<5;i++)
    {for (j=0;j<5;j++)
        printf("%d ",a[i][j]);
     printf("\n");
    }
  return 0;
}

void change(int *p)                  //交换函数
  {int i,j,temp;
    int *pmax,*pmin;
    pmax=p;
    pmin=p;
    for (i=0;i<5;i++)                //找最大值和最小值的地址,并赋给  pmax,pmin
       for (j=i;j<5;j++)
         {if (*pmax<*(p+5*i+j)) pmax=p+5*i+j;
          if (*pmin>*(p+5*i+j)) pmin=p+5*i+j;
         }
    temp=*(p+12);                        //将最大值与中心元素互换
    *(p+12)=*pmax;
    *pmax=temp;

    temp=*p;                             //将最小值与左上角元素互换
    *p=*pmin;
    *pmin=temp;

    pmin=p+1;
                                         //将 a[0][1]的地址赋给 pmin，从该位置开始找最小的元素
    for (i=0;i<5;i++)                    //找第二最小值的地址赋给  pmin
       for (j=0;j<5;j++)
       {if(i==0 && j==0) continue;
        if    (*pmin > *(p+5*i+j)) pmin=p+5*i+j;
       }
    temp=*pmin;                          //将第二最小值与右上角元素互换
    *pmin=*(p+4);
    *(p+4)=temp;

    pmin=p+1;
    for (i=0;i<5;i++)                    //找第三最小值的地址赋给 pmin
       for (j=0;j<5;j++)
       {if((i==0    && j==0) ||(i==0    && j==4)) continue;
        if(*pmin>*(p+5*i+j)) pmin=p+5*i+j;
       }
    temp=*pmin;                          //  将第三最小值与左下角元素互换
```

```
   *pmin=*(p+20);
   *(p+20)=temp;

   pmin=p+1;
   for (i=0;i<5;i++)            //  找第四最小值的地址赋给 pmin
        for (j=0;j<5;j++)
        {if ((i==0    && j==0) ||(i==0    && j==4)||(i==4    && j==0)) continue;
         if (*pmin>*(p+5*i+j)) pmin=p+5*i+j;
        }
   temp=*pmin;                        //将第四最小值与右下角元素互换
   *pmin=*(p+24);
   *(p+24)=temp;
 }
```

8-11-1

```
#include <stdio.h>
#include <string.h>
int main()
{void sort(char s[][6]);
 int i;
 char str[10][6];
 printf("input 10 strings:\n");
 for (i=0;i<10;i++)
    scanf("%s",str[i]);
 sort(str);
 printf("Now,the sequence is:\n");
 for (i=0;i<10;i++)
    printf("%s\n",str[i]);
 return 0;
}

void sort(char s[10][6])
{int i,j;
 char *p,temp[10];
 p=temp;
 for (i=0;i<9;i++)
    for (j=0;j<9-i;j++)
       if (strcmp(s[j],s[j+1])>0)
         {strcpy(p,s[j]);
          strcpy(s[j],s[+j+1]);
          strcpy(s[j+1],p);
         }
}
```

8-11-2

```
#include <stdio.h>
#include <string.h>
int main()
{void sort(char (*p)[6]);
  int i;
  char str[10][6];
  char (*p)[6];
  printf("input 10 strings:\n");
  for (i=0;i<10;i++)
     scanf("%s",str[i]);
  p=str;
  sort(p);
  printf("Now,the sequence is:\n");
  for (i=0;i<10;i++)
     printf("%s\n",str[i]);
  return 0;
  }

void sort(char (*s)[6])
{int i,j;
  char temp[6],*t=temp;
  for (i=0;i<9;i++)
     for (j=0;j<9-i;j++)
        if (strcmp(s[j],s[j+1])>0)
          {strcpy(t,s[j]);
           strcpy(s[j],s[+j+1]);
           strcpy(s[j+1],t);
          }
}
```

8-12

```
#include <stdio.h>
#include <string.h>
int main()
{void sort(char *[]);
  int i;
  char *p[10],str[10][20];
  for (i=0;i<10;i++)
     p[i]=str[i];
  printf("input 10 strings:\n");
  for (i=0;i<10;i++)
```

```
    scanf("%s",p[i]);
  sort(p);
  printf("Now,the sequence is:\n");
  for (i=0;i<10;i++)
     printf("%s\n",p[i]);
  return 0;
  }

void sort(char *s[])
{int i,j;
  char *temp;
  for (i=0;i<9;i++)
     for (j=0;j<9-i;j++)
        if (strcmp(*(s+j),*(s+j+1))>0)
           {temp=*(s+j);
            *(s+j)=*(s+j+1);
            *(s+j+1)=temp;
           }
}
```

8-13

```
#include<stdio.h>
#include<math.h>
int main()
{float integral(float(*)(float),float,float,int);//对 integarl 函数的声明
float fsin(float);                  //对 fsin 函数的声明
float fcos(float);                  //对 fcos 函数的声明
float fexp(float);                  //对 fexp 函数的声明
float a1,b1,a2,b2,a3,b3,c,(*p)(float);
int n=20;
printf("input a1,b1:");
scanf("%f,%f",&a1,&b1);
printf("input a2,b2:");
scanf("%f,%f",&a2,&b2);
printf("input a3,b3:");
scanf("%f,%f",&a3,&b3);
p=fsin;
c=integral(p,a1,b1,n);
printf("The integral of sin(x) is:%f\n",c);
p=fcos;
c=integral(p,a2,b2,n);
printf("The integral of cos(x) is:%f\n",c);
p=fexp;
```

```
c=integral(p,a3,b3,n);
printf("The integral of exp(x) is:%f\n",c);
return 0;
}

float integral(float(*p)(float),float a,float b,int n)
{int i;
  float x,h,s;
  h=(b-a)/n;
  x=a;
  s=0;
  for(i=1;i<=n;i++)
    {x=x+h;
      s=s+(*p)(x)*h;
    }
    return(s);
}
    float fsin(float x)
        {return sin(x);}
    float fcos(float x)
        {return cos(x);}
    float fexp(float x)
        {return exp(x);}
```

8-14

```
#include <stdio.h>
int main()
{void sort (char *p,int m);
  int i,n;
  char *p,num[20];
  printf("input n:");
  scanf("%d",&n);
  printf("please input these numbers:\n");
  for (i=0;i<n;i++)
      scanf("%d",&num[i]);
  p=&num[0];
  sort(p,n);
  printf("Now,the sequence is:\n");
  for (i=0;i<n;i++)
    printf("%d ",num[i]);
printf("\n");
```

```
return 0;
}

void sort (char *p,int m) //  将 n 个数逆序排列函数
{int i;
  char temp, *p1,*p2;
  for (i=0;i<m/2;i++)
    {p1=p+i;
      p2=p+(m-1-i);
      temp=*p1;
      *p1=*p2;
      *p2=temp;
    }
  }

8-15
#include <stdio.h>
int main()
{void avsco(float *,float *);
  void avcour1(char (*)[10],float *);
  void fali2(char course[5][10],int num[],float *pscore,float aver[4]);
  void good(char course[5][10],int num[4],float *pscore,float aver[4]);
  int i,j,*pnum,num[4];
  float score[4][5],aver[4],*pscore,*paver;
  char course[5][10],(*pcourse)[10];
  printf("input course:\n");
  pcourse=course;
  for (i=0;i<5;i++)
     scanf("%s",course[i]);
  printf("input NO. and scores:\n");
  printf("NO.");
  for (i=0;i<5;i++)
     printf(",%s",course[i]);
  printf("\n");
  pscore=&score[0][0];
  pnum=&num[0];
  for (i=0;i<4;i++)
  {scanf("%d",pnum+i);
   for (j=0;j<5;j++)
      scanf("%f",pscore+5*i+j);
  }
  paver=&aver[0];
  printf("\n\n");
  avsco(pscore,paver);                          //  求出每个学生的平均成绩
```

```
  avcour1(pcourse,pscore);                    // 求出第一门课的平均成绩
  printf("\n\n");
  fali2(pcourse,pnum,pscore,paver);          // 找出 2 门课不及格的学生
  printf("\n\n");
  good(pcourse,pnum,pscore,paver);           // 找出成绩好的学生
  return 0;
}

void avsco(float *pscore,float *paver)   // 求每个学生的平均成绩的函数
  {int i,j;
   float sum,average;
   for (i=0;i<4;i++)
    {sum=0.0;
     for (j=0;j<5;j++)
        sum=sum+(*(pscore+5*i+j));          //累计每个学生的各科成绩
     average=sum/5;                         //计算平均成绩
     *(paver+i)=average;
    }
}

void avcour1(char (*pcourse)[10],float *pscore)        // 求第一课程的平均成绩的函数
  {int i;
   float sum,average1;
   sum=0.0;
   for (i=0;i<4;i++)
      sum=sum+(*(pscore+5*i));                  //累计每个学生的得分
   average1=sum/4;                              //计算平均成绩
   printf("course 1:%s average score:%7.2f\n",*pcourse,average1);
}

void fali2(char course[5][10],int num[],float *pscore,float aver[4])
                // 找两门以上课程不及格的学生的函数
  {int i,j,k,labe1;
   printf("            ==========Student who is fail in two courses=======    \n");
   printf("NO. ");
   for (i=0;i<5;i++)
      printf("%11s",course[i]);
   printf("       average\n");
   for (i=0;i<4;i++)
   {labe1=0;
    for (j=0;j<5;j++)
       if (*(pscore+5*i+j)<60.0) labe1++;
    if (labe1>=2)
      {printf("%d",num[i]);
```

```
        for (k=0;k<5;k++)
           printf("%11.2f",*(pscore+5*i+k));
        printf("%11.2f\n",aver[i]);
      }
   }
}

void good(char course[5][10],int num[4],float *pscore,float aver[4])
    // 找成绩优秀学生(各门 85 以上或平均 90 分以上)的函数
 {int i,j,k,n;
   printf("              ======Students whose score is good======\n");
   printf("NO. ");
   for (i=0;i<5;i++)
      printf("%11s",course[i]);
   printf("       average\n");
   for (i=0;i<4;i++)
     {n=0;
      for (j=0;j<5;j++)
         if (*(pscore+5*i+j)>85.0) n++;
      if ((n==5)||(aver[i]>=90))
        {printf("%d",num[i]);
         for (k=0;k<5;k++)
            printf("%11.2f",*(pscore+5*i+k));
         printf("%11.2f\n",aver[i]);
        }
 }
}



8-16
#include <stdio.h>
int main()
{
 char str[50],*pstr;
 int i,j,k,m,e10,digit,ndigit,a[10],*pa;
 printf("input a string:\n");
 gets(str);
 pstr=&str[0];       /*字符指针 pstr 置于数组 str  首地址*/
 pa=&a[0];             /*指针 pa 置于 a 数组首地址*/
 ndigit=0;            /*ndigit 代表有多少个整数*/
 i=0;                  /*代表字符串中的第几个字符*/
 j=0;
 while(*(pstr+i)!='\0')
      {if((*(pstr+i)>='0') && (*(pstr+i)<='9'))
```

```
            j++;
        else
            {if (j>0)
             {digit=*(pstr+i-1)-48;              /*将个数位赋予 digit*/
              k=1;
              while (k<j)        /*将含有两位以上数的其它位的数值累计于 digit*/
                 {e10=1;
                  for (m=1;m<=k;m++)
                  e10=e10*10;                          /*e10 代表该位数所应乘的因子*/
                  digit=digit+(*(pstr+i-1-k)-48)*e10;   /*将该位数的数值\累加于 digit*/
                  k++;                                 /*位数 K 自增*/
                 }
              *pa=digit;                        /*将数值赋予数组 a*/
              ndigit++;
              pa++;                               /*指针 pa 指向 a 数组下一元素*/
              j=0;
             }
            }
        i++;
       }
  if (j>0)                                    /*以数字结尾字符串的最后一个数据*/
    {digit=*(pstr+i-1)-48;                 /*将个数位赋予 digit*/
     k=1;
     while (k<j)                 /*  将含有两位以上数的其它位的数值累加于 digit*/
       {e10=1;
        for (m=1;m<=k;m++)
           e10=e10*10;                    /*e10 代表位数所应乘的因子*/
        digit=digit+(*(pstr+i-1-k)-48)*e10;   /*将该位数的数值累加于 digit*/
        k++;   /*位数 K 自增*/
       }
     *pa=digit;                            /*将数值赋予数组 a*/
     ndigit++;
     j=0;
    }
    printf("There are %d numbers in this line, they are:\n",ndigit);
    j=0;
    pa=&a[0];
    for (j=0;j<ndigit;j++)                    /*打印数据*/
       printf("%d ",*(pa+j));
    printf("\n");
    return 0;
}
```

8-17

```
#include<stdio.h>
int main()
{int strcmp(char *p1,char *p2);
  int m;
  char str1[20],str2[20],*p1,*p2;
  printf("input two strings:\n");
  scanf("%s",str1);
  scanf("%s",str2);
  p1=&str1[0];
  p2=&str2[0];
  m=strcmp(p1,p2);
  printf("result:%d,\n",m);
  return 0;
}

int strcmp(char *p1,char *p2)                    //两个字符串比较函数
{int i;
  i=0;
  while(*(p1+i)==*(p2+i))
     if (*(p1+i++)=='\0') return(0);          //相等时返回结果 0
  return(*(p1+i)-*(p2+i));                       //不等时返回结果为第一个不等字符 ASCII 码的差值
}
```

8-18

```
#include <stdio.h>
int main()
{char *month_name[13]={"illegal month","January","February","March","April",
     "May","June","july","August","September","October", "November","December"};
int n;
printf("input month:\n");
scanf("%d",&n);
if ((n<=12) && (n>=1))
     printf("It is %s.\n",*(month_name+n));
else
   printf("It is wrong.\n");
return 0;
}
```

8-19-1

```
#include <stdio.h>
#define NEWSIZE 1000                              //指定开辟存区的最大容量
```

```
char newbuf[NEWSIZE];                          //定义字符数组 newbuf
char *newp=newbuf;                             //定义指针变量 newp，指向可存区的始端

char  *new(int  n)                             //定义开辟存区的函数 new,开辟存储区后返回指
针
    {if (newp+n<=newbuf+NEWSIZE)               //  开辟区未超过 newbuf 数组的大小
       {newp+=n;                               // newp 指向存储区的末尾
        return(newp-n);                        //  返回一个指针,它指向存区的开始位置
       }
      else
        return(NULL);                          //  当存区不够分配时,返回一个空指针
    }


8-19-2
#include <stdio.h>
#define NEWSIZE 1000
char newbuf[NEWSIZE];
char *newp=newbuf;

void free(char *p)                             //释放存区函数
   {if (p>=newbuf && p< newbuf + NEWSIZE)
       newp=p;
   }


8-20
#define LINEMAX 20                    /*定义字符串的最大长度*/
int main()
{int i;
 char **p,*pstr[5],str[5][LINEMAX];
 for (i=0;i<5;i++)
    pstr[i]=str[i];     /*将第 i 个字符串的首地址赋予指针数组  pstr  的第 i 个元素*/
 printf("input 5 strings:\n");
 for (i=0;i<5;i++)
     scanf("%s",pstr[i]);
 p=pstr;
 sort(p);
 printf("strings sorted:\n");
 for (i=0;i<5;i++)
     printf("%s\n",pstr[i]);
}

sort(char **p)                 /*冒泡法对 5 个字符串排序函数*/
```

```
{int i,j;
  char *temp;
  for (i=0;i<5;i++)
    {for (j=i+1;j<5;j++)
       {if (strcmp(*(p+i),*(p+j))>0)          /*比较后交换字符串地址*/
          {temp=*(p+i);
            *(p+i)=*(p+j);
            *(p+j)=temp;
          }
        }
    }
  return 0;
}
```

8-21

```
#include<stdio.h>
int main()
{void sort(int **p,int n);
  int i,n,data[20],**p,*pstr[20];
  printf("input n:\n");
  scanf("%d",&n);
  for (i=0;i<n;i++)
     pstr[i]=&data[i];     //将第 i 个整数的地址赋予指针数组  pstr  的第 i 个元素
  printf("input %d integer numbers:",n);
  for (i=0;i<n;i++)
     scanf("%d",pstr[i]);
  p=pstr;
  sort(p,n);
  printf("Now,the sequence is:\n");
  for (i=0;i<n;i++)
     printf("%d    ",*pstr[i]);
  printf("\n");
  return 0;
}
void sort(int **p,int n)
{int i,j,*temp;
  for (i=0;i<n-1;i++)
    {for (j=i+1;j<n;j++)
        {if (**(p+i)>**(p+j))    //比较后交换整数地址
           {temp=*(p+i);
       *(p+i)=*(p+j);
       *(p+j)=temp;
           }
        }
```

```
    }
}
```

第 9 章用户自己建立数据类型 【第 330 页】

9-1-1

```
#include <stdio.h>
struct
     { int year;
        int month;
        int day;
     }date;
int main()
  {int days;
    printf("input year,month,day:");
    scanf("%d,%d,%d",&date. year,&date.month,&date.day);
    switch(date.month)
    { case 1: days=date.day;        break;
       case 2: days=date.day+31; break;
       case 3: days=date.day+59; break;
       case 4: days=date.day+90; break;
       case 5: days=date.day+120; break;
       case 6: days=date.day+151; break;
       case 7: days=date.day+181; break;
       case 8: days=date.day+212; break;
       case 9: days=date.day+243; break;
       case 10: days=date.day+273; break;
       case 11: days=date.day+304; break;
       case 12: days=date.day+334; break;
  }
  if ((date.year %4== 0 && date.year % 100 != 0
          ||date.year % 400 == 0) && date.month >=3)    days+=1;
  printf("%d/%d is the %dth day in %d.\n",date.month,date.day,days,date.year);
  return 0;
}
```

9-1-2

```
#include <stdio.h>
struct
     { int year;
```

```
        int month;
        int day;
    }date;
int main()
 {int i,days;
  int day_tab[13]={0,31,28,31,30,31,30,31,31,30,31,30,31};
  printf("input year,month,day:");
  scanf("%d,%d,%d",&date. year,&date.month,&date.day);
  days=0;
  for(i=1;i<date.month;i++)
      days=days+day_tab[i];
  days=days+date.day;
  if((date.year%4==0 && date.year%100!=0 || date.year%400==0) && date.month>=3)
    days=days+1;
  printf("%d/%d is the %dth day in %d.\n",date.month,date.day,days,date.year);
  return 0;
}
```

9-2-1

```
#include <stdio.h>
struct y_m_d
    { int year;
        int month;
        int day;
    }date;
int main()
{ int   days(struct y_m_d date1);
  printf("input year,month,day:");
  scanf("%d,%d,%d",&date.year,&date.month,&date.day);
  printf("%d/%d is the %dth day in %d.\n",date.month,date.day,days(date),date.year);
 }

int   days(struct y_m_d date1)
 {int sum;
  switch(date1.month)
     {case 1: sum=date1.day;          break;
      case 2: sum=date1.day+31;    break;
      case 3: sum=date1.day+59;    break;
      case 4: sum=date1.day+90;    break;
      case 5: sum=date1.day+120; break;
      case 6: sum=date1.day+151; break;
      case 7: sum=date1.day+181; break;
      case 8: sum=date1.day+212; break;
```

```
        case 9: sum=date1.day+243; break;
        case 10: sum=date1.day+273; break;
        case 11: sum=date1.day+304; break;
        case 12: sum=date1.day+334; break;
       }
   if ((date1.year % 4 == 0 && date1.year % 100!=0|| date1.year % 400 == 0) &&
date1.month >=3)
        sum+=1;
   return(sum);
}
```

9-2-2

```
#include <stdio.h>
struct y_m_d
       {int year;
        int month;
        int day;
       } date;
int main()
{ int days(int year,int month,int day);
   int days(int,int,int);
   int day_sum;
   printf("input year,month,day:");
   scanf("%d,%d,%d",&date. year,&date.month,&date.day);
   day_sum=days(date.year,date.month,date.day);
   printf("%d / %d is the %dth day in %d.\n",date.month,date.day,day_sum,date.year);
}

int days(int year,int month,int day)
{int day_sum,i;
 int day_tab[13]={0,31,28,31,30,31,30,31,31,30,31,30,31};
 day_sum=0;
 for (i=1;i<month;i++)
    day_sum+=day_tab[i];
 day_sum+=day;
 if ((year%4==0 && year%100!=0 || year%4==0) && month>=3)
        day_sum+=1;
 return(day_sum);
}
```

9-3

```
#include <stdio.h>
```

```
#define N 5

struct student
{ char num[6];
   char name[8];
   int score[4];
}stu[N];

int main()
{void print(struct student stu[6]);
int i,j;
for (i=0;i<N;i++)
{printf("\ninput score of student %d:\n",i+1);
   printf("NO.: ");
   scanf("%s",stu[i].num);
   printf("name: ");
   scanf("%s",stu[i].name);
   for (j=0;j<3;j++)
      {printf("score %d:",j+1);
       scanf("%d",&stu[i].score[j]);
      }
   printf("\n");
}
print(stu);
return 0;
}

void print(struct student stu[6])
  {int i,j;
   printf("\n    NO.        name       score1     score2     score3\n");
   for (i=0;i<N;i++)
     {printf("%5s%10s",stu[i].num,stu[i].name);
      for (j=0;j<3;j++)
         printf("%9d",stu[i].score[j]);
      printf("\n");
     }
  }

9-4
#include <stdio.h>
#define N 5

struct student
  {char num[6];
```

```
    char name[8];
    int score[4];
  } stu[N];

int main()
{void input(struct student stu[]);
  void print(struct student stu[]);
  input(stu);
  print(stu);
  return 0;
}

void input(struct student stu[])
{int i,j;
  for (i=0;i<N;i++)
    {printf("input scores of student %d:\n",i+1);
      printf("NO.: ");
    scanf("%s",stu[i].num);
    printf("name:      ");
          scanf("%s",stu[i].name);
    for (j=0;j<3;j++)
        {printf("score %d:",j+1);
         scanf("%d",&stu[i].score[j]);
    }
    printf("\n");
    }
  }

void print(struct student stu[6])
  {int i,j;
    printf("\n      NO.          name        score1      score2      score3

\n");
    for (i=0;i<N;i++)
      {printf("%5s%10s",stu[i].num,stu[i].name);
        for (j=0;j<3;j++)
            printf("%9d",stu[i].score[j]);
        printf("\n");
      }


9-5
#include <stdio.h>
#define N 10
```

```
struct student
{ char num[6];
   char name[8];
   float score[3];
   float avr;
} stu[N];

int main()
{ int i,j,maxi;
   float sum,max,average;
   for (i=0;i<N;i++)
      {printf("input scores of student %d:\n",i+1);
       printf("NO.:");
       scanf("%s",stu[i].num);
       printf("name:");
       scanf("%s",stu[i].name);
       for (j=0;j<3;j++)
          {printf("score %d:",j+1);
           scanf("%f",&stu[i].score[j]);
          }
      }

   average=0;
   max=0;
   maxi=0;
   for (i=0;i<N;i++)
      {sum=0;
       for (j=0;j<3;j++)
          sum+=stu[i].score[j];
       stu[i].avr=sum/3.0;
       average+=stu[i].avr;
       if (sum>max)
         {max=sum;
          maxi=i;
         }
      }
   average/=N;

   printf("  NO.    name   score1   score2   score3   average\n");
   for (i=0;i<N;i++)
      {printf("%5s%10s",stu[i].num,stu[i].name);
       for (j=0;j<3;j++)
          printf("%9.2f",stu[i].score[j]);
       printf("    %8.2f\n",stu[i].avr);
```

```
    }
    printf("average=%5.2f\n",average);
    printf("The highest score is : student %s,%s\n",stu[maxi].num,stu[maxi].name);
    printf("his scores are:%6.2f,%6.2f,%6.2f,average:%5.2f.\n",
          stu[maxi].score[0],stu[maxi].score[1],stu[maxi].score[2],stu[maxi].avr);
    return 0;
}
```

9-6

```
#include <stdio.h>
#define N 13

struct person
 {int number;
  int nextp;
 } link[N+1];

int main()
 {int i,count,h;
  for (i=1;i<=N;i++)
    {if (i==N)
       link[i].nextp=1;
     else
       link[i].nextp=i+1;
     link[i].number=i;
    }
  printf("\n");
  count=0;
  h=N;
  printf("sequence that persons leave the circle:\n");
  while(count<N-1)
    {i=0;
     while(i!=3)
       {h=link[h].nextp;
        if (link[h].number)
        i++;
       }
     printf("%4d",link[h].number);
     link[h].number=0;
     count++;
    }
  printf("\nThe last one is ");
  for (i=1;i<=N;i++)
    if (link[i].number)
```

```
        printf("%3d",link[i].number);
    printf("\n");
    return 0;
}
```

9-7

```
#include <stdio.h>
struct student
{long num;
 float score;
 struct student *next;
};
int n;

struct student *del(struct student *head,long num)
 {struct student *p1,*p2;
   if (head==NULL)                                  //  是空表
       {printf("\nlist null!\n");
        return(head);
       }
   p1=head;                                         //使 p1 指向第一个结点
   while(num!=p1->num && p1->next!=NULL) //p1 指向的不是所要找的结点且后面还有结点
*/
       {p2=p1;p1=p1->next;}                         // p1 后移一个结点
   if(num==p1->num)                                  //  找到了
       {if(p1==head)head=p1->next; //若 p1 指向的是首结点,把第二个结点地址赋予 head */
        else p2->next=p1->next;       //  否则将下一结点地址赋给前一结点地址
        printf("delete:%ld\n",num);
        n=n-1;
       }
   else printf("%ld not been found!\n",num);       //找不到该结点
   return(head);
}
```

9-8

```
#include <stdio.h>
struct student
{long num;
 float score;
 struct student*next;
};
int n;
```

```
struct student *insert(struct student *head,struct student *stud)
{struct student *p0,*p1,*p2;
  p1=head;                                    //使 p 1 指向第一个结点
  p0=stud;                                    //指向要插入的结点
  if(head==NULL)                              //原来的链表是空表
     {head=p0;p0->next=NULL;}                 //使 p 0 指向的结点作为头结点
  else
     {while((p0->num>p1->num) && (p1->next!=NULL))
         {p2=p1;                                    //使 p 2 指向刚才 p 1 指向的结点
          p1=p1->next;
         }                                //p1 后移一个结点
      if(p0->num<=p1->num)
         {if(head==p1) head=p0;                     //插到原来第一个结点之前
          else p2->next=p0;                         //插到 p2 指向的结点之后
          p0->next=p1;
         }
      else
         {p1->next=p0;
          p0->next=NULL;                            //插到最后的结点之后
         }
     }
   n=n+1;                                    //结点数加 1
   return (head);
}
```

9-9-1

```
#include <stdio.h>
#include <malloc.h>
#define LEN sizeof(struct student)
struct student
   {long num;
    float score;
    struct student *next;
   };
int n;

int main()
   {struct student *creat();
    struct student *del(struct student * ,long);
    struct student *insert(struct student *, struct student *);
    void print(struct student *);
    struct student *head,stu;
```

```
    long del_num;
    printf("input records:\n");
    head=creat();
    print(head);
    printf("input the deleted number:");
    scanf("%ld",&del_num);
    head=del(head,del_num);
    print(head);
    printf("input the inserted record:");
    scanf("%ld,%f",&stu.num,&stu.score);
    head=insert(head,&stu);
    print(head);
    return 0;
  }

struct student *creat()
  {struct student *head;
    struct student *p1,*p2;
    n=0;
    p1=p2=( struct student*) malloc(LEN);
    scanf("%ld,%f",&p1->num,&p1->score);
    head=NULL;
    while(p1->num!=0)
      {n=n+1;
       if(n==1)head=p1;
       else p2->next=p1;
       p2=p1;
       p1=(struct student*)malloc(LEN);
       scanf("%ld,%f",&p1->num,&p1->score);
      }
    p2->next=NULL;
    return(head);
  }

struct student *del(struct student *head,long num)
  {struct student *p1,*p2;
    if (head==NULL)
       {printf("\nlist null!\n");
        return(head);
       }
    p1=head;
    while(num!=p1->num && p1->next!=NULL)
        {p2=p1;p1=p1->next;}
    if(num==p1->num)
```

```
        {if(p1==head)head=p1->next;
         else p2->next=p1->next;
         printf("delete:%ld\n",num);
         n=n-1;
        }
    else printf("%ld not been found!\n",num);
    return(head);
  }

struct student *insert(struct student *head, struct student *stud)
    {struct student *p0,*p1,*p2;
     p1=head;
     p0=stud;
     if(head==NULL)
        {head=p0; p0->next=NULL;}
     else
        {while((p0->num>p1->num) && (p1->next!=NULL))
            {p2=p1;
             p1=p1->next;
            }
         if(p0->num<=p1->num)
            {if(head==p1) head=p0;
             else p2->next=p0;
             p0->next=p1;
            }
         else
            {p1->next=p0; p0->next=NULL;}
         }
      n=n+1;
      return(head);
     }

void print(struct student *head)
    {struct student *p;
     printf("\nNow,These %d records are:\n",n);
     p=head;
     if(head!=NULL)
        do
        {printf("%ld %5.1f\n",p->num,p->score);
         p=p->next;
        }while(p!=NULL);
    }
```

9-9-2

```
#include <stdio.h>
#include <malloc.h>
#define NULL 0
#define LEN sizeof(struct student)
struct student
   {long num;
    float score;          struct student *next;
   };
int n;

int main()
   {struct student *creat();
    struct student *del(struct student * ,long    );
    struct student *insert(struct student *, struct student *);
    void print(struct student *);
    struct student *head,stu;
    long del_num;
    printf("input records:\n");
    head=creat();
    print(head);
    printf("input the deleted number:");
    scanf("%ld",&del_num);
    head=del(head,del_num);
    print(head);
    printf("input the inserted record:");
    scanf("%ld,%f",&stu.num,&stu.score);
    head=insert(head,&stu);
    print(head);
    printf("input the inserted record:");
    scanf("%ld,%f",&stu.num,&stu.score);
    head=insert(head,&stu);
    print(head);
    return 0;
   }

struct student *creat()
   {struct student *head;
    struct student *p1,*p2;
    n=0;
    p1=p2=( struct student*) malloc(LEN);
    scanf("%ld,%f",&p1->num,&p1->score);
    head=NULL;
```

```
    while(p1->num!=0)
      {n=n+1;
       if(n==1)head=p1;
       else p2->next=p1;
       p2=p1;
       p1=(struct student*)malloc(LEN);
       scanf("%ld,%f",&p1->num,&p1->score);
      }
    p2->next=NULL;
    return(head);
  }

struct student *del(struct student *head,long num)
  {struct student *p1,*p2;
    if (head==NULL)
       {printf("\nlist null!\n");
        return(head);
       }
    p1=head;
    while(num!=p1->num && p1->next!=NULL)
        {p2=p1;p1=p1->next;}
    if(num==p1->num)
        {if(p1==head)head=p1->next;
         else p2->next=p1->next;
         printf("delete:%ld\n",num);
         n=n-1;
        }
    else printf("%ld not been found!\n",num);
    return(head);
  }

struct student *insert(struct student *head, struct student *stud)
    {struct student *p0,*p1,*p2;
     p1=head;
     p0=stud;
     if(head==NULL)
        {head=p0; p0->next=NULL;}
     else
        {while((p0->num>p1->num) && (p1->next!=NULL))
            {p2=p1;
             p1=p1->next;
            }
         if(p0->num<=p1->num)
            {if(head==p1) head=p0;
```

```
            else p2->next=p0;
            p0->next=p1;
           }
        else
           {p1->next=p0; p0->next=NULL;}
        }
      n=n+1;
      return(head);
    }

void print(struct student *head)
    {struct student *p;
     printf("\nNow,These %d records are:\n",n);
     p=head;
     if(head!=NULL)
        do
        {printf("%ld %5.1f\n",p->num,p->score);
         p=p->next;
        }while(p!=NULL);
    }


9-9-3
#include <stdio.h>
#include <malloc.h>
#define LEN sizeof(struct student)
struct student
   {long num;
    float score;
    struct student *next;
   };
int n;

int main()
   {struct student *creat();
    void print(struct student *);
    struct student *del(struct student *,long);
    struct student *insert(struct student *, struct student *);
    struct student *head,*stu;
    long del_num;
    printf("input records:\n");
    head=creat();
    print (head);
    printf("input the deleted number:");
```

```
    scanf("%ld",&del_num);
    while (del_num!=0)
        {head=del(head,del_num);
    print (head);
    printf ("input the deleted number:");
    scanf("%ld",&del_num);}
    printf("\ninput the inserted record:");
    stu=(struct student *) malloc(LEN);
    scanf("%ld,%f",&stu->num,&stu->score);
    while(stu->num!=0)
       {head=insert(head,stu);
        print(head);
        printf("input the inserted record:");
        stu=(struct student *)malloc(LEN);
        scanf("%ld,%f",&stu->num,&stu->score);
       }
    return 0;
}

struct student *creat()
   {struct student *head;
        struct student *p1,*p2;
        n=0;
        p1=p2=( struct student*) malloc(LEN);
        scanf("%ld,%f",&p1->num,&p1->score);
        head=NULL;
        while(p1->num!=0)
          {n=n+1;
        if(n==1)head=p1;
        else p2->next=p1;
        p2=p1;
        p1=(struct student*)malloc(LEN);
        scanf("%ld,%f",&p1->num,&p1->score);
       }
        p2->next=NULL;
        return(head);
}

struct student *del(struct student *head,long num)
    {struct student *p1,*p2;
     if (head==NULL)
        {printf("\nlist null!\n");return(head);}
         p1=head;
         while(num!=p1->num && p1->next!=NULL)
```

```
            {p2=p1;p1=p1->next;}
             if(num==p1->num)
                {if(p1==head)head=p1->next;
                 else p2->next=p1->next;
                 printf("delete:%ld\n",num);
                 n=n-1;
                }
             else printf("%ld not been found!\n",num);
      return(head);
    }

struct student *insert(struct student *head, struct student *stud)
    {struct student *p0,*p1,*p2;
     p1=head;
     p0=stud;
     if(head==NULL)
        {head=p0; p0->next=NULL;}
     else
     {while((p0->num>p1->num) && (p1->next!=NULL))
         {p2=p1;
          p1=p1->next;
         }
      if(p0->num<=p1->num)
         {if(head==p1) head=p0;
          else p2->next=p0;
          p0->next=p1;}
      else
          {p1->next=p0; p0->next=NULL;}
     }
     n=n+1;
     return(head);
    }

void print(struct student *head)
     {struct student *p;
      printf("\nNow,These %d records are:\n",n);
      p=head;
      if(head!=NULL)
         do
           {printf("%ld %5.1f\n",p->num,p->score);
            p=p->next;
           }while(p!=NULL);
      }
```

9-10

```
#include <stdio.h>
#include <malloc.h>
#define LEN sizeof(struct student)

struct    student
{long num;
  int score;
  struct student *next;
};

struct student lista,listb;
int n,sum=0;

int main()
{struct student *creat(void);
  struct student *insert(struct student    *,struct student *);
  void print(struct student *);
  struct student *ahead,*bhead,*abh;
  printf("input list a:\n");
  ahead=creat();
  sum=sum+n;
  printf("input list b:\n");
  bhead=creat();
  sum=sum+n;
  abh=insert(ahead,bhead);
  print(abh);
  return 0;
}

struct student *creat(void)            //建立链表函数
  {struct student *p1,*p2,*head;
    n=0;
    p1=p2=(struct student *)malloc(LEN);
    printf("input number & scores of student:\n");
    printf("if number is 0,stop inputing.\n");
    scanf("%ld,%d",&p1->num,&p1->score);
    head=NULL;
    while(p1->num !=0)
       {n=n+1;
         if (n==1)
            head=p1;
         else
```

```
            p2->next=p1;
          p2=p1;
          p1=(struct student *)malloc(LEN);
          scanf("%ld,%d",&p1->num,&p1->score);
        }
        p2->next=NULL;
    return(head);
}

struct student *insert(struct student *ah,struct student *bh)      //插入函数
  {struct student * pa1,* pa2,* pb1,* pb2;
    pa2=pa1=ah;
    pb2=pb1=bh;
    do
    {while((pb1->num>pa1->num) && (pa1->next !=NULL))
        {pa2=pa1;
          pa1=pa1->next;
        }
       if (pb1->num <= pa1->num)
        {if (ah==pa1)
            ah=pb1;
          else
            pa2->next=pb1;
          pb1=pb1->next;
          pb2->next=pa1;
          pa2=pb2;
          pb2=pb1;
        }
       }while ((pa1->next!=NULL) || (pa1==NULL && pb1!=NULL));
       if ((pb1!=NULL) && (pb1->num>pa1->num) && (pa1->next==NULL))
          pa1->next=pb1;
       return(ah);
  }

void print(struct student *head)    //输出函数
    {struct student    *p;
     printf("There are %d records:    \n",sum);
     p=head;
     if (p !=NULL)
        do
          {printf("%ld %d\n",p->num,p->score);
            p=p->next;
          }while (p !=NULL);
     }
```

9-11

```
#include <stdio.h>
#include <string.h>
#define LA 4
#define LB 5
struct    student
   {int num;
    char name[8];
    struct student *next;
   } a[LA],b[LB];

int main()
{struct student a[LA]={{101,"Wang"},{102,"Li"},{105,"Zhang"},{106,"Wei"}};
 struct student b[LB]={{103,"Zhang"},{104,"Ma"},{105,"Chen"},{107,"Guo"},{108,"lui"}};
 int    i;
 struct student *p,*p1,*p2,*head1,*head2;

 head1=a;
 head2=b;
 printf(" list A:    \n");
  for (p1=head1,i=1;i<=LA;i++)
     {if(i<LA) p1->next=a+i;
      else p1->next=NULL;
      printf("%4d%8s\n",p1->num,p1->name);
      if(i<LA) p1=p1->next;
     }
  printf("\n list B:\n");
  for (p2=head2,i=1;i<=LB;i++)
     {if (i<LB) p2->next=b+i;
      else p2->next=NULL;
      printf("%4d%8s\n",p2->num,p2->name);
      if (i<LB) p2=p2->next;
     }

  p1=head1;
  while(p1!=NULL)
     {p2=head2;
      while ((p1->num != p2->num) && (p2->next!=NULL))
         p2=p2->next;
      if (p1->num == p2->num)
         {if (p1==head1)
              head1=p1->next;
```

```
            else
               {p->next=p1->next;p1=p1->next;}
           }
        else
            {p=p1;p1=p1->next;}
        }

   printf("\nresult:\n");
   p1=head1;
   while(p1!=NULL)
      {printf("%4d %7s    \n",p1->num,p1->name);
       p1=p1->next;
      }
   return 0;
}
```

9-12

```
#include <stdio.h>
#include <malloc.h>
#define LEN sizeof(struct    student)
struct    student
{    char num[6];
     char name[8];
     char sex[2];
     int age;
     struct student *next;
} stu[10];

int main()
{ struct student *p,*pt,*head;
   int i,length,iage,flag=1;
   int find=0;                    //找到待删除元素  find=1,否则 find=0
   while (flag==1)
     {printf("input length of list(<10):");
      scanf("%d",&length);
      if (length<10)
         flag=0;
     }

      //建立链表
   for (i=0;i<length;i++)
       {p=(struct student *) malloc(LEN);
        if (i==0)
```

```
            head=pt=p;
        else
            pt->next=p;
        pt=p;
        printf("NO.:");
        scanf("%s",p->num);
        printf("name:");
        scanf("%s",p->name);
        printf("sex:");
        scanf("%s",p->sex);
        printf("age:");
        scanf("%d",&p->age);
    }
p->next=NULL;
p=head;
printf("\n NO.       name       sex    age\n");          //显示
while(p!=NULL)
     {printf("%4s%8s%6s%6d\n",p->num,p->name,p->sex,p->age);
      p=p->next;
     }

   //   删除
printf("input age:");                 //输入待删年龄
scanf("%d",&iage);
pt=head;
p=pt;
if (pt->age==iage)                    //链头是待删元素
   {p=pt->next;
    head=pt=p;
    find=1;
   }
else                                    //链头不是待删元素
   pt=pt->next;
while (pt!=NULL)
   {if (pt->age==iage)
      {p->next=pt->next;
       find=1;
      }
    else                                 //  中间结点不是待删元素
       p=pt;
    pt=pt->next;
   }
if (!find)
   printf(" not found    %d.",iage);
```

```
  p=head;
  printf("\n NO.       name       sex    age\n"); //显示结果
  while (p!=NULL)
     {printf("%4s%8s",p->num,p->name);
      printf("%6s%6d\n",p->sex,p->age);
      p=p->next;
     }
  return 0;
 }
```

第 10 章对文件的输入输出 【第 354 页】

10-3

```
#include <stdio.h>
#include <string.h>
#include <stdlib.h>
int main ()
{
 FILE *fp;
 char str[100];
 int i=0;
 if ((fp=fopen("a1","w"))==NULL)
    { printf("can not open file\n");
       exit(0);
    }
 printf("input a string:\n");
 gets(str);
 while (str[i]!='!')
  {if (str[i]>='a'&& str[i]<='z')
       str[i]=str[i]-32;
    fputc(str[i],fp);
    i++;
  }
 fclose(fp);
 fp=fopen("a1","r");
 fgets(str,strlen(str)+1,fp);
 printf("%s\n",str);
 fclose(fp);
 return 0;
}
```

10-4

```
#include <stdio.h>
#include <stdlib.h>
int main ()
{
  FILE *fp;
  int i,j,n,i1;
  char c[100],t,ch;
  if ((fp=fopen("a1","r"))==NULL)
     { printf("\ncan not open file\n");
        exit(0);
     }
  printf("file A :\n");
  for (i=0;(ch=fgetc(fp))!=EOF;i++)
   {
     c[i]=ch;
     putchar(c[i]);
   }
  fclose(fp);

  i1=i;
  if ((fp=fopen("b1","r"))==NULL)
   {printf("\ncan not open file\n");
     exit(0);
   }
  printf("\nfile B:\n");
  for (i=i1;(ch=fgetc(fp))!=EOF;i++)
     {c[i]=ch;
       putchar(c[i]);
     }
  fclose(fp);

  n=i;
  for (i=0;i<n;i++)
     for (j=i+1;j<n;j++)
          if (c[i]>c[j])
               {t=c[i];
                 c[i]=c[j];
                 c[j]=t;
               }
  printf("\nfile C :\n");
  fp=fopen("c1","w");
  for (i=0;i<n;i++)
         {putc(c[i],fp);
          putchar(c[i]);
```

```
        }
  printf("\n");
  fclose(fp);
  return 0;
}

10-5-1
#include <stdio.h>
struct student
{char num[10];
  char name[8];
  int score[3];
  float ave;
  }    stu[5];

int main()
  { int i,j,sum;
     FILE *fp;
     for(i=0;i<5;i++)
     {printf("\ninput score of student %d:\n",i+1);
     printf("NO.:");
     scanf("%s",stu[i].num);
     printf("name:");
     scanf("%s",stu[i].name);
     sum=0;
     for (j=0;j<3;j++)
        {printf("score %d:",j+1);
         scanf("%d",&stu[i].score[j]);
         sum+=stu[i].score[j];
     }
     stu[i].ave=sum/3.0;
     }

      /*将数据写入文件*/
   fp=fopen("stud","w");
   for (i=0;i<5;i++)
       if (fwrite(&stu[i],sizeof(struct student),1,fp)!=1)
      printf("file write error\n");
   fclose(fp);

   fp=fopen("stud","r");
   for (i=0;i<5;i++)
      {fread(&stu[i],sizeof(struct student),1,fp);
       printf("\n%s,%s,%d,%d,%d,%6.2f\n",stu[i].num,stu[i].name,stu[i].score[0],
```

```
            stu[i].score[1],stu[i].score[2],stu[i].ave);}
    return 0;
    }


10-5-2
#include <stdio.h>
#define SIZE 5
struct student
{char name[10];
  int num;
  int score[3];
  float ave;
  }    stud[SIZE];

int main()
  { void save(void);
     int i;
     float sum[SIZE];
     FILE *fp1;
     for (i=0;i<SIZE;i++)
        { scanf("%s %d %d %d %d",stud[i].name,&stud[i].num,&stud[i].score[0],
            &stud[i].score[1],&stud[i].score[2]);
          sum[i]=stud[i].score[0]+stud[i].score[1]+stud[i].score[2];
          stud[i].ave=sum[i]/3;
         }
  save();
  fp1=fopen("stu.dat","rb");
  printf("\n name          NO.       score1    score2    score3      ave\n");
  printf("-----------------------------------------------\n");
  for (i=0;i<SIZE;i++)
     {fread(&stud[i],sizeof(struct student),1,fp1);
      printf("%-10s %3d %7d %7d %7d %8.2f\n",stud[i].name,stud[i].num,
      stud[i].score[0],stud[i].score[1],stud[i].score[2],stud[i].ave);
     }
  fclose (fp1);
  return 0;
  }

  void save(void)
  {
     FILE *fp;
     int i;
     if ((fp=fopen("stu.dat","wb"))==NULL)
```

```
        {printf("The file can not open\n");
         return;
        }
    for(i=0;i<SIZE;i++)
       if (fwrite(&stud[i],sizeof(struct student),1,fp)!=1)
          {printf("file write error\n");
    return;
          }
    fclose(fp);
 }
```

10-6-1

```
#include <stdio.h>
#include <stdlib.h>
#define N 10
struct student
{char num[10];
 char name[8];
 int score[3];
 float ave;
 } st[N],temp;

int main()
 {FILE *fp;
  int i,j,n;

        /*读文件*/
  if ((fp=fopen("stud","r"))==NULL)
     {printf("can not open.\n");
      exit(0);
     }
  printf("File 'stud': ");
  for (i=0;fread(&st[i],sizeof(struct student),1,fp)!=0;i++)
     {printf("\n%8s%8s",st[i].num,st[i].name);
      for (j=0;j<3;j++)
         printf("%8d",st[i].score[j]);
      printf("%10.2f",st[i].ave);
     }
  printf("\n");
  fclose(fp);
  n=i;

        /*排序*/
```

```
    for (i=0;i<n;i++)
        for (j=i+1;j<n;j++)
       if (st[i].ave < st[j].ave)
          {temp=st[i];
           st[i]=st[j];
           st[j]=temp;
          }

          /*输出*/
    printf("\nNow:");
    fp=fopen("stu_sort","w");
    for (i=0;i<n;i++)
        {fwrite(&st[i],sizeof(struct student),1,fp);
         printf("\n%8s%8s",st[i].num,st[i].name);
         for (j=0;j<3;j++)
            printf ("%8d",st[i].score[j]);
         printf("%10.2f",st[i].ave);
        }
    printf("\n");
    fclose(fp);
    return 0;
  }

10-6-2
#include <stdio.h>
#include <stdlib.h>
#define SIZE 5
struct student
{
  char name[10];
  int num;
  int score[3];
  float ave;
  }    stud[SIZE],work;
int main()
  {
     void sort(void);
     int i;
     FILE *fp;
     sort();
     fp=fopen("stud_sort.dat","rb");
     printf("sorted student's scores list as follow\n");
     printf("-----------------------------------------------------\n");
     printf(" NAME          N0.         SCORE1     SCORE2     SCORE3       AVE        \n");
```

```
    printf("----------------------------------------------------\n");
    for (i=0;i<SIZE;i++)
          {
    fread(&stud[i],sizeof(struct student),1,fp);
    printf("%-10s %3d %8d %8d %8d %9.2f\n",stud[i].name,stud[i].num,
            stud[i].score[0],stud[i].score[1],stud[i].score[2],stud[i].ave);
          }
    fclose(fp);
    return 0;
}

void sort(void)
 {FILE *fp1,*fp2;
    int i,j;
    if ((fp1=fopen("stu.dat","rb"))==NULL)
       {printf("The file can not open\n\n");
        exit(0);
       }
    if ((fp2=fopen("stud_sort.dat","wb"))==NULL)
       {printf("The file write error\n");
        exit(0);
       }
    for (i=0;i<SIZE;i++)
       if (fread(&stud[i],sizeof(struct student),1,fp1)!=1)
          {printf("file read error\n");
           exit(0);
          }
    for (i=0;i<SIZE;i++)
       {for (j=i+1;j<SIZE;j++)
          if (stud[i].ave<stud[j].ave)
             {work=stud[i];
              stud[i]=stud[j];
              stud[j]=work;
             }
        fwrite(&stud[i],sizeof(struct student),1,fp2);
       }
    fclose(fp1);
    fclose(fp2);
}
```

10-7

```
#include <stdio.h>
#include <stdlib.h>
```

```
struct student
{char num[10];
  char name[8];
  int score[3];
  float ave;
  }    st[10],s;

  int main()
  {FILE *fp,*fp1;
    int i,j,t,n;
    printf("\nNO.:");
    scanf("%s",s.num);
    printf("name:");
    scanf("%s",s.name);
    printf("score1,score2,score3:");
    scanf("%d,%d,%d",&s.score[0],&s.score[1],&s.score[2]);
    s.ave=(s.score[0]+s.score[1]+s.score[2])/3.0;

              /*从文件读数据*/
    if((fp=fopen("stu_sort","r"))==NULL)
       {printf("can not open file.");
        exit(0);
       }
    printf("original data:\n");
       for (i=0;fread(&st[i],sizeof(struct student),1,fp)!=0;i++)
          {printf("\n%8s%8s",st[i].num,st[i].name);
              for (j=0;j<3;j++)
                 printf("%8d",st[i].score[j]);
            printf("%10.2f",st[i].ave);
          }

    n=i;
    for (t=0;st[t].ave>s.ave && t<n;t++);

              /*向文件写数据*/
    printf("\nNow:\n");
    fp1=fopen("sort1.dat","w");
    for (i=0;i<t;i++)
       {fwrite(&st[i],sizeof(struct student),1,fp1);
        printf("\n %8s%8s",st[i].num,st[i].name);
        for (j=0;j<3;j++)
           printf("%8d",st[i].score[j]);
        printf("%10.2f",st[i].ave);
       }
```

```
   fwrite(&s,sizeof(struct student),1,fp1);
   printf("\n %8s %7s %7d %7d %7d%10.2f",s.num,s.name,s.score[0],
            s.score[1],s.score[2],s.ave);

   for (i=t;i<n;i++)
      {fwrite(&st[i],sizeof(struct student),1,fp1);
       printf("\n %8s%8s",st[i].num,st[i].name);
       for(j=0;j<3;j++)
          printf("%8d",st[i].score[j]);
       printf("%10.2f",st[i].ave);
       }
   printf("\n");
   fclose(fp);
   fclose(fp1);
   return 0;
  }
```

10-8

```
#include <stdio.h>
#include <stdlib.h>
struct student
{char num[10];
  char name[8];
  int score[3];
  float ave;
  }    st[10],s;

  int main()
  {FILE *fp,*fp1;
    int i,j,t,n;
    printf("\nNO.:");
    scanf("%s",s.num);
    printf("name:");
    scanf("%s",s.name);
    printf("score1,score2,score3:");
    scanf("%d,%d,%d",&s.score[0],&s.score[1],&s.score[2]);
    s.ave=(s.score[0]+s.score[1]+s.score[2])/3.0;

              /*从文件读数据*/
    if((fp=fopen("stu_sort","r"))==NULL)
       {printf("can not open file.");
        exit(0);
       }
```

```
   printf("original data:\n");
      for (i=0;fread(&st[i],sizeof(struct student),1,fp)!=0;i++)
         {printf("\n%8s%8s",st[i].num,st[i].name);
             for (j=0;j<3;j++)
                printf("%8d",st[i].score[j]);
          printf("%10.2f",st[i].ave);
         }

   n=i;
   for (t=0;st[t].ave>s.ave && t<n;t++);

               /*向文件写数据*/
   printf("\nNow:\n");
   fp1=fopen("sort1.dat","w");
   for (i=0;i<t;i++)
      {fwrite(&st[i],sizeof(struct student),1,fp1);
       printf("\n %8s%8s",st[i].num,st[i].name);
       for (j=0;j<3;j++)
          printf("%8d",st[i].score[j]);
       printf("%10.2f",st[i].ave);
      }
   fwrite(&s,sizeof(struct student),1,fp1);
   printf("\n %8s %7s %7d %7d %7d%10.2f",s.num,s.name,s.score[0],
            s.score[1],s.score[2],s.ave);

   for (i=t;i<n;i++)
      {fwrite(&st[i],sizeof(struct student),1,fp1);
       printf("\n %8s%8s",st[i].num,st[i].name);
       for(j=0;j<3;j++)
          printf("%8d",st[i].score[j]);
       printf("%10.2f",st[i].ave);
       }
   printf("\n");
   fclose(fp);
   fclose(fp1);
   return 0;
 }
```

10-9

```
#include <stdio.h>
#include <stdlib.h>
#include <string.h>
struct emploee
```

```
{char       num[6];
 char       name[10];
 char       sex[2];
 int        age;
 char       addr[20];
 int        salary;
 char       health[8];
 char       class[10];
 }em[10];

 struct emp
 {char name[10];
  int   salary;
 }em_case[10];

int main()
 {FILE *fp1,*fp2;
  int i,j;
  if ((fp1=fopen("emploee","r"))==NULL)
    {printf("can not open file.\n");
     exit(0);
    }
  printf("\n  NO.    name   sex    age      addr     salary    health   class\n");
  for (i=0;fread(&em[i],sizeof(struct emploee),1,fp1)!=0;i++)
     {printf("\n%4s%8s%4s%6d%10s%6d%10s%8s",em[i].num,em[i].name,em[i].sex,
           em[i].age,em[i].addr,em[i].salary,em[i].health,em[i].class);
      strcpy(em_case[i].name,em[i].name);
      em_case[i].salary=em[i].salary;
     }
    printf("\n\n    ********************************          ");
    if((fp2=fopen("emp_salary","wb"))==NULL)
      {printf("can not open file\n");
       exit(0);
      }
    for (j=0;j<i;j++)
      {if(fwrite(&em_case[j],sizeof(struct emp),1,fp2)!=1)
          printf("error!");
       printf("\n    %12s%10d",em_case[j].name,em_case[j].salary);
      }
    printf("\n     ******************************          ");
    fclose(fp1);
    fclose(fp2);
    return 0;
 }
```

10-10

```
#include <stdio.h>
#include <stdlib.h>
#include <string.h>
struct emploee
{char    name[10];
 int     salary;
}emp[20];

int main()
 { FILE *fp;
    int i,j,n,flag;
    char name[10];
    if ((fp=fopen("emp_salary","rb"))==NULL)
       {printf("can not open file.\n");
        exit(0);
       }
    printf("\noriginal data:\n");
    for (i=0;fread(&emp[i],sizeof(struct emploee),1,fp)!=0;i++)
        printf("\n    %8s      %7d",emp[i].name,emp[i].salary);
    fclose(fp);
    n=i;
    printf("\ninput name deleted:\n");
    scanf("%s",name);
    for (flag=1,i=0;flag && i<n;i++)
       {if (strcmp(name,emp[i].name)==0)
           {for (j=i;j<n-1;j++)
               {strcpy(emp[j].name,emp[j+1].name);
                emp[j].salary=emp[j+1].salary;
               }
            flag=0;
           }
       }
    if(!flag)
       n=n-1;
    else
       printf("\nnot found!");
    printf("\nNow,The content of file:\n");
    if((fp=fopen("emp_salary","wb"))==NULL)
       {printf("can not open file\n");
        exit(0);
       }
```

```
    for (i=0;i<n;i++)
        fwrite(&emp[i],sizeof(struct emploee),1,fp);
    fclose(fp);
    fp=fopen("emp_salary","r");
    for (i=0;fread(&emp[i],sizeof(struct emploee),1,fp)!=0;i++)
        printf("\n%8s      %7d",emp[i].name,emp[i].salary);
    printf("\n");
    fclose(fp);
    return 0;
 }
```

10-11

```
#include <stdio.h>
int main()
 { int i,flag;
    char str[80],c;
    FILE *fp;
    fp=fopen("text","w");
    flag=1;
    while(flag==1)
       {printf("input string:\n");
        gets(str);
        fprintf(fp,"%s ",str);
        printf("continue?");
        c=getchar();
        if ((c=='N')||(c=='n'))
           flag=0;
        getchar();
       }
    fclose(fp);
    fp=fopen("text","r");
    while(fscanf(fp,"%s",str)!=EOF)
       {for (i=0;str[i]!='\0';i++)
           if ((str[i]>='a') && (str[i]<='z'))
                str[i]-=32;
        printf("%s\n",str);
       }
    fclose(fp);
    return 0;
 }
```

11-1

```
#include <stdio.h>
#define swap(a,b)t=b;b=a;a=t
int main()
{
   int a,b,t;
   printf("input two integer a,b:");
   scanf("%d,%d",&a,&b);
   swap(a,b);
   printf("Now,a=%d,b=%d\n",a,b);
   return 0;
 }
```

11-2

```
#include <stdio.h>
#define SURPLUS(a,b)((a)%(b))
int main()
{
   int a,b;
   printf("input two integer a,b:");
   scanf("%d,%d",&a,&b);
   printf("remainder is %d\n",SURPLUS(a,b));
   return 0;
 }
```

11-4

```
#include<stdio.h>
#define LEAP_YEAR(y)((y%4==0)&&(y%100!=0)||(y%400==0))
int main()
{
 int year;
 printf("\ninput year:");
 scanf("%d",&year);
 if(LEAP_YEAR(year))
    printf("%d is a leap year.\n",year);
```

```
  else
     printf("%d is not a leap year. \n",year);
  return 0;
}
```

11-5

```
#include <stdio.h>
#define NL putchar('\n')
#define PR(format,value) printf("value=%format\t",(value))
#define PRINT1(f,x1) PR(f,x1);NL
#define PRINT2(f,x1,x2) PR(f,x1);PRINT1(f,x2)
int main()
{
   float x=5.0,x1=3.0,x2=8.0;
   char d='f';
   PR(d,x);
   PRINT1(d,x);
   PRINT2(d,x1,x2);
   return 0;
}
```

11-6

```
#include<stdio.h>
#define PR printf
#define NL "\n"
#define Fs "%f"
#define F    "%6.2f"
#define F1 F NL
#define F2 F"\t" F NL
#define F3 F"\t" F "\t" F NL
int main()
{float a,b,c;
  PR("input three floating number a,b,c:\n");
  scanf(Fs,&a);
  scanf(Fs,&b);
  scanf(Fs,&c);
  PR(NL);
  PR("output one floating number each line:\n");
  PR(F1,a);
  PR(F1,b);
  PR(F1,c);
  PR(NL);
```

```
  PR("output two floating number:\n");
  PR(F2,a,b);
  PR(F1,c);
  PR(NL);
  PR("output three floating number:\n");
  PR(F3,a,b,c);
  return 0;
}
```

11-7

```
#include <stdio.h>
#include "format.h"
int main ()
{
   int d,num;
   float f;
   char s[80];
   printf("choice data format: 1-integer,2-float,3-string:");
   scanf("%d",&num);
   switch(num)
   {case 1: printf("input integer: ");
                scanf("%d",&d);
                INTEGER(d);
                break;
     case 2: printf("input float: ");
                scanf("%f",&f);
                FLOAT(f);
                break;
     case 3: printf("input string: ");
                scanf("%s",&s);
                STRING(s);
                break;
     default: printf("input error!\n");
     }
  return 0;
  }
```

11-8-1

```
#include<stdio.h>
int main()
{int max(int x,int y,int z);
  int a,b,c;
  printf("input three integer: ");
```

```
 scanf("%d,%d,%d",&a,&b,&c);
 printf("max=%d\n",max(a,b,c));
 return 0;
 }
```

11-8-2

```
#include<stdio.h>
#define MAX(a,b)((a)>(b)?(a):(b))
int main()
{
    int a,b,c;
    printf("input three integer: ");
    scanf("%d,%d,%d",&a,&b,&c);
    printf("max=%d\n",MAX(MAX(a,b),c));
    return 0;
 }
```

11-10

```
#include <stdio.h>
#define MAX 80
#define CHANGE 1
int main()
{
 char str[MAX];
 int i;
 printf("input text:\n");
 gets(str);
 #if(CHANGE)
   {for(i=0;i<MAX;i++)
      {if(str[i]!='\0')
         if(str[i]>='a'&&str[i]<'z'||str[i]>='A'&&str[i]<='Z')
              str[i]+=1;
         else if(str[i]=='z'||str[i]=='Z')
              str[i]-=25;
        }
      }
 #endif
 printf("output:\n%s\n",str);
 return 0;
 }
```

12-1

```
#include <stdio.h>
```

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

12-2

```
#include <stdio.h>
int main()
{unsigned short a,b,c;
  int n;
  printf("please enter a & n:\n");
  scanf("a=%o,n=%d",&a,&n);
  b=a<<(16-n);
  c=a>>n;
  c=c|b;
  printf("a:%o\nc:%o\n",a,c);
  return 0;
}
```