

## 实验题目：循环结构程序设计

实验目的：

- 1 熟悉 VC++6.0 的运行环境，掌握 C 程序的执行方法；
- 2 掌握三种基本数据类型、部分运算符和常用函数；
- 3 掌握三种循环控制语句及 break 和 continue 语句的使用方法；
- 4 掌握循环结构的嵌套及正确控制计数型循环结构次数的方法。

实验内容：

程序 1：搬砖问题：36 块砖，36 人搬，男搬 4，女搬 3，两个小孩抬 1 砖，要求一次搬完，问男、女和小孩各若干？

```
#include "stdio.h"
main()
{
    int men, women, children;
    men=0;
    while (men<=8)
        {women=0;
            while (women<=11)
                {children=36-men-women;
                    if ((men*4+women*3+children/2==36)&&children%2==0)
                        printf("men=%d, women=%d, children=%d", men, women, children);
                    women++;}
                men++;}
}
```

程序 2 打印形状为直角三角形的乘法口诀。

```
#include "stdio.h"
main()
{int i, j, result;
for (i=1; i<10; i++)
{ for (j=1; j<i; j++)
{
result=i*j;
printf("%d*%d=%-3d", i, j, result) /*表示左对齐, 占 3 位*/
}
printf("\n") /*行后换行*/
}
}
```

程序 3 求 1000 以内的所有素数。

```
#include "stdio.h"
main( )
{
    int m, i;
    for (m=2; m<=100; m++)
        { for (i=2; i<=m-1; i++)
            if (m%i==0) break;
            if (i>m-1) printf("%5d", m);
        }
}
```

程序 4：计算和统计旅客运费程序。

由键盘输入旅客行李重量 (w)，按公式：

若 (  $w \leq 50\text{kg}$  )  $f = 1.5 * w$  若 (  $w > 50\text{kg}$  )  $f = 1.5 * 50 + 2.8 * (w - 50)$  (  $w > 50\text{kg}$  )

---

计算运费，要求：

(1) 打印每位旅客的顺序号，行李重量，应付运费；

(2) 假定每天最多办理 100 人的行李托运手续或当  $w \leq 0$  时循环结束，下班时打印出全天的运费总收入和办理的总人数。

方法一（while 结构）

```
#include "stdio.h"
main( )
{
    int num=0;
    float w,f,total=0;
    scanf("%f",&w);
    while (w>0&&num<100 )
        { if (w<=50) f=1.5*w;
          else f=1.5*50+2.8*(w-50);
          num++;
          printf ( "%d,%6.0f,%8.2f\n",num,w,f);
          total+=f; scanf ("%f",&w); }
    printf ("Customers=%d  TotalIncome=%10.2f\n",num,total);
}
```

方法二（do-while 结构）

```
#include "stdio.h"
main( )
{
    int num=0;
    float w,f,total=0;
    do
        { scanf("%f",&w);
          if (w<=0) break;
          if (w<=50) f=1.5*w;
          else f=1.5*50+2.8*(w-50);
          num++;
          printf ("%d %6.0f %8.2f\n",num,w,f);
          total+=f;
        } while (num<100);
    printf ("Customers=%d  TotalIncome=%10.2f\n",num,total);
}
```

方法三（for 结构）

```
#include "stdio.h"
main( )
{
    int num;
    float w,f,total=0;
    scanf("%f",&w);
    for (num=1;w>0&&num<=100;num++)
        {if (w<=50) f=1.5*w;
          else f=1.5*50+2.8*(w-50);
          printf ("%d %6.0f %8.2f\n",num,w,f);
          total+=f; scanf ("%f",&w);
        }
    printf ("Customers=%d  TotalIncome=%10.2f\n",num,total);
}
```



## 实验二：数组

实验目的：1. 更加灵活的使用一维数组和二维数组

2. 把数组和指针结合使用

实验内容：

### 1. 约瑟夫环问题

由 n 个人围成一个首尾相连的圈报数。从第一个人开始，从 1 开始报数，报到 3 的人出圈，剩下的人继续从 1 开始报数，直到所有的人都出圈为止。求出所有人的出圈顺序。

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

```
main()
```

```
{int a[10];
```

```
    int n=8, i, k=0, j=1, len;
```

```
    len=n;
```

```
    i=0;
```

```
    //给数组赋值
```

```
    for (i=0; i<n; i++)
```

```
        a[i]=i+1;
```

```
    //i 为元素下标，j 代表当前要报的数，len 表示圈中当前人数
```

```
    while(len>0) {
```

```
        if(a[k%n]>0) {
```

```
            if(j%3==0) {
```

```
                printf("%d\n", a[k%n]); //找到要出圈的人，并把圈中人数减一，最后
```

一个人才输出

```
                a[k%n]=-1;
```

```
                j=1;
```

```
                k++;
```

```
                len--;
```

```
            }
```

```
            else{//报数不是 3
```

```
                k++;
```

```
                j++;
```

```
            }
```

```
        }
```

```
        else//遇到空位了，就跳到下一位，但 j 不加一，也就是这个位置没有报数
```

```
            k++;
```

```
    }
```

```
}
```

### 2. 学生成绩统计与分析

输入 10 名学生的英语、C 语言、计算机文化基础成绩，统计每名学生的三科成绩的平均分，分析三门课程的优秀率，85 分以上为优秀。

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

```
void main()
```

```
{
```

```
    int grade[10][4];
```

```
    int i, j, average, egood, cgood, jgood;
```

```

    for (i=0; i<10; i++)
    {
        printf("请输入第%d 个学生的英语、c 语言、计算机\n", i+1);
        for (j=0; j<3; j++)
            scanf("%d", &grade[i][j] );
    }
    egood=cgood=jgood=0;
    for (i=0; i<10; i++)        //10 个人
    {
        average=0;
        for (j=0; j<3; j++)    //三门课程
        {
            average+=grade[i][j];
            if(grade[i][j]>=85)
            {
                if(j==0) egood++; if(j==1) cgood++; if(j==2) jgood++;
            }
        }
        grade[i][3]=average/3;
    }

    for (i=0; i<10; i++)
    printf("第%d 个学生的三科平均分为: %d\n", i+1, grade[i][3]);
    printf("英语成绩优秀的学生为%d\n", egood);
    printf("C 语言成绩优秀的学生为%d\n", cgood);
    printf("计算机成绩优秀的学生为%d\n", jgood);
}

```

### 3. 利用指针实现数组合并

数组 a: “1, 7, 9, 11, 13, 15, 17, 19”； 数组 b: “2, 4, 6, 8, 10”， 两个数组合并为数组 c，按升序排列。

```

#include<stdio.h>
void main()
{ int paixu (int c[]);
  int a[8]={1, 7, 9, 11, 13, 15, 17, 19};
  int b[5]={2, 3, 6, 8, 10};
  int c[13];
  int *p, *q, *s;
  for (p=a, s=c; p<8+a; p++, s++)
    *s=*p;
  for (q=b; q<b+5; q++, s++)
    *s=*q;
  paixu(c);
}
paixu(int c[])
{int i, j, k;

```

---

```
for (j=0; j<12; j++)
for (i=0; i<12-j; i++)
if (c[i]>c[i+1])
{ k=c[i];
c[i]=c[i+1];
c[i+1]=k;
}
for (i=0; i<13; i++)
printf("%d\n", c[i]);
}
```

### 实验三：函数

#### 实验目的：

1. 更加灵活的使用数组和指针作为函数参数；
2. 更加灵活使用指针处理字符串。

#### 实验内容：

##### 1. 字符串复制

输入一个长度不大于 30 的字符串，将此字符串中从第 m 个字符开始的剩余全部字符复制成为另一个字符串，并将这个新字符串输出。要求用指针方法处理字符串。

##### 2. 插入字符串

从键盘上先后读入两个字符串，假定存储在字符数组 s1 和 s2 中。注意，这两个字符串最长均可达到 26 个字符、最短均可 0 个字符。将字符串 s2 插入字符串 s1 中，生成一个新的字符串 s。插入方法为 s2 的第 i 个字符插入到原 s1 的第 i 个字符后，如果 s2 比 s1 (假定 s1 的长度为 L1) 长，则 s2 的第 L1 个字符开始到 s2 结尾的所有字符按在 s2 中的顺序放在新生成的字符串 s 之后。用指针完成程序。

##### 3. 学生成绩管理系统

某班期末考试科目为数学 (MT)、英语 (EN) 和物理 (PH)，有最多不超过 30 人参加考试。

- (1) 计算每个学生的总分；
- (2) 按总分成绩由高到低排出成绩的名次；
- (3) 打印出名次表，包括名次、学生编号、各科分数、总分；

要求程序运行后先打印出一个菜单，提示用户选择：成绩录入、成绩排序、成绩查找，在选择某项功能后执行相应的操作。

## 实验四：结构体

### 实验目的：

1. 更加灵活的使用数组作为函数参数；
2. 初步掌握开发一个小型实用系统的基本方法；
3. 初步掌握书写程序设计开发文档的能力。

### 实验内容：

#### 程序一：学生信息管理系统

编写程序，实现学生信息的管理（包括学生信息的建立、增加、删除、修改、查找），要求：

#### 小组分工情况

小组分工情况统计表

班级	信息工程学院 计算机科学与技术专业 2012 级 软件工程（方向） 班____组				
名称	学生信息管理系统				
项目组长	（本次项目小组组长）				
组员					
组员学号					
工作简介					

### 一 需求分析

I 设计一个学生成绩管理系统至少具有以下功能：

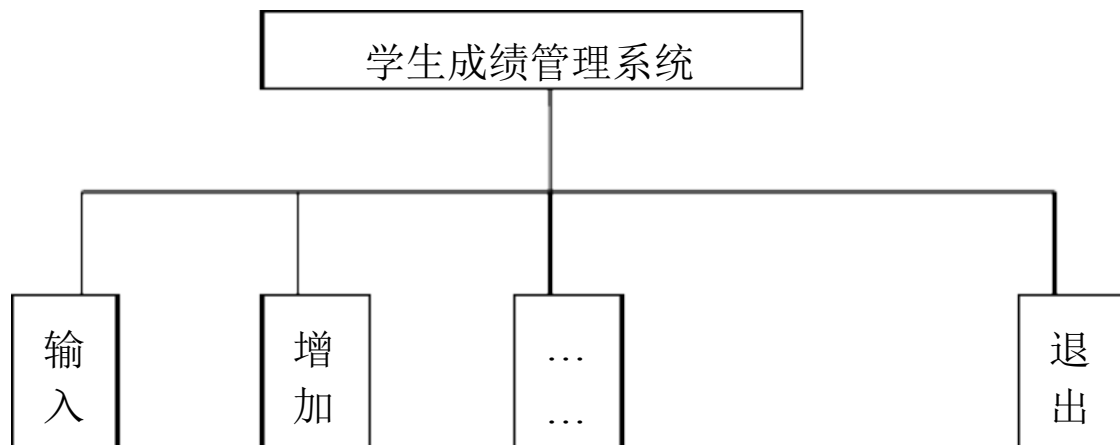
- (1) 菜单功能：采用文本菜单界面。
- (2) 学生信息的建立。
- (3) 学生信息的增加。
- (4) 学生信息的删除。
- (5) 学生信息的修改。
- (6) 学生信息的查找。

### II 数据库结构

```
struct student
{
    charnum[10];
    char name[10];
    char xibie[10];
    char sex;
    int age;
}st[30];
```

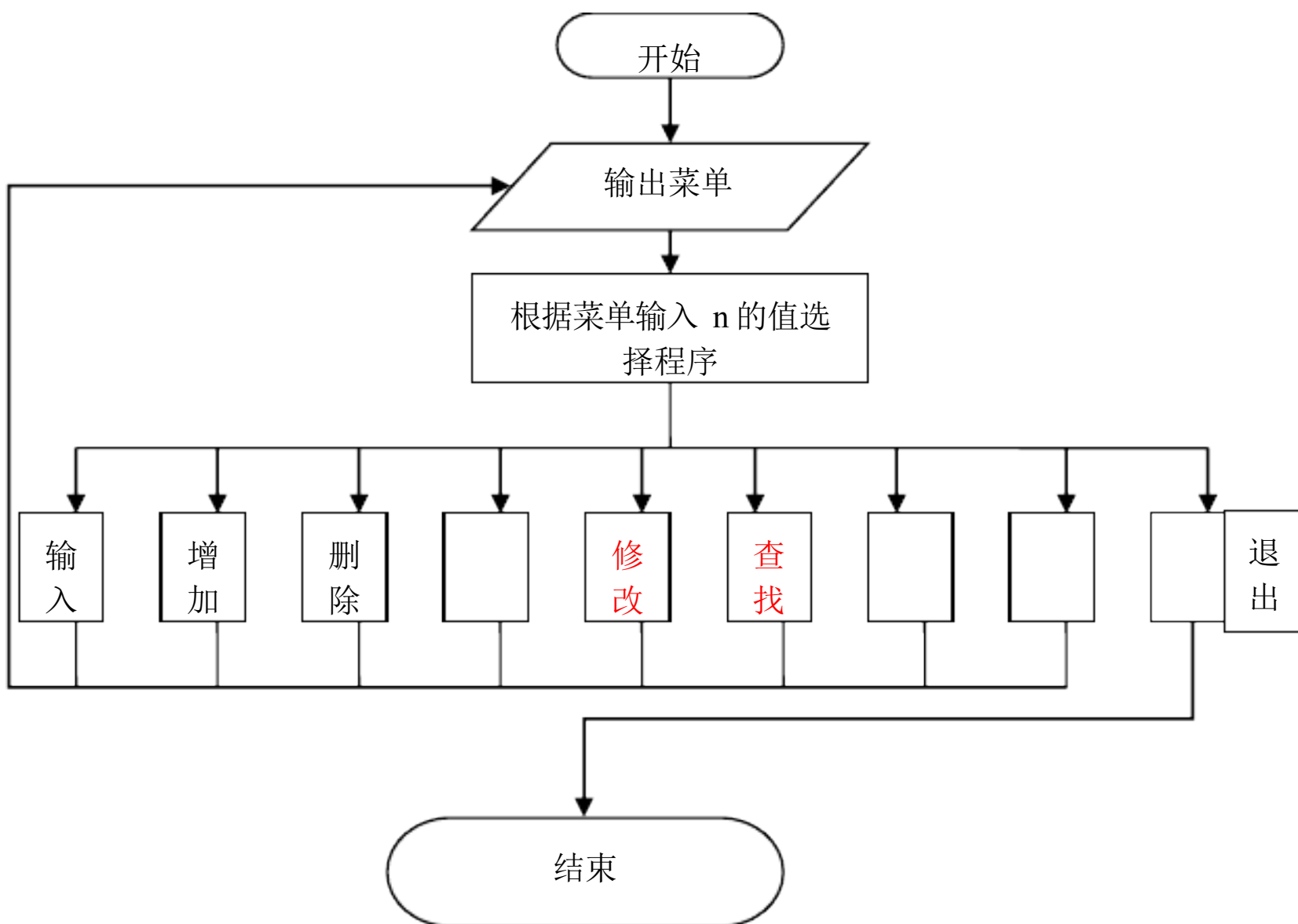
### 二 总体设计

框架结构图（系统的功能结构图）





程序流程图：(visio)



三、详细设计

(流程图!!)

四、程序源代码(参考答案)

1. 用结构体数组完成

2. 用结构体指针完成

1. #include "stdio.h"

#include "string.h"

typedef structnode

{ charnum[10]; char

name[10]; char

xibie[10]; char sex;

int age;

}st;

int n;

st \*creslink()

{

int i;

st s[10];

printf("请输入要录入的人数: ");

scanf("%d",&n);

for (i=0;i<=n-1;i++)

{ printf("请输入第%d 名学生的学号, 姓名, 系别, 年龄, 性别: ",i+1);

scanf("%s%s%s%d%c",s[i].num,s[i].name,s[i].xibie,&s[i].age,&s[i].sex);

}

return s;

```

}
int getlen(st s[])
{
    return n;
}
insert(st s[],char num[],char name[],char xibie[],int age,char sex)
{ strcpy(s[n].num,num);
  strcpy(s[n].name,name);
  strcpy(s[n].xibie,xibie); s[n].age=age;
  s[n].sex=sex;
  n++;
}
delete(st s[],char num[])
{int i,j;
 for(i=0;i<n;i++)
   if(strcmp(s[i].num,num)==0)break;
 for(j=i+1;j<n;j++)
   s[j-1]=s[j];
 n--;
}
void list(sts[])
{ int i;
  printf("学号\t姓名\t系别\t年龄\t性别\n");
  for(i=0;i<n;i++)
    printf("%s\t%s\t%s\t%d\t%c\n",s[i].num,s[i].name,s[i].xibie,s[i].age,s[i].sex);
}
void main()
{
  int k,f,age;
  char num[10],name[10],xibie[10],sex; st *s;
  while(1)
  {
    printf("\n\n\t\t\t学生信息管理系统\n");
    printf("\t\t\t*****\n"); printf("\t\t\t1.
    录入学生信息\n");
    printf("\t\t\t2.插入新学生信息\n"); printf("\t\t\t3.删除
    学生信息\n"); printf("\t\t\t4.输出学生信息\n");
    printf("\t\t\t5.输出学生总数\n"); printf("\t\t\t6.退
    出\n");
    printf("\t\t\t*****\n");
    printf("请选择:");
    scanf("%d",&k);
    switch(k)
    {case 1:s=creslink();break;
     case 2: printf("请输入新学生的学号, 姓名, 系别, 年龄, 性别: ");
              scanf("%s%s%s%d%c",num,name,xibie,&age,&sex); insert(s,num,name,xibie,age,sex);
              break;
    }
  }
}

```

```

    case 3:printf("请输入要删除的学生的学号: ");
        scanf("%s",num);
        delete(s,num); break;
    case 4:list(s);break;
    case 5:f=getlen(s);printf("学生总数为: %d",f);break; case
    6:return;
}
}
}

2. #include "stdio.h"
#include "string.h"
#define null 0
typedef struct node
{ charnum[10]; char
  name[10]; char
  xibie[10]; char sex;
  int age;
  struct node *next;
}slink;
slink *creslink()
{
slink *head,*p,*s;
int i,n;
printf("请输入要录入的人数: ");
scanf("%d",&n);
if(n<1)
return null;
p=head=(slink *)malloc(sizeof(slink)); head-
>next=null;
for (i=1;i<=n;i++)
{
s=(slink *)malloc(sizeof(slink));
printf("请输入第%d 名学生的学号, 姓名, 系别, 年龄, 性别: ",i);
scanf("%s%s%s%d%c",s->num,s->name,s->xibie,&s->age,&s->sex);
p->next=s;
p=s;
}
p->next=null; return
head;
}
int getlen(slink *head)
{
slink *p;
int n;
p=head->next; n=0;
while (p!=null)

```

```

    {
n++;
p=p->next;
    }
    return n;
}
insert(slink *head,char num[],char name[],char xibie[],int age,char sex)
{
    slink *p,*q;
    p=head;
    while(p->next!=null)    p=p->next;
    q=(struct node *)malloc (sizeof(slink));
    strcpy(q->num,num);
    strcpy(q->name,name);
    strcpy(q->xibie,xibie); q-
    >sex=sex;
    q->age=age;
    q->next=p->next; p-
    >next=q;
}
delete(slink *head,char num[])
{
    slink *p,*q;
    p=head;
    while(p->next!=null&&strcmp(p->next->num,num)!=0) p=p-
        >next;
    q=p->next;
    p->next=q->next; free(q);
}
void list(slink *head)
{
    slink *p;
    printf("学号\t姓名\t系别\t年龄\t性别\n");
    p=head->next;
    while(p!=null)
    {
        printf("%s\t%s\t%s\t%d\t%c\n",p->num,p->name,p->xibie,p->age,p->sex); p=p->next;
    }
    printf("\n");
}
void main()
{
    int k,f,age;
    char num[10],name[10],xibie[10],sex; slink
    *head;
    while(1)
    {
        printf("\n\n\t\t\t学生信息管理系统\n");
        printf("\t\t\t*****\n"); printf("\t\t\t1.
        录入学生信息\n");
    }
}

```

```
printf("\t\t2.插入新学生信息\n"); printf("\t\t3.删除
学生信息\n"); printf("\t\t4.输出学生信息\n");
printf("\t\t5.输出学生总数\n"); printf("\t\t6.退
出\n");
printf("\t\t*****\n");
printf("请选择:");
scanf("%d",&k);
switch(k)
{case 1:head=creslink();break;
case 2: printf("请输入新学生的学号, 姓名, 系别, 年龄, 性别: ");
scanf("%s%s%s%d%c",num,name,xibie,&age,&sex); insert(head,num,name,xibie,age,sex);
break;
case 3:printf("请输入要删除的学生的学号: ");
scanf("%s",num);
delete(head,num);
break;
case 4:list(head);break;
case 5:f=getlen(head);printf("学生总数为: %d",f);break; case
6:return;
}
}
}
```

---

## 实验五 文件操作

实验目的:

1. 初步了解 C 语言中文件操作的类型;
2. 掌握文件的打开、关闭及读写的操作
3. 能熟练进行字符及数据块的读写操作, 为后续的课程设计打好基础。

实验内容:

程序 1: 从键盘上输入若干个学生的学号及考试成绩, 将这些数据写入磁盘文件 y.dat 中。(由学号等于 0 作终止标记)

```
#include "stdio.h"
main()
{ struct student
  { int num;
    float score; };
  struct student stu;
  FILE *fp;
  fp=fopen("y.dat","w");
  scanf("%d%f",&stu.num,&stu.score);
  while(stu.num!=0)
  { fprintf(fp,"%d %f\n",stu.num,stu.score);
    scanf("%d%f",&stu.num,&stu.score);}
  fclose(fp);}
```

程序 2: 设有一文件 cj.dat 存放了 50 个人的成绩 (英语、计算机、数学), 存放格式为: 每人一行, 成绩间由逗号分隔。计算三门课平均成绩, 统计个人平均成绩大于或等于 90 分的学生人数

```
.
#include <stdio.h> main()
{FILE *fp;
  int i=0,num=0;
  float x,y,z,s1=0,s2=0,s3=0;
  fp=fopen("cj.txt","r");
  while(i<=50)
  {fscanf(fp,"%f,%f,%f",&x,&y,&z);
  s1=s1+x;
  s2=s2+y; s3=s3+z;
  if((x+y+z)/3>=90)
  num=num+1; i++;}
  printf("分数高于 90 的人数为: %d",num);
  fclose(fp);}
```

程序 3: 统计上题 cj.dat 文件中每个学生的总成绩, 并将原有数据和计算出的总分数存放在磁盘文件 "stud" 中。

```
#include "stdio.h"
main()
{ FILE *fp1,*fp2; float
  x,y,z;
  fp1=fopen("cj.txt","r");
  fp2=fopen("stud.txt","w");
  while(!feof(fp1))
```

```
{ fscanf(fp1,"%f,%f,%f",&x,&y,&z);  
  printf("%f,%f,%f,%f\n",x,y,z,x+y+z);  
fprintf(fp2,"%f,%f,%f,%f\n",x,y,z,x+y+z);}  
fclose(fp1); fclose(fp2);}
```

## 实验六：万年历

### 实验目的：

- 1、复习、巩固 C 语言的基础知识，进一步加深对 C 语言的理解和掌握；
- 2、将课本上的理论知识和实际有机的结合起来，锻炼分析解决实际问题的能力
- 3、提高学生适应实际，实践编程的能力。

### 实验内容：

程序一：编写万年历

### 小组分工情况

小组分工情况统计表

班级	信息工程学院 计算机科学与技术专业 2012 级 软件工程（方向） 班____组				
名称	学生成绩管理系统				
项目组长	（本次项目小组组长）				
组员					
组员学号					
工作简介					

### 一 需求分析

万年历能实现一些功能，主要包括以下几点：

(1) 日历显示功能。能根据系统日期进行初始化，如果没有任何输入，则显示系统日期所在月份的月历。

判断星期，但根据蔡勒公式，可以很容易地有年月日推断出星期：

```
nDay=year-1+(year-1)/4-(year-1)/100+(year-1)/400+nday;
```

```
w=nDay%7;
```

其中 year：要查询的年份 nday：

从该年的一月一日起到该天的天数 w：得

出的星期数，w==0 表示星期天。

在打印月历时，只要知道第一天的星期，就可以依次输入后面的日期，通过定长度输出日期，在遇到某天时星期六时，就换行输出，因此可以按照月历的格式输出。对于初始输入当前月历，可以通过<time.h>中的 time 和 localtime 函数取出系统时间进行初始化。

```
time_t timer;
```

```
struct tm* gmt;
```

```
time(&timer);
```

```
gmt=localtime(&timer);
```

gmt 就是一个时间结构体，可以从中取出年月日。

(2) 日历查询、闰年判断功能。输入日期进行查询，显示查询日期所在月份的月历，还将显示所在年份是否为闰年

能被 4 整除但不能被 100 整除，或者能被 400 整除的年份为闰年，否则为平年。日期的查询，就是直接打印改天所在月份的月历

(3) 参数检查功能。如果输入日期不合法，查询将被拒绝，并显示提示信息。

判断日期是否合法，可以先判断年月日是否有小于 0 的值，月和日是否有超出最大值 12 和



31，如果通过此判断合法，再判断是否时闰年，如果时闰年，则判断二月份是否满足要求，闰年二月份不能超过 29 天，平年二月份不能超过 28 天。在查询过程中，如果发现输入不合法或者不满足输入要求，则要求重新输入。

(4) 通过键盘操作（如：光标上、下、左、右移动键，pageup、pagedown 键）调整年份、月份和日期，并显示调整后的日期及月历。

捕获键盘的输入，通过两个 getch(); 将后一个 getch() 的返回值付给整型或字符型 k, k 为相应键的 ascii 值，判断 ascii 值，就可以知道时哪个键的输入。若捕获到某键的输入，则设置相应的年或月或日的值，再进行刷新输出，就可以实现日期的调整。

## 二 总体设计 (visio)

本系统包括 4 个模块, 分别是功能控制模块、打印输出模块、日历显示模块和键值获取模块。

1、功能控制模块。该模块用于实现日期有效性检查、判断是否是闰年和返回指定日期对应的日期。

2、打印输出模块。该模块主要是进行输出显示，包括打印指定个数的空格、打印分隔线、打印用户使用手册、打印当前日期对应的星期以及打印指定的星期。

3、日历显示模块。该模块是系统的核心模块，是用来显示指定日期所在月份的月历。

4、键值获取模块。该模块接受键盘操作，获取键值，来进行日期调整，并调用日历显示模块中的函数来显示调整后的日历。

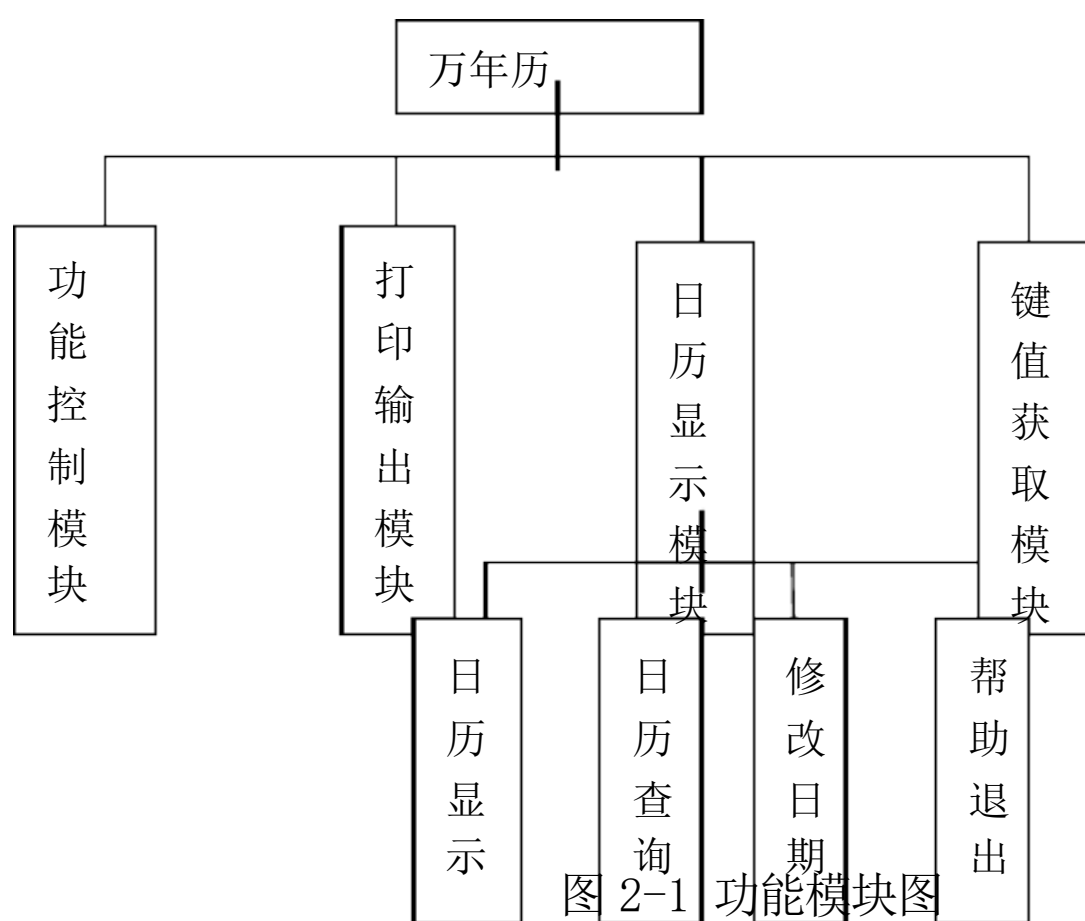


图 2-1 功能模块图

## 总函数流程图

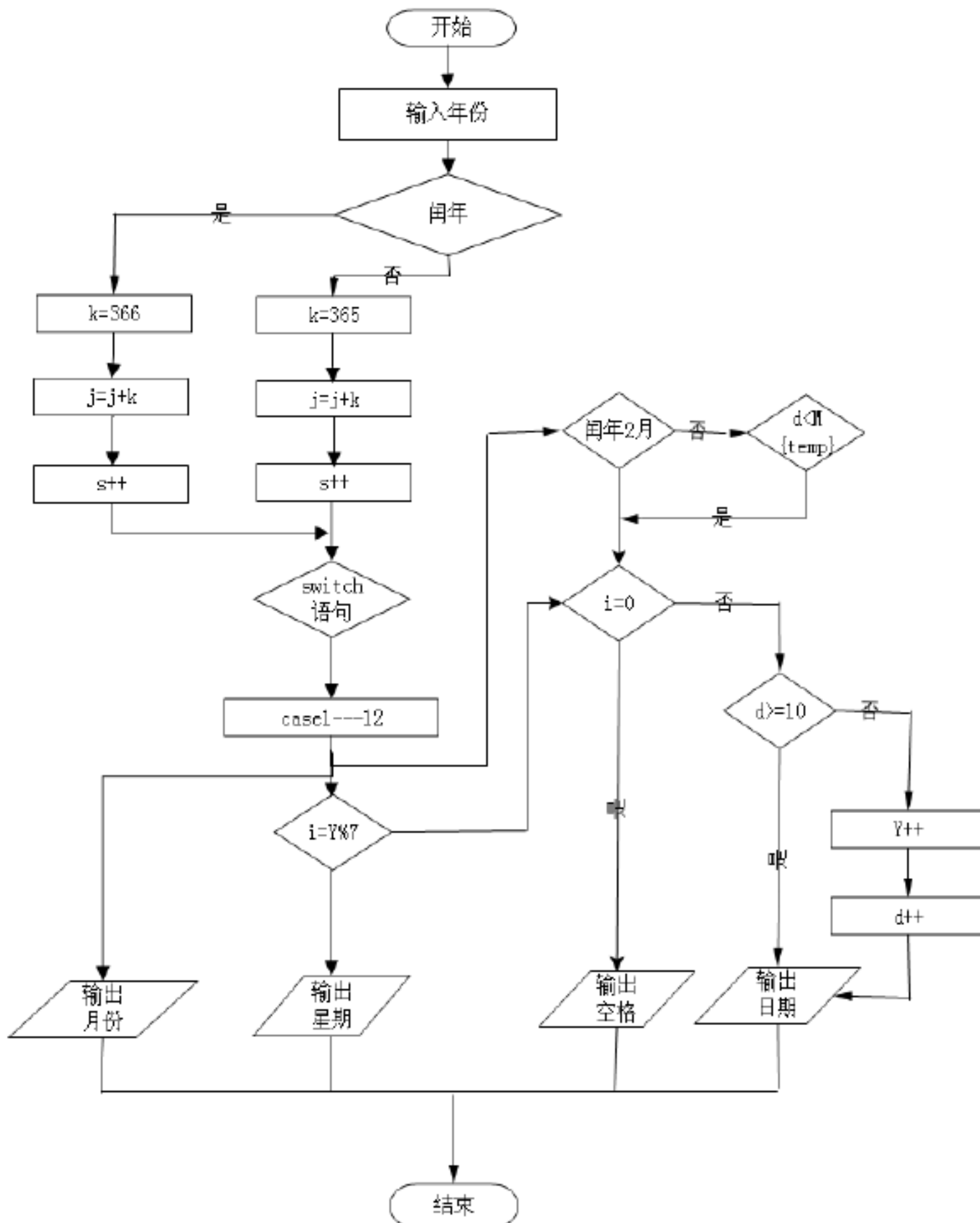


图 3-1 总函数流程图

### 三、详细设计

(1) 由于万年历具有以下特点：

平年 365 天 (52 周+1 天)，闰年 366 天 (52 周+2 天)。平年 2 月 28 天，闰年 2 月 29 天。

(2) 根据万年历以上特点进行编写

1) 首先对万年历年、月、日进行编写，编写程序先定义每月的天数为 28 天，如月份为 1、3、5、7、8、10、12 就定义天数为 31 天反之如果月份为 4、6、9、11 就输出天数为 30 天，由上可见 2 月份为 28 天但是如果为闰年就有 29 天就要定义一个数组存放天数，用 while 循环控制。

2) 再对其中的星期进行编写：

由于公元 1 月 1 日设为星期六，故 3 月 1 日为星期三，可以用万年 3 月 1 日星期算法 (特别是那个三) 由于其公式为：某年 3 月 1 日星期几 = (百年 % 4 \* 5 天 + 年 / 4 \* 5 天 + 年 % 4 + 月星期表 + 日 + 2 天) % 7

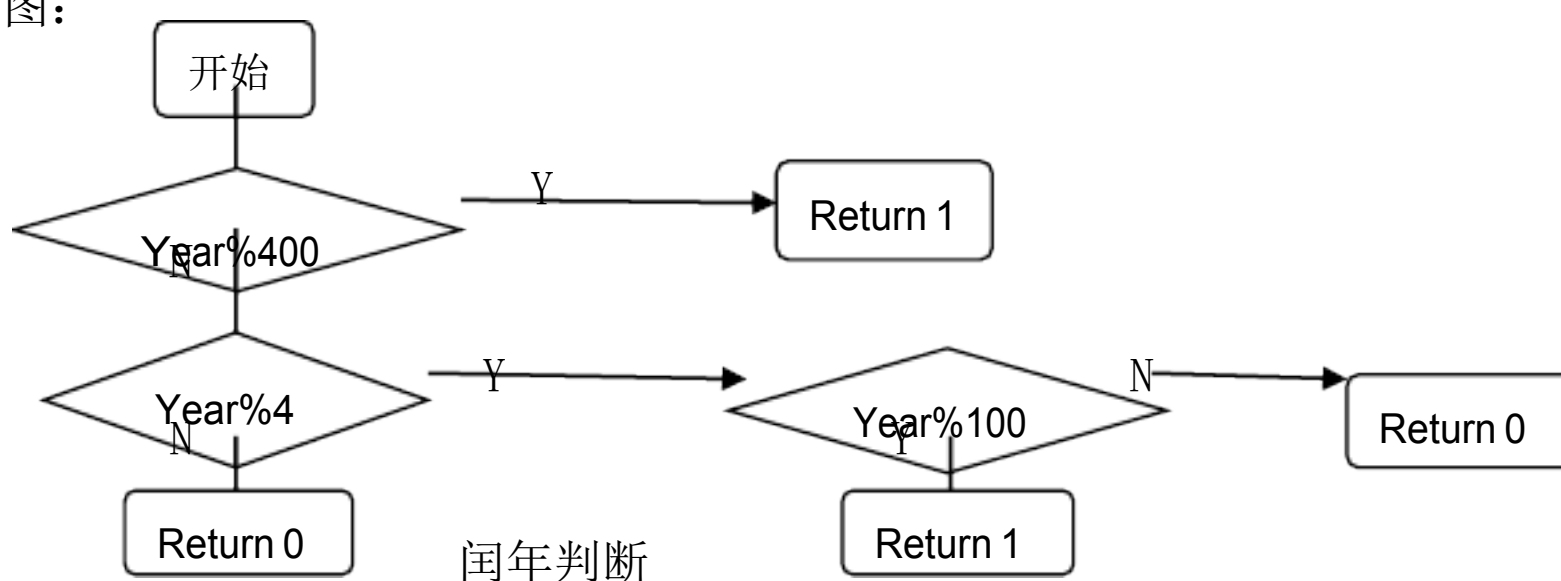
闰 4 百年 3 月 1 日星期算法 (百年 % 4 = 0) 其公式为：某年 3 月 1 日星期几 = (年 + 年 / 4 + 月星期表 + 日 + 2 天) % 7 例：2000 年 3 月 1 日星期几 = (0 + 0 / 4 + 0 + 1 + 2) % 7 = 3 % 7 = 星期三

平 4 百年 3 月 1 日星期算法 (百年%4<>0) 其公式为: 某年 3 月 1 日星期几=(百年%4\*5 天+年+年 /4+ 月 星期 表 + 日 +2 天 )%7 例: 2100 年 3 月 1 日 星期 几 = (21%4\*5+0/4\*5+0%4+3) %7=(5+0+0+3)%7=8%7=星期一(注意: 2100 年是平年)  
其中万年某日星期算法 其公式为: 某日星期几 =(百年%4\*5 天+年+年/4+月星期表+日+2天)%7

(1) int isleapYear(int year);  
该函数判断某年是否是闰年, 参数year表示要判断的年份, 返回值1表示该年为闰年, 0表示该年为平年。

```
int IsLeapYear( int year )
{if ((year %4 == 0) && (year % 100 != 0) || (year % 400 == 0) )
return 1; else return 0;
```

流程图:



(2) struct tm\* getDay();  
该函数用以取得系统的时间。该函数没有参数, 返回值类型为 struct tm\*, 是一个包含时间的结构体

```
struct tm* getDay() //从系统取得当前时间
{
time_t timer;
struct tm* gmt;
time(&timer);
gmt=localtime(&timer);
return gmt;
}
```

(3) int getwDay(int year, int month, int day);  
该函数用以根据一个日期推断星期, 参数为要判断的年月日。返回值为整型, 分别表示星期一到星期六, 0表示星期天。

```
int getwDay(int year, int month, int day) //取得星期
{
int nday=0, nDay, i, w;
for(i=0; i<month-1; i++)
nday+=daysOfMonth[runYear(year)][i];
nday+=day;
nDay=year-1+(year-1)/4-(year-1)/100+(year-1)/400+nday;
w=nDay%7;
return w;
}
```

---

(流程图!!)

#### 四、程序源代码

```
#include<stdio.h>
#include<time.h>
#include<math.h>
#include<windows.h>
#include<conio.h>

#define KEYNUMUp 0x48
#define KEYNUMDown 0x50
#define KEYNUMLeft 0x4b
#define KEYNUMRight 0x4d
#define KEYNUMPageUp 0x49
#define KEYNUMPageDown 0x51

int year,month,day; //全局变量记录时间
int daysOfMonth[2][12]={{31,28,31,30,31,30,31,31,30,31,30,31},{31,29,31,30,31,30,31,31,30,31,30,31}};
int runYear(int year) //判断是否是闰年
{
    int flag=0; if(year%400==0||(year%4==0&&year%100!=0))
        flag=1;
    return flag;
}
struct tm* getDay()//从系统取得当前时间
{
    time_t timer;
    struct tm* gmt;
    time(&timer);
    gmt=localtime(&timer);
    return gmt;
}
int dayExame(int year,int month,int day)//检查日期是否正确
{
    if(year<0||month<1||month>12||day<1||day>31) return 0;
    switch(month)
    {
        case 1:
        case 3:
        case 5:
        case 7:
        case 8:
        case 10:
        case 12:
            if(day>31)return 0;break; case
        4:
        case 6:
        case 9:
```

以上内容仅为本文档的试下载部分，为可阅读页数的一半内容。如要下载或阅读全文，请访问：

<https://d.book118.com/477102033141006133>