**1、万年历**

#include<stdio.h>

void main()

{

int year, month, day,day\_y=365,i,daysum=0,feb=28,week,daym=0,day\_year=0;

printf("请输入年月日 如“2000.1.1”（不得早于2000年）\n");

scanf\_s("%d.%d.%d", &year, &month, &day);

for (i = 2000; i <= year-1; i++)

{

if ((i % 4 == 0 && i % 100 != 0) || (i % 400 == 0))day\_y = 366;

else day\_y = 365;

daysum += day\_y;

}

if ((year % 4 == 0 && year % 100 != 0) || (year % 400 == 0))feb = 29;

switch (month)

{

case 12:daym += 30;

case 11:daym += 31;

case 10:daym += 30;

case 9:daym += 31;

case 8:daym += 31;

case 7:daym += 30;

case 6:daym += 31;

case 5:daym += 30;

case 4:daym += 31;

case 3:daym += feb;

case 2:daym += 31;

}

day\_year = daym + day;

daysum += daym+day+5;

week = daysum % 7;

printf("这一天是%d年第%d天星期",year,day\_year);

switch (week)

{

case 0:printf("日"); break;

case 1:printf("一"); break;

case 2:printf("二"); break;

case 3:printf("三"); break;

case 4:printf("四"); break;

case 5:printf("五"); break;

case 6:printf("六"); break;

}

}

2、查找数组

#include<stdio.h>

#include<time.h>

#include<stdlib.h>

void main()

{

int a[10];

int i,n,j,temp, min=0,max=9,mid=0;

srand(time(NULL));

for(i=0;i<=8;i++) a[i]=rand()%100+1;

a[9]=60;

scanf("%d",&n);

for(i=0;i<9;i++)

for(j=0;j<9-i;j++)

if(a[j]>a[j+1])

{

temp=a[j];

a[j]=a[j+1];

a[j+1]=temp;

}

while(min<=max)

{

mid=min+(max-min)/2;

if(a[mid]==n)

break;

else if(a[mid]<n)

min=mid+1;

else if(a[mid]>n)

max=mid-1;

}

if(min<=max)

printf("该数在此数组内");

else

printf("不在");

}

3、学生课程成绩

#include<stdio.h>

#include<time.h>

#include<stdlib.h>

#include<math.h>

int i, j,x, y;

void avest(float a[11][6])

{

float s = 0;

for (i = 0; i <= 9; i++)

{

for (j = 0; j <= 4; j++)s += a[i][j];

a[i][5] = s / 5;

s = 0;

}

}

void avegr(float a[11][6])

{

float s = 0;

for (j = 0; j <= 4; j++)

{

for (i = 0; i <= 9; i++)s += a[i][j];

a[10][j] = s / 10;

s = 0;

}

}

void maxar(float a[11][6])

{

float mx = a[0][0];

for (i = 0; i <= 9; i++)

for (j = 0; j <= 4; j++)if (a[i][j] > mx)

{

mx = a[i][j];

x = i + 1, y = j + 1;

}

}

float avesqr(float a[11][6])

{

float s = 0, s2 = 0,sr=0;

for (i = 0; i <= 9; i++)

{

s += a[i][5];

s2 += pow(a[i][5], 2);

}

sr = s2 / 10 - pow(s / 10, 2);

return sr;

}

void printar(float a[11][6])

{

for(i=0;i<=10;i++)

{

for (j = 0; j <= 5; j++)printf("%4.1f ", a[i][j]);

putchar('\n');

}

}

void main()

{

float grade[11][6],s2r;

grade[10][5] = '\0';

srand(time(NULL));

for (i = 0; i <= 9; i++)

for (j = 0; j <= 4; j++)grade[i][j] = rand() % 100 + 1;

avest(grade);

avegr(grade);

maxar(grade);

s2r = avesqr(grade);

printf("最高学生与课程有序数对为(%d,%d)\n平均分方差为%f\nworning:\n每最后一行或最后一列为学生或课程的平均值，右下角0为无效数字请忽略\n", x, y,s2r);

printar(grade);

}

4、整数后移 并不成功 需要改进

#include<stdio.h>

void does(int a[100], int n,int m)

{

int\* p;

int la=a[n-1];

for (p = &a[n-1]; p > a; p--)\*p = \*(p-1);

\*a = la;

m--;

if (m > 0)does(a, n, m);

}

int main()

{

int a[100], m, n, i;

int\* sc = a, \* pr = a;

printf("请输入整数个数\n");

scanf\_s("%d", &n);

printf("请输入后移位数\n");

scanf\_s("%d", &m);

for (i=1; sc <= a + n - 1; i++,sc++)\*sc = i;

for (i = 0; i <= n - 1; i++, pr++)printf("%d ", \*pr);

does(a, m, n);

for (i = 0, pr = a; i <= n - 1; i++, pr++)printf("%d ", \*pr);

}

5、

#include<stdio.h>

struct grade

{

int num;

char name[20];

float score[3];

float avesc;

};

void print(struct grade st[])

{

struct grade \*p;

float \*pi,s=0,\*pm=st[0].score;

int m=0,i;

printf("平均成绩为:\n");

for(p=st;p<st+10;p++)

{

for(pi=(\*p).score;pi<(\*p).score+3;pi++)s+=\*pi;

(\*p).avesc=s/3;

s=0;

printf("%f ",p->avesc);

}

for(i=0;i<=9;i++)if(st[i].avesc>st[m].avesc)m=i;

printf("平均成绩最高的学生为\nnum:%d name:%s score:%f %f %f average:%f\n",st[m].num,st[m].name,st[m].score[0],st[m].score[1],st[m].score[2],st[m].avesc);

}

void main()

{

struct grade \*p;

struct grade student[10];

for(p=student;p<student+10;p++)scanf("%d%s%f%f%f",&p->num,&p->name,&p->score[0],&p->score[1],&p->score[2]);

print(student);

}

6、查找数组素数

#include<stdio.h>

#include<stdlib.h>

#include<time.h>

#include<math.h>

int q(int n)

{

int i, k;

k = sqrt(n);

for (i = 2; i <= k; i++)if (n % i == 0)break;

if (i <= k)return 0;

else return 1;

}

int main()

{

int a[10],i,t;

srand(time(0));

for (i = 0; i <= 9; i++)

{

a[i] = rand() % 100 + 2;

printf("%d ", a[i]);

}

printf("\n数组内的素数为\n");

for (i = 0; i < 9; i++)

{

t = q(a[i]);

if (t)printf("%d ", a[i]);

}

}

7、矩阵乘法

#include<stdio.h>

#include<stdlib.h>

#include<time.h>

int main()

{

int a[2][3] = { 0 }, b[3][2] = { 0 }, c[2][2] = { 0 }, i, j, k;

srand(time(NULL));

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

for (j = 0; j <= 2; j++)

{

a[i][j] = rand();

b[j][i] = rand();

}

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

for (j = 0; j <= 1; j++)

for (k = 0; k <= 2; k++)c[i][j] += a[i][k] \* b[k][j];

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

{

for (j = 0; j <= 1; j++)

printf("%d ", c[i][j]);

putchar('\n');

}

}

1. 一元二次方程

#include<stdio.h>

#include<math.h>

int main()

{

int a,b,c;

scanf("%d%d%d",&a,&b,&c);

if(a<0) {

a=-a;

b=-b;

c=-c;

}

if(b\*b-4\*a\*c>0)

ax(a,b,c);

if(b\*b-4\*a\*c==0)

bx(a,b,c);

if(b\*b-4\*a\*c<0)

cx(a,b,c);

return 0;

}

void ax(int a,int b,int c) {

double x0,x1,x2;

x0 = sqrt(b\*b-4\*a\*c);

x1=(-b+x0)/(2.0\*a);

x2=(-b-x0)/(2.0\*a);

printf("x1=%.3f x2=%.3f\n",x1,x2);

}

void bx(int a,int b,int c) {

double x = -b/(2.0\*a);

printf("x1=%.3f x2=%.3f\n",x,x);

}

void cx(int a,int b,int c) {

double m = -b/(2.0\*a);

double n=sqrt(-(b\*b-4\*a\*c))/(2.0\*a);

printf("x1=%.3lf+%.3lfi x2=%.3lf-%.3lfi\n",m,n,m,n);

}

9、行列式计算

int i,j,k;

int matrix\_2\_3[2][3];

int matrix\_3\_2[3][2];

int result[2][2];

printf("请为第一个矩阵赋值：\n");

for(i=0; i<2; i++)

{

for(j=0; j<3; j++)

{

scanf("%d",&matrix\_2\_3[i][j]);

}

}

printf("请为第二个矩阵赋值：\n");

for(i=0; i<3; i++)

{

for(j=0; j<2; j++)

{

scanf("%d",&matrix\_3\_2[i][j]);

}

}

//三层循环计算两矩阵相乘

for(i=0; i<2; i++)

{

for(j=0; j<2; j++) //注意条件，这段代码块是为result数组计算并赋值

{

int temp = 0;

for(k=0; k<3; k++)

{

temp += matrix\_2\_3[i][k] \* matrix\_3\_2[k][j];

}

result[i][j] = temp;

}

}

printf("两矩阵相乘后：\n");

for(i=0; i<2; i++)

{

for(j=0; j<2; j++)

{

printf("%d ",result[i][j]);

}

printf("\n");

}