第三章 顺序

知识点：printf 常用格式字符串

格式字符形式： %[标志][输出最小宽度][.精度][长度]类型

1. 循环

知识点：while,do while,for会用就行；break和continue的用法；

switch……case(有break才能跳出)

int main()

{

int i;

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

{

switch(i)

{

case 0:printf("%d",i);

case 2:printf("%d",i);

default:printf("%d",i);

}

}

return 0;

} 输出为000122

1. 计算x的值使1+4+7+10+13+…+x的和小于200，求出x的值。

int main()

{

int i=1,sum=0;

do

{

sum+=i;//本来此时算出来的i是刚好超过300的

i+=3;

printf("sum=%d\n",sum);

}while(sum<200);

printf("%d\n",i-6);

return 0;

}

2.求1～100之间的全部素数。

#include<stdio.h>

#include<math.h>

int main()

{

int x,cnt=0,i,j;

scanf("%d",&x);

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

{

for(j=2;j<=sqrt(i);j++)

if(i%j==0) break;

if(j>sqrt(i)) {printf("%4d",i);cnt++;}

if(cnt%10==0) printf("\n",i);

}

printf("\n%d\n",cnt);

return 0;

}

3.字母转换为其后第6个字母，非字母不变。

int main()

{

char c;

while((c=getchar())!='\n')

{

if((c>='a'&&c<='z')||(c>='A'&&c<='Z'))

{

c=c+6;

if(c>'z'||c>'Z') c=c-26;

}

printf("%c",c);

}

return 0;

}

4.小猴吃桃问题。有一天小猴摘下了若干个桃子，当即吃掉一半，还不过瘾，又多吃了一个；第二天接着吃了剩下的桃子的一半后又多吃一个；以后每天都吃尚存桃子的一半零一个，到第10天早上要吃时只剩下一个了。问小猴第一天共摘下了多少个挑子？

int main()

{

int sum=1,i;

for(i=1;i<10;i++)

{

sum=(sum+1)\*2;

}

printf("%d\n",sum);

return 0;

}

1. 打印三角形

int main()

{

int i,j,k;

for(i=0;i<8;i++)

{

for(j=7;j>i;j--)

{

printf(" ");

}

for(k=0;k<=i\*2;k++)

{

printf("\*");

}

printf("\n");

}

return 0;

}

1. 公式求pi的近似值

#include <stdio.h>

#include <math.h>

main()

{

int sign=1,count=0; //sign 表示当前项的符号，count 表示循环次数

double pi=0,n=1.0,term=1.0; //pi 表示多项式的值，term 表示当前项的值

while(fabs(term)>=1e-6)

{

count++;

pi=pi+sign\*term;

n+=2;

sign=-sign;

term=1.0/n;

}

pi=pi\*4;

printf("pi=%10.6f\n",pi);

printf("count=%d\n",count);

}

1. 求最大公约数和最小公倍数

int main() /\* 辗转相除法求最大公约数 \*/

{

int m, n, a, b, t, c;

printf("Input two integer numbers:\n");

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

m=a; n=b;

while(b!=0) /\* 余数不为0，继续相除，直到余数为0 \*/

{ c=a%b; a=b; b=c;}

printf("The largest common divisor:%d\n", a);

printf("The least common multiple:%d\n", m\*n/a);

}

1. 数组

知识点：二维数组赋值方式，字符数组的结束方式

1. 任意输入10个数到数组a，然后找出其中最大数、最小值及他们的位置，并输出。

int main()

{

int a[10],i,j,min,max,imax=0,imin=0;

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

{

scanf("%d",&a[i]);

}

max=a[0];

min=a[0];

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

{

if(a[i]>max) {max=a[i];imax=i;}

if(a[i]<min) {min=a[i];imin=i;}

}

printf("max=a[%d]=%d,min=a[%d]=%d\n",imax,max,imin,min);

return 0;

}

1. 从小到大排序

（1）冒泡法（相邻两两对比，像泡泡一样浮上来）

#define N 10

int main()

{

int a[10],i,j,t;

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

scanf("%d",&a[i]);

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

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

{

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

{

t=a[i];

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

a[i+1]=t;

}

}

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

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

return 0;

}

1. 数据选择法

#define N 10

int main()

{

int a[10],i,j,t;

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

scanf("%d",&a[i]);

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

for(j=i+1;j<N;j++)

{

if(a[i]>a[j])

{

t=a[i];

a[i]=a[j];

a[j]=t;

}

}

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

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

return 0;

}

1. 向一个有序数据系列中插入数据，并保持数据的有序性。

for(i=0;i<m;i++) /\* 确定插入位置 i \*/

if(k<n[i])break;

for(j=m;j>i;j--) /\*后移数据，空出插入位置 n[i]\*/

n[j]=n[j-1];

n[i]=k; /\*插入k到n[i] \*/

其他和排序原理相同

删除则后移数据

1. 矩阵转置

int main()

{

int i,j,a[3][4]={1,2,3,4,5,6,7,8,9,10,11,12},b[4][3];

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

{

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

{

printf("%5d",a[i][j]);

b[j][i]=a[i][j];

}

printf("\n");

}

printf("\n");

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

{

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

{

printf("%5d",b[i][j]);

}

printf("\n");

}

return 0;

}

1. 出杨辉三角的前10行。

#include <stdio.h>

int main()

{

int a[10][10],i,j;

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

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

a[i][j]=1;

for(i=2;i<10;i++)

for(j=1;j<=i-1;j++)

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

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

{

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

printf("%4d",a[i][j]);

printf("\n");

}

return 0;

}

6.字符串数组的寻址方式（注意’\0’结束和’\n’换行）

#include <string.h>

int main()

{

char c[]="BASIC\ndBASE";

printf("%s\n",c);

printf("%s\n",&c[2]);

return 0;

}

7.字符串从长到短输出

#include<string.h>

int main()

{

char c[5][50],t[50];

int i,j,l[2][5],s;

for(i=0;i<5;i++)

{

printf("Please input string No.%d: ",i+1);

gets(c[i]);

}

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

{

s=i;

for(j=i+1;j<=4;j++)

if(strlen(c[s])>strlen(c[j])) s=j;

strcpy(t,c[i]);

strcpy(c[i],c[s]);

strcpy(c[s],t);

}

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

puts(c[i]);

printf("\n");

return 0;

}