**程序示例**

**1. 求和运算：如等差、等比数列求和，奇数、偶数求和，加减规律组项(-…+…)求和，a+aa+aaa+… (n个a)，以及对特征数据（如质因数、倍数、素数等）求和，数据的来源或是键盘输入、或是限定在某范围**

**求累加和：**

**#include <iostream>**

**using namespace std;**

**void main( )**

**{**

**int sum=0,i=1,n; //定义变量**

**cout<<"input n:";**

**cin>>n; //从键盘接收n**

**while(i<=n)**

**{**

**sum=sum+i;**

**i=i+1;**

**}**

**cout<<"sum="<<sum<<endl;**

**}**

**求n! （迭代方法）**

**#include <iostream>**

**using namespace std;**

**void main( )**

**{**

**int n,t=1,i=1;**

**cout<<"plese input n:"<<endl;**

**cin>>n;**

**if(n>0)**

**{**

**while(i<=n)**

**{ t=t\*i; i=i+1; }**

**cout<<"N!="<<t<<endl;**

**}**

**else**

**if(n==0) cout<<"N!=1"<<endl;**

**else cout<<"input error."<<endl;**

**}**

**求n! （递归方法）**

**#include <iostream>**

**using namespace std;**

**long fact(int n)**

**{**

**long f;**

**if (n<0)**

**cout<<"n<0,data error!"<<endl;**

**else if (n==0) f=1;**

**else f=fact(n-1)\*n; //函数递归调用**

**return(f);**

**}**

**void main()**

**{**

**int n;**

**long y;**

**cout<<"Enter a positive integer:";**

**cin>>n;**

**y=fact(n); // 函数调用**

**cout<<n<<"!="<<y<<endl;**

**}**

**自定义子函数功能：实现计算n!；主函数的功能：键盘输入n和m（n>0,m>0, m > n），计算p=m!/(m-n)!，并输出p。**

1. **迭代方法**

**#include <iostream>**

**using namespace std;**

**int fac(int n)**

**{**

**int f,i=1;**

**f=1;**

**while(i<=n)**

**{**

**f=f\*i;**

**i=i+1;**

**}**

**return f;**

**}**

**void main()**

**{**

**int n,m;**

**int y;**

**cout<<"Enter the positive integer m and n(m>n):";**

**cin>>m>>n;**

**y=fact(m)/fact(m-n); // 函数调用**

**cout<<n<<"!="<<y<<endl;**

**}**

**2.递归方法**

**#include <iostream>**

**using namespace std;**

**int fact(int n)**

**{**

**int f;**

**if (n<0)**

**cout<<"n<0,data error!"<<endl;**

**else if (n==0) f=1;**

**else f=fact(n-1)\*n; //函数递归调用**

**return(f);**

**}**

**void main()**

**{**

**int n,m;**

**int y;**

**cout<<"Enter the positive integer m and n(m>n):";**

**cin>>m>>n;**

**y=fact(m)/fact(m-n); // 函数调用**

**cout<<n<<"!="<<y<<endl;**

**}**

**求x的n次幂**

* **方法一：**

**#include <iostream>**

**using namespace std;**

**void main( )**

**{**

**double power (double x, int n);**

**double x,y;**

**int n;**

**cout<<"Input the x and n:";**

**cin>>x>>n;**

**y= power (x, n);**

**cout << endl<<x<<" to the power "<<n<<" is "<< y<< endl;**

**}**

**double power (double x, int n)**

**{**

**double val = 1.0;**

**while (n!=0)**

**{**

**val = val\*x;**

**n--;**

**}**

**for (int i=1;i<=n;i++)**

**val=val\*x;**

**while (n--)**

**val = val\*x;**

**return(val);**

**}**

* **方法二：递归函数**

**#include <iostream>**

**using namespace std;**

**double power(double x,int n)**

**{**

**double f=0;**

**if (x==0)**

**cout<<"0的任何次幂均为!"<<endl;**

**else if (n==0) f=1; //递归出口**

**else f=power(x,n-1)\*x; //函数递归调用**

**return(f);**

**}**

**void main()**

**{**

**int n;**

**double x,y;**

**cout<<"请输入x(基数)和n(幂次)：";**

**cin>>x>>n;**

**y=power(x,n); // 函数调用**

**cout<<x<<"的"<<n<<"次幂="<<y<<endl;**

**}**

**斐波那契数列的计算（迭代法、递归法）**

* **迭代法：**

**#include <iostream>**

**#include <iomanip>**

**using namespace std;**

**void main( )**

**{**

**int i,f[20]={1,1};**

**for(i=2;i<20;i++)**

**f[i]=f[i-2]+f[i-1];**

**for(i=0;i<20;i++)**

**{**

**if(i%5==0) cout<<endl;**

**cout<<setw(5)<<f[i];**

**}**

**}**

**递归法：**

**#include <iostream>**

**using namespace std;**

**int fibonacci(int n)**

**{**

**int fibo;**

**if(n==1||n==2)**

**fibo=1;**

**else**

**fibo=fibonacci(n-1)+fibonacci(n-2); //函数递归调用**

**return fibo;**

**}**

**void main( )**

**{**

**int n,fibon;**

**cout<<"Please input a positive integer:";**

**cin>>n;**

**fibon=fibonacci(n); //函数调用**

**cout<<"The "<<n<<" number of fibonacci is:"<<fibon;**

**}**

**求一个数的所有因子**

**分析：求一个整数n的所有因子可以采用穷举法，对1到n的全部整数进行判断，凡是能够整除n的均为n的因子。**

**#include <iostream>**

**using namespace std;**

**void main ()**

**{**

**int n,i;**

**cout<<"Enter a positive integer :" ;**

**cin >> n ;**

**cout<<“Number ”<<n<< “Factors” ; //输出时的提示信息**

**for (i=1 ; i<= n ; i++ )**

**if ( n % i ==0 )**

**cout << i << " " ;**

**cout << endl ;**

**}**

**判断一个数是否为素数**

**#include <iostream>**

**#include<cmath>**

**using namespace std;**

**void main ()**

**{**

**int n,i;**

**cout<<"Enter a positive integer :" ;**

**cin>> n ;**

**for ( i=2; i<=sqrt((double)n);i++)**

**{**

**if(n%i==0) break;**

**}**

**if(i<=sqrt((double)n))**

**cout<<“不是素数“<<endl;**

**else**

**cout<<“是素数“<<endl;**

**}**

**求100以内所有奇数之和**

**#include<iostream>**

**using namespace std;**

**void main()**

**{**

**int k,s=0;**

**for(k=1;k<=100;k++)**

**{**

**if(k%2==0)**

**continue;**

**s=s+k;**

**}**

**cout<<“s=”<<s<<endl;**

**}**

**求sinx的近似值，要求误差小于0.0001。近似计算公式如下：**



**#include <iostream.h>**

**void main(){**

**double x,sinx,item;**

**int n=1,sign=-1; //保存符号**

**cout<<"input x:";**

**cin>>x;**

**sinx=x; item=x;**

**while(item>0.0001)**

**{**

**item=item\*x\*x/((2\*n)\*(2\*n+1)); //通项**

**sinx=sinx+item\*sign;**

**sign=-sign; //符号的变换**

**n++;**

**}**

**cout<<"sin("<<x<<")="<<sinx<<endl;**

**}**

**求n之内所有质数之和**

**#include <iostream>**

**#include <cmath>**

**using namespace std;**

**void main ()**

**{**

**int n,i=3,j,sum=2;**

**cout<<"请输入质数求和的范围："<<endl;**

**cin>>n;**

**cout<<n<<"之内的所有质数之和为："<<endl<<"sum=2";**

**for(;i<=n;i++)**

**{**

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

**{**

**if(i%j==0)**

**break;**

**}**

**if(j>sqrt(double(i)))**

**{**

**cout<<'+'<<i;**

**sum=sum+i;**

**}**

**}**

**cout<<"="<<sum<<endl;**

**}**

**求完全数（除了本身之外的所有因子之和等于本身的数，如 28）**

**#include <iostream>**

**using namespace std;**

**void main ()**

**{**

**int n,i,sum=1;**

**cout<<"请输入整数n："<<endl;**

**cin>>n;**

**cout<<n<<"的所有因子为:"<<endl;**

**cout<<n<<"=1";**

**for(i=2;i<=n;i++)**

**{**

**if(n%i==0)**

**{**

**cout<<'\*'<<i;**

**sum=sum+i;**

**}**

**}**

**if((sum-n)==n)**

**cout<<endl<<n<<"是一个完全数！"<<endl;**

**else**

**cout<<endl<<n<<"不是一个完全数！"<<endl;**

**}**

**求1000以内的完全数**

**#include <iostream>**

**using namespace std;**

**void main()**

**{ int i,j,s;**

**for( i=1; i<=1000; i++ )**

**{**

**s = 0;**

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

**if ( i % j == 0 ) s = s + j;**

**if ( i == s ) cout << i << endl;**

**}**

**}**

**求回文**

**#include <iostream>**

**using namespace std;**

**void main( )**

**{ int n,m,num=0,r;**

**cout <<"Enter the number:";**

**cin >>n;**

**m=n;**

**while (n!=0)**

**{ r=n%10; //输出数字左移1位**

**num=num\*10+r; // 求逆序数**

**n/=10; /\*从原始数据中去掉已经反序输出的数位，为获**

**取下一个反序输出数字做准备\*/**

**}**

**if(num==m)**

**cout <<"整数"<<m<<"是回文"<<endl;**

**else**

**cout <<"整数"<<m<<"不是回文"<<endl;**

**}**

**将一个偶数分解成两个素数之和**

**#include <iostream>**

**#include <cmath>**

**using namespace std;**

**void main( )**

**{**

**int n,i,j,k,f1,f2;**

**cout<<"请输入需要分解的偶数："<<endl;**

**cin>>n;**

**for(i=3;i<=n/2;i+=2)**

**{**

**f1=1;f2=1; //设置标志变量**

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

**if(i%j==0)**

**{**

**f1=0;**

**break;**

**}**

**if(f1==1)**

**{**

**k=n-i;**

**for(j=2;j<=sqrt((double)k);j++)**

**if(k%j==0)**

**{**

**f2=0;**

**break;**

**}**

**}**

**if(f1==1&&f2==1)**

**cout<<n<<"="<<i<<"+"<<n-i<<endl;**

**}**

**}**

**用函数方法实现：**

**#include <iostream>**

**#include <cmath>**

**using namespace std;**

**//用户自定义函数的定义，判断一个整数是否素数**

**bool prime(int n)**

**{**

**bool f=1;**

**int i;**

**for(i=2;i<=sqrt((double)n);i++)**

**if(n%i==0)**

**{**

**f=0;**

**break;**

**}**

**return f;**

**}**

**void main( )**

**{**

**int n,i;**

**cout<<"请输入需要分解的偶数："<<endl;**

**cin>>n;**

**for(i=3;i<=n;i+=2)**

**if(prime(i)&&prime(n-i)) //函数调用**

**{**

**cout<<n<<"="<<i<<"+"<<n-i<<endl;**

**//若在此处加上break; 则输出唯一的分解式！**

**}**

**}**

**利用辗转相除法（阿基里德算法）求最大公约数和最小公倍数**

**#include <iostream>**

**using namespace std;**

**void main( )**

**{**

**int p, q, r,a,b; // 变量定义,p,q,r,a,b**

**// 提示用户由键盘输入两个正整数**

**cout<< "Please input two integer numbers:"<<endl;**

**cin>>p>>q;**

**a=p;**

**b=q; //保留原来p,q的值，分别放在a,b中**

**//保证p为大数，q为小数**

**if(p<q)**

**{ r=p; p=q; q=r; }//交换p,q**

**// 计算 p 除q 的余数r**

**r=p%q;**

**// 只要r 不等于 0, 重复进行下列计算**

**while(r!=0)**

**{ p=q;**

**q=r;**

**r=p%q;**

**} //辗转法**

**cout <<"两个正整数的最大公因数是"<<q<<endl;**

**cout <<"两个正整数的最小公倍数是"<<a\*b/q<<endl;**

**}**

**用枚举方法（也称为范举的方法）解决问题（如水仙花数）**

**法一：**

**#include <iostream>**

**using namespace std;**

**void main()**

**{ //穷举法的思想，列举各个数位可能的数字**

**for(int x=1; x<=9; x++)**

**for(int y=0; y<=9;y++)**

**for(int z=0; z<=9; z++)**

**if(x\*x\*x+y\*y\*y+z\*z\*z== 100\*x+10\*y+z)**

**cout <<(100\*x+10\*y+z) <<"是水仙花数.\n";**

**}**

**法二：**

**#include <iostream>**

**using namespace std;**

**void main()**

**{**

**int i,a,b,c;**

**for(i=101; i<999; i++)**

**{**

**a=i%10;**

**b=(i/10)%10;**

**c=i/100; //取出输入数据各个数位的数字**

**if(a\*a\*a+b\*b\*b+c\*c\*c==i) //判断条件即为水仙花数的定义！**

**cout <<i<<"是水仙花数。\n";**

**}**

**}**

**编程求a+aa+aaa+aaaa+……+aa…a(n个), 其中a和n为1~9之间的整数。**

**#include<iostream>**

**using namespace std;**

**void main( )**

**{ int a,n,sum=0,num=0,i=1;**

**cout<<"Please input two integer(1~9):"<<endl;**

**cin>>a>>n;**

**while( i<=n )**

**{**

**num= num\*10+a ;**

**sum=sum+num;**

**i++;**

**}**

**cout<<sum<<endl;**

**}**

**“数据---各位数字”的获取、重组及转存**

**已知int a=3467,写出计算其个、十、百、千位的表达式。**

**解：a%10, a/10%10, a/100%10，a/1000%10或a/1000**

* **将键盘输入的整数反序输出**

**#include <iostream>**

**using namespace std;**

**void main( )**

**{**

**int n,num;**

**cout <<"Enter the number:";**

**cin >>n;**

**cout <<"The number in reverse order is: ";**

**do**

**{**

**num = n % 10;**

**cout << num;**

**n /= 10;**

**} while (n!=0);**

**cout <<endl;**

**}**

* **求一个整数的反序数**

**#include <iostream>**

**using namespace std;**

**void main( )**

**{ int n,num=0;**

**cout <<"Enter the number:";**

**cin >>n;**

**while (n!=0)**

**{**

**r=n%10; //输出数字左移1位**

**num=num\*10+r; // 求逆序数**

**n/=10; /\*从原始数据中去掉已经反序输出的数位，为获**

**取下一个反序输出数字做准备\*/**

**}**

**cout <<"The number in reverse order is: "<<num<<endl;**

**}**

**十进制转换为八进制**

**#include <iostream>**

**#include <cmath>**

**using namespace std;**

**void main( ) {**

**int a,j; //a为十进制整数，j为a除以8的余数**

**double i=0.0; //用于记录得到的余数是第几位，最低位为第0位**

**double b=0.0; //b为转换后的八进制整数**

**cout<<"请输入一个十进制整数：";**

**cin>>a;**

**do**

**{**

**j=a%8; //转换成那种进制就除以对应进制的基数**

**b=b+j\*pow(10.0,i); /\*最先得到的余数是转换后的八进制数的最低位,乘以位权后存放到b中 \*/**

**a=a/8; // a除8的商，为下一次“除基数取余”做准备**

**i=i+1;**

**}while(a>0); // 循环，直到商为0为止**

**cout<<fixed;**

**cout.precision(0);**

**cout<<"转换成八进制数是： "<<b<<endl;**

**}**

**八进制转换为十进制**

**#include <iostream>**

**#include <cmath>**

**using namespace std;**

**void main( ) {**

**int a,j; //a为八进制整数，j为a除以10的余数**

**double i=0.0; //用于记录得到的余数是第几位，最低位为第0位**

**double b=0.0; //b为转换后的十进制整数**

**cout<<"请输入一个八进制整数：";**

**cin>>a;**

**do**

**{**

**j=a%10; //取八进制数的各个位上的数字，然后用位权法进行转换 b=b+j\*pow(8.0,i); /\*最先得到的余数是转换后的八进制数的最低位,乘以位权后存放到b中 \*/**

**a=a/10; // 取八进制数下一位上的数字（从右往左取）**

**i=i+1;**

**}while(a>0); // 循环，直到商为0为止**

**cout<<fixed;**

**cout.precision(0);**

**cout<<"转换成十进制数是： "<<b<<endl;**

**}**

**3. 数组元素的基本操作：删除和插入数组元素，对称和反序操作查找特征元素（如：主对角线---自左至右对角线、次对角线元素---自右至左对角线、最大值、最小值、数组平均值、鞍点—其值所在行最大且所在列最小时等）、记录特征位置、输出（或交换、或删除、或插入等）特征元素**

* **一维数组求平均值**

**#include <iostream>**

**using namespace std;**

**void main( )**

**{**

**const int N=10;**

**int scores[N];**

**int n,i,total=0; // n为数组实际长度**

**float avg=0.0;**

**cout<<“请输入数组长度：”;**

**cin>>n; // 输入数组实际长度**

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

**cin>>scores[i];**

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

**total=total+scores[i];**

**avg=float(total)/n; //计算平均成绩**

**cout<<“Average:”<<avg<<endl;**

**}**

* **求一维数组中的最大值及其位置**

**#include <iostream>**

**using namespace std;**

**void main( )**

**{**

**const int N=10;**

**int dollars[N]; //声明数组**

**int n,i,j,high;**

**cout<<“请输入数组实际长度：”;**

**cin>>n;**

**for(i=0;i<n;i++) cin>>dollars[i];**

**high=dollars[0]; //默认第一个元素为最大值**

**j=0; //记录最大元素的下标**

**for(i=1;i<n;i++) //从第二个元素开始比较**

**{**

**if(dollars[i]>high)**

**{**

**high=dollars[i]; //记录最大值**

**j=i; //记录最大元素的下标**

**}**

**}**

**cout<<“最大元素为第”<<j+1<<“个元素，它的值为：”<< high <<endl;**

**// 最大值或者用下标表示dollars[j];**

**}**

* **数组的反序（逆序）输出**

**法一：普通数组  
#include <iostream>**

**#include <iomanip>**

**using namespace std;**

**void main()**

**{**

**const int N=10;**

**int a[N],i,j,n,temp;**

**cout<<"请输入数组的实际长度："<<endl;**

**cin>>n;**

**cout<<"请输入数组元素："<<endl;**

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

**cin>>a[i];**

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

**{**

**temp=a[i]; a[i]=a[j]; a[j]=temp; //实现反序存储**

**}**

**cout<<"反序之后的数组元素为："<<endl;**

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

**cout<<setw(4)<<a[i];**

**}**

**法二：指针方法**

**#include <iostream>**

**#include <iomanip>**

**using namespace std;**

**void main()**

**{**

**const int N=10;**

**int a[N],i,j,n,temp, \*p;**

**cout<<"请输入数组的实际长度："<<endl;**

**cin>>n;**

**cout<<"请输入数组元素："<<endl;**

**for(p=a;p<a+n;p++) cin>>\*p;**

**p=a;**

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

**{**

**temp=\*(p+i); \*(p+i)=\*(p+j); \*(p+j)=temp; //实现反序存储**

**}**

**cout<<"反序之后的数组元素为："<<endl;**

**for(p=a;p<a+n;p++)**

**cout<<setw(4)<<\*p;**

**}**

* **判断数组中的元素是否对称**

**#include <iostream>**

**using namespace std;**

**void main()**

**{**

**const int N=10;**

**int a[N],i,j,n,f=1;**

**cout<<"请输入数组的实际长度："<<endl;**

**cin>>n;**

**cout<<"请输入数组元素："<<endl;**

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

**cin>>a[i];**

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

**{**

**if(a[i]!=a[j]) // 判断是否对称**

**{**

**f=0;**

**break;**

**}**

**}**

**if(f==1)**

**cout<<"数组元素对称"<<endl;**

**else**

**cout<<"数组元素不对称"<<endl;**

**}**

* **把一个数列中的所有相同的数删到只剩一个。**

**#include<iostream>**

**using namespace std;**

**void main( )**

**{ int a[10],i,j,pos=1;**

**cout<<"Please input ten integer:"<<endl;**

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

**cin>>a[i];**

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

**{**

**for(j=0; j<pos ;j++)**

**if(a[i]= =a[j])**

**break;**

**if( j>=pos )**

**{**

**a[pos]=a[i];**

**pos++;**

**}**

**}**

**cout<<"The number after delete is:"<<endl;**

**for(i=0; i<pos ;i++)**

**cout<<a[i]<<endl;**

**}**

* **插入算法**

**法一：普通数组**

**#include <iostream>**

**#include <iomanip>**

**using namespace std;**

**void main()**

**{**

**const int N=10;**

**int a[N+1],p,x,i;**

**cout<<“输入a数组：”<<endl;**

**for(i=0;i<N;i++) cin>>a[i];**

**cout<<"输入待插入的数x："<<endl;**

**cin>>x;**

**p=0;**

**while(x>a[p]&&p<n) p++; //找到x应插入的正确位置**

**for(i=n-1;i>=p;i--)**

**a[i+1]=a[i]; //将a[p]~a[n-1]后移**

**a[p]=x; //x插入正确位置**

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

**cout<<setw(3)<<a[i]<<endl;**

**}**

**法二：指针**

**#include <iostream>**

**#include <iomanip>**

**using namespace std;**

**void main(void)**

**{**

**const int N=10;**

**int a[N+1],p,x, \*t ;**

**cout<<“输入a数组：”<<endl;**

**for(t=a;t<a+N;t++)**

**cin>>\*t;**

**cout<<“输入待插入的数x：”<<endl;**

**cin>>x;**

**t=a; p=0;**

**while (x>\*(t+p)&&p<N)**

**p++;**

**for (t=a+N-1; t>=a+p; t--)**

**\*(t+1)=\*t; // ②**

**for (i=N-1; i>=p; i--)**

**\*(t+i+1)=\*(t+i); // ①**

**t=a;**

**t=a+p; //将指针t指向由p记录的插入位置**

**\*t=x; // ②**

**\*(t+p)=x; // ①**

**for(t=a;t<a+N+1;t++)**

**cin>>\*t;**

**}**

* **求N×N方阵中，主对角线和次对角线上所有元素之和**

**#include <iostream>**

**using namespace std;**

**void main()**

**{**

**const int N=10;**

**int a[N][N];**

**int i,j,n;**

**int sum = 0;**

**cout<<"input num:"<<endl;**

**cin>>n;**

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

**for(j=0;j<n;j++)**

**{**

**cin>>a[i][j];**

**}**

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

**{ sum+=a[i][i]+ a[i][n-i-1]; }**

**cout<<sum<<endl;**

**}**

* **由随机数产生器产生一维数组的元素，求最大值及其所在数组中的位置，求最小值及其所在数组中的位置，并把最大最小值进行交换。**

**#include<iostream>**

**#include<cstdlib>**

**#include<ctime>**

**using namespace std;**

**void main()**

**{**

**const int N=20;**

**int a[N],n,i,j,k,t,max,min;**

**cout<<"请输入n值："<<endl;**

**cin>>n;**

**srand(time(NULL));**

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

**{**

**a[i]=1+rand()%(200-1+1);**

**}**

**cout<<"产生的随机数为："<<endl;**

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

**cout<<" "<<a[i];**

**min=a[0];**

**for(i=1;i<n;i++)**

**{**

**if(a[i]<min)**

**{**

**min=a[i];**

**j=i;**

**}**

**}**

**cout<<endl;**

**cout<<"最小数为:"<<min<<" 最小值是数组的第"<<j+1<<"个元素。"<<endl;**

**max=a[0];**

**for(i=1;i<n;i++)**

**{**

**if(a[i]>max)**

**{**

**max=a[i];**

**k=i;**

**}**

**}**

**cout<<"最大数为:"<<max<<" 最大值是数组的第"<<k+1<<"个元素。"<<endl;**

**t=a[j];a[j]=a[k];a[k]=t;**

**cout<<"最大、最小值交换后的结果为："<<endl;**

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

**cout<<" "<<a[i];**

**}**

* **求一维数组的平均值并且把小于平均值的那些元素删掉。**

**#include <iostream>**

**using namespace std;**

**void main()**

**{**

**const int N=20;**

**int a[N];**

**int i,j,n;**

**double ave=0,sum=0;**

**cout<<"input the n:";**

**cin>>n;**

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

**{**

**cin>>a[i];**

**if(a[i]==0) break;**

**}**

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

**{**

**cout<<a[i]<<" ";**

**}**

**cout<<endl;**

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

**{**

**sum=sum+a[i];**

**}**

**ave=sum/n;**

**cout<<"the ave is:"<<ave<<endl;**

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

**{**

**if(a[i]<ave)**

**{**

**for(j=i;j<n;j++)**

**{**

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

**} //删除小于平均值的数组元素**

**n--; //删除小于平均值的元素后，数组的实际长度**

**i--;**

**}**

**}**

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

**{**

**cout<<a[i]<<" ";**

**}**

**}**

* **矩阵的转置**

**#include <iostream>**

**#include <iomanip>**

**using namespace std;**

**void main(void)  
{**

**const int N=2,M=3;**

**float a[N][M],b[M][N]；  
  int i,j；  
cout<<“Input array a”<<endl；  
  for(i=0；i<N；i++)  
    for(j=0；j<M；j++)  
      cin>>a[i][j]；**

**for(i=0;i<M;i++)  
   {**

**for(j=0;j<N;j++)  
        cout<<setw(4)<<b[i][j]；//输出数组b，并进行格式控制  
      cout<<endl； //输完一行后，换行再继续输出下一行**

**}**

**}**

* **二维数组求马鞍点（其值所在行最大且所在列最小）**

**#include <iostream>**

**#include <iomanip>**

**using namespace std;**

**void main()**

**{**

**const int N=100;**

**int i,j,k,m,n,c,flag,min;**

**int a[N][N];**

**flag=0;**

**cout<<"请输入矩阵的行数: "; /\*输入矩阵的行数\*/**

**cin>>m;**

**cout<<"请输入矩阵的列数: "; /\*输入矩阵的列数\*/**

**cin>>n;**

**cout<<"请输入矩阵:"<<endl;**

**/\*输入矩阵，元素间以空格隔开，回车换行\*/**

**for(i=0;i<m;i++)**

**for(j=0;j<n;j++)**

**{**

**cin>>a[i][j];**

**}**

**cout<<"您输入的矩阵是:"<<endl;**

**for(i=0;i<m;i++) /\*显示当前输入的矩阵\*/**

**{**

**for(j=0;j<n;j++)**

**cout<<setw(6)<<a[i][j];**

**cout<<endl;**

**}**

**for(i=0;i<m;i++)**

**{**

**min=a[i][0];**

**c=0; //记录一行中最小值最在的列数**

**for(j=0;j<n;j++) /\*找出i行中的最小值\*/**

**if(a[i][j]<min)**

**{**

**min=a[i][j];**

**c=j;**

**}**

**for(j=0;j<n;j++)**

**{ if(a[i][j]==min) /\*找出i行中所有最小值\*/**

**{**

**for(k=0;k<m;k++) /\*对所有min进行列比较\*/**

**if(a[k][j]>min) break;**

**if(k==m) /\*min为马鞍点\*/**

**{ c=j;**

**cout<<"该矩阵的马鞍点是:"<<a[i][c]<<endl;**

**flag=1;**

**}**

**}**

**}**

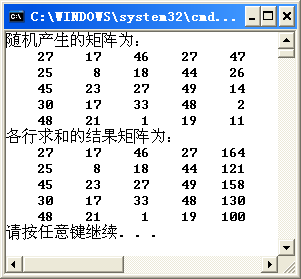
**}**

**if(flag==0)**

**cout<<"该矩阵没有马鞍点!"<<endl;**

**}**

* **一个二维矩阵，元素随机产生，将每行元素求和后放在最后一列保存**



**#include <iostream>**

**#include <cstdlib>**

**#include <ctime>**

**#include <iomanip>**

**using namespace std;**

**void main()**

**{**

**const int N=5;**

**int matrix[N][N];**

**int i,j,sum;**

**srand(time(NULL));**

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

**for(j=0;j<N;j++)**

**matrix[i][j]=1+rand()%(50-1+1); //随机产生元素**

**cout<<"随机产生的矩阵为："<<endl;**

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

**{**

**for(j=0;j<N;j++)**

**cout<<setw(6)<<matrix[i][j];**

**cout<<endl;**

**}**

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

**{**

**sum=0; //每行均需求和，故每次求和前对累加变量初始化**

**for(j=0;j<N;j++)**

**sum=sum+matrix[i][j];**

**matrix[i][N-1]=sum; //对每行元素求和，之后存于改行最后一个元素位置**

**}**

**cout<<"各行求和的结果矩阵为："<<endl;**

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

**{**

**for(j=0;j<N;j++)**

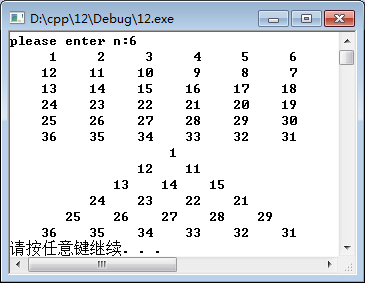
**cout<<setw(6)<<matrix[i][j];**

**cout<<endl;**

**}**

**}**

* **按规律自动产生一维或二维 N\*N 数据（如蛇形规律数据）**

****

**#include<iostream>**

**#include<iomanip>**

**using namespace std;**

**void main()**

**{**

**const int t=6;**

**int i,j,k=1,count;**

**int a[t][t];**

**for(i=0;i<t;i++)**

**{**

**if(i%2==0)**

**{**

**count=i\*t+1;**

**for(j=0;j<t;j++)**

**a[i][j]=count++;**

**}**

**else**

**{**

**count=(i+1)\*t;**

**for(j=0;j<t;j++)**

**a[i][j]=count--;**

**}**

**}**

**for(i=0;i<t;i++)**

**{**

**for(j=0;j<t;j++)**

**cout<<setw(4)<<a[i][j];**

**cout<<endl;**

**}**

**for(i=0;i<t;i++)**

**{**

**for(k=0;k<3\*(t-i);k++)**

**cout<<' ';**

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

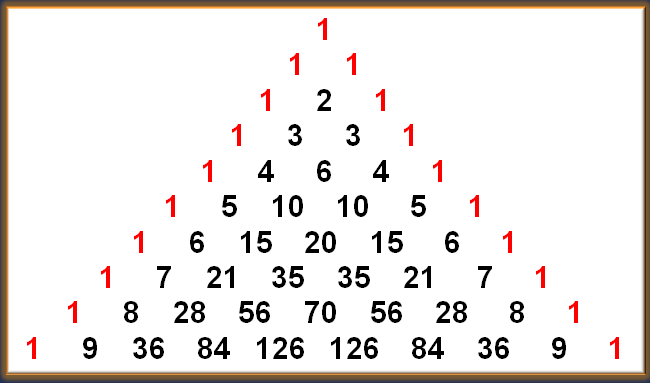
**cout<<setw(6)<<a[i][j];**

**cout<<endl;**

**}**

**}**

* **按照指定图形或格式输出指定区域数据。**

****

**观察输出结果，可用发现规律：**

**共输出10行数据，第10行的“1”前面有0个空格；第9行的“1”前面有3个空格；第8行的“1”前面有6个空格；第7行的“1”前面有9个空格…… 第1行的“1”前面有24个空格。即每行先输出3\*(n-1-i)个空格后再输出杨辉三角数。（其中n为输出的行数，本例n取10，i为行的循环控制变量，i从0取到9）**

**#include <iostream>**

**#include <iomanip>**

**using namespace std;**

**void main()**

**{**

**const int N=20;**

**int a[N][N];**

**int n;**

**cout<<"请输入杨辉三角需要输出的层数："<<endl;**

**cin>>n;**

**a[0][0]=1;a[1][0]=1;a[1][1]=1;**

**int i,j;**

**for (i=2;i<n;i++)**

**{**

**a[i][0]=1;**

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

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

**a[i][i]=1;**

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

**{for(k=0;k<=18-(2\*i);k++)**

**cout<<" ";**

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

**cout<<setw(4)<<a[i][j];**

**}**

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

**{**

**for (j=1;j<=(3\*(n-1-i));j++)**

**cout<<' ';**

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

**cout<<setw(6)<<a[i][j];**

**cout<<endl;**

**}**

**}**

**求交集和并集**

* **求两个数组的并集（有序）**

**#include<iostream>**

**using namespace std;**

**void main( )**

**{**

**const int N1=5;**

**const int N2= 5;**

**int a[N1] ={1,2,3,5,6},b[N2]={4,7,8,9,10};**

**int c[N1+N2];**

**int pa=0,pb=0,pc=0,end;**

**while(pa<N1 && pb<N2)**

**{**

**if(a[pa]<b[pb])**

**{**

**c[pc]=a[pa];**

**pc++;**

**pa++;**

**}**

**else**

**{**

**c[pc]=b[pb];**

**pc++;**

**pb++;**

**}**

**}**

**if(pa==N1)**

**{**

**end=pb;**

**for(;pc<N1+N2;pc++,end++)**

**c[pc]=b[end];**

**}**

**else**

**{**

**end=pa;**

**for(;pc<N1+N2;pc++,end++)**

**c[pc]=a[end];**

**}**

**for(pc=0;pc<N1+N2;pc++)**

**cout<<c[pc];**

**}**

* **求两个数组的并集（无序）**

**思想（以数组a为基础，用数组b的元素分别与数组a的每个元素比较，得出b中不同的元素，然后写入数组a的末端）**

**#include <iostream>**

**#include <iomanip>**

**using namespace std;**

**void main()**

**{**

**const int M=10,N=10;**

**int a[M],b[N];**

**int m,n,i=0,j=0,f=0,k,flag;**

**cout<<"输入数组a中元素的个数："<<endl;**

**cin>>m;**

**cout<<"输入数组a的"<<m<<"个元素(数组元素无序)："<<endl;**

**for(i=0;i<m;i++)**

**cin>>a[i];**

**cout<<"输入数组b中元素的个数："<<endl;**

**cin>>n;**

**cout<<"输入数组b的"<<n<<"个元素(数组元素无序)："<<endl;**

**for(j=0;j<n;j++)**

**cin>>b[j];**

**//以数组a为基础，寻找b中不同于a的那些元素，然后写入a数组的末端。**

**k=m; //指向a数组的末端**

**for(j=0;j<n;j++)**

**{**

**flag=0; //设置标志，如果b中有于a相同的元素，则flag=1；**

**for(i=0;i<m;i++)**

**if(a[i]==b[j])**

**{**

**flag=1; //b中若有与a中元素相同的，则结束本次处理，继续判断b的下一个元素**

**break;**

**}**

**if(flag==0) //如果b的元素与a中所有元素均不相同，则写入a数组的末端**

**{**

**a[k++]=b[j];**

**f++; //a数组实际长度增加1**

**}**

**}**

**cout<<"并集的各个元素依次为:"<<endl;**

**for(i=0;i<m+f;i++)**

**cout<<setw(4)<<a[i];**

**}**

* **用指针方法合并两个数组的数据（两个数组无序）**

**方法一：普通数组**

**思想（把数组a全部付给数组c，再用数组a的每个元素分别和数组b元素比较，得出b中不同的元素全部赋给c）**

**#include <iostream>**

**#include <iomanip>**

**using namespace std;**

**void main()**

**{**

**const int M=30,N=20;**

**int a[M],b[N],c[M+N];**

**int d,e,f=0,\*pa,\*pb,\*pc;**

**cout<<"输入数组a中元素的个数："<<endl;**

**cin>>d;**

**cout<<"输入数组a的"<<d<<"个元素："<<endl;**

**for(pa=a;pa<a+d;pa++)**

**cin>>\*pa;**

**cout<<"输入数组b中元素的个数："<<endl;**

**cin>>e;**

**cout<<"输入数组b的"<<e<<"个元素："<<endl;**

**for(pb=b;pb<b+e;pb++)**

**cin>>\*pb;**

**for(pa=a,pc=c;pa<a+d;pa++,pc++)**

**\*pc=\*pa;**

**for(pb=b;pb<b+e;pb++)**

**{**

**for(pa=a;pa<a+d;pa++)**

**if(\*pb==\*pa)**

**break;**

**if(pa==a+d) /\*表示a中所有数据都和b的某个元素比对完，**

**没有找到相同的，则应该将b中的该元素写入c\*/**

**{**

**\*pc=\*pb;**

**pc++;**

**f++; //记录b中写入c的元素个数**

**}**

**}**

**cout<<"并集ｃ的元素为："<<endl;**

**for(pc=c;pc<c+d+f;pc++)**

**cout<<setw(3)<<\*pc;**

**}**

* **求交集（两个数组无序）**

**#include <iostream>**

**#include <iomanip>**

**using namespace std;**

**void main(void)**

**{**

**const int M=20,N=10;**

**int a[M],b[N],c[N];**

**int m,n,f=0,\*pa,\*pb,\*pc;**

**/\* m、n为数组a、b的实际长度；f记录数组c实际长度；pa、pb、pc为指向数组a、b、c的指针 \*/**

**cout<<"输入数组a中元素的个数："<<endl;**

**cin>>m;**

**cout<<"输入数组a的"<<m<<"个元素："<<endl;**

**for(pa=a;pa<a+m;pa++)**

**cin>>\*pa; //⑤用指针变量进行数组元素的输入**

**cout<<"输入数组b中元素的个数："<<endl;**

**cin>>n;**

**cout<<"输入数组b的"<<n<<"个元素："<<endl;**

**for(pb=b;pb<b+n;pb++)**

**cin>>\*pb; //⑤用指针变量进行数组元素的输入**

**for( pa=a,pc=c;pa<a+m;pa++ )**

**for( pb=b;pb<b+n;pb++ )**

**if(\*pa==\*pb)**

**{**

**\*pc++=\*pa; //将a、b中相同元素写入c,同时指针pc后移**

**f++; //用变量f记录交集数组c的实际长度**

**break; //找到a、b中相同元素即退出本轮比较**

**}**

**cout<<"交集c的各个元素依次为:"<<endl;**

**for (pc=c;pc<c+f;pc++)**

**cout<<setw(3)<<\*pc;**

**}**

**求交集（两个数组有序）**

**#include <iostream>**

**#include <iomanip>**

**using namespace std;**

**void main(void)**

**{**

**const int M=20,N=10;**

**int a[M],b[N],c[N];**

**int m,n,i=0,j=0,f=0,\*pa,\*pb,\*pc;**

**/\* m、n为数组a、b的实际长度；f记录数组c实际长度；pa、pb、pc为指向数组a、b、c的指针 \*/**

**cout<<"输入数组a中元素的个数："<<endl;**

**cin>>m;**

**cout<<"输入数组a的"<<m<<"个元素(要求有序，可重复)："<<endl;**

**for(pa=a;pa<a+m;pa++)**

**cin>>\*pa; //⑤用指针变量进行数组元素的输入**

**cout<<"输入数组b中元素的个数："<<endl;**

**cin>>n;**

**cout<<"输入数组b的"<<n<<"个元素(要求有序，可重复)："<<endl;**

**for(pb=b;pb<b+n;pb++)**

**cin>>\*pb; //⑤用指针变量进行数组元素的输入**

**pa=a;pb=b;pc=c;**

**while(i<m && j<n )**

**{**

**if (\*(pa+i)>\*(pb+j))**

**{**

**j++;**

**}**

**else**

**if (\*(pa+i)<\*(pb+j))**

**{**

**i++;**

**}**

**else**

**{**

**\*(pc+f)=\*(pa+i);**

**while(\*(pa+(++i))==\*(pc+f)); //去掉a中重复元素**

**while(\*(pb+(++j))==\*(pc+f)); //去掉b中重复元素**

**f++;**

**}**

**}**

**cout<<"交集c的各个元素依次为:"<<endl;**

**for (pc=c;pc<c+f;pc++)**

**cout<<setw(3)<<\*pc;**

**}**

* **对输入的一串密码信息（不超过40个字符）进行密码翻译，翻译的规则如下：凡是字母都译成字母表中的下一个字母，即A、a译成B、b，B、b译成C、c，……，Z、z译成A、a其它字符不变。**

**#include<iostream>**

**using namespace std;**

**void main( )**

**{ char str[50],ch;**

**int i;**

**cout<<"Please input an string:"<<endl;**

**cin>>str;**

**for(i=0;(ch=str[i])！='\0';i++)**

**{ if('a'<=ch&&ch<'z'||'A'<=ch&&ch<'Z') //注意不要写成’a’<=ch<’z’**

**str[i]+=1;**

**else if(ch=='z'||ch=='Z')**

**str[i]-=26;**

**}**

**cout<<str<<endl;**

**}**

* **P95 例4.11**

**#include <iostream>**

**#include <string>**

**using namespace std;**

**void main(void)**

**{**

**string a="What must they do!";**

**char b='t';**

**int i,j,k,c=0;**

**k=a.size();**

**for(i=0;i<a.size();i++)**

**if(a[i]==b)**

**{**

**c++;**

**for(j=i;j<a.size()-1;j++)**

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

**i--;**

**k--;**

**}**

**for(j=0;j<c;j++)**

**a[k+j]='\0';**

**cout<<c<<endl<<a<<endl;**

**}**

* **猴子选大王**

**#include <iostream>**

**using namespace std;**

**void main (void)**

**{**

**cout<<"请输入猴子总数N和报数节点m："<<endl;**

**int N,m;**

**cin>>N>>m;**

**m=m%N;**

**int a[500],i,j;**

**for(i=0,j=1;i<=N-1;i++,j++) a[i]=j;**

**for(;N>1;N--)**

**{**

**for(i=m;i<=2\*m+1;i++)**

**{**

**a[i-m+N]=a[i-m];**

**}**

**for(j=0;j<=N-1;j++) a[j]=a[j+m];**

**}**

**cout<<"猴王是："<<a[0]<<endl;**

**}**

**字符数组的应用**：

1. **奇数和偶数字符的统计，**

**#include <iostream>**

**using namespace std;**

**// 统计一含有数字的字符串中基数与偶数字符的个数**

**void main(void)**

**{**

**char str[50];**

**int odd=0,even=0;**

**cin>>str;**

**for(int i=0;str[i]!='\0';i++)**

**if(str[i]>='0'&&str[i]<='9'){**

**if((str[i]-'0')%2==0) even++;**

**else if((str[i]-'0')%2!=0) odd++;**

**}**

**cout<<"odd is"<<odd<<endl;**

**cout<<"even is"<<even<<endl;**

**}**

1. **删除 ASCII 码值能被m 整除的字符，**

**#include <iostream>**

**using namespace std;**

**// 删除 ASCII 码值能被m 整除的字符//假设被5整除,因为字符的存储是按照ASCII码存放的，所以字符数据可以参与数学运算。**

**void main(void)**

**{**

**char str[50];**

**int i,j;**

**cin>>str;**

**for( i=0;str[i]!='\0';i++)**

**if(str[i]%5==0)**

**for(j=i;str[j]!=0;j++)**

**str[j]=str[j+1];**

**cout<<str<<endl;**

**}**

**3.删除其中的数字字符，**

**#include <iostream>**

**using namespace std;**

**// 输入一字符串，其中含有数字字符，请删除其中的数字字符**

**void main(void)**

**{**

**char str[50];**

**int i,j;**

**cin>>str;**

**i=0;**

**while(str[i]!='\0')**

**{**

**if(str[i]>='0'&&str[i]<='9')**

**for(j=i;str[j]!=0;j++)**

**str[j]=str[j+1];**

**else i++;**

**}**

**cout<<str<<endl;**

**}**

**4.查找指定字符第一次出现的位置，**

**#include <iostream>**

**using namespace std;**

**// 输入一字符串，查找指定字符第一次出现的位置**

**void main(void)**

**{**

**char str[50],ch;**

**int i,pos;**

**cout<<"input string:";**

**cin>>str;**

**cout<<"input char:";**

**cin>>ch;**

**for(i=0;str[i]!='\0';i++)**

**if(str[i]==ch){pos=i; break;}**

**cout<<"the char"<<ch<<" pos is:"<<pos<<endl;**

**}**

**5.逆序重新存放，**

**#include <iostream>**

**using namespace std;**

**// 输入一字符串，将其按反序重新存放输出**

**void main(void)**

**{**

**char str[50],ch;**

**int i,j;**

**cout<<"input string:";**

**cin>>str;**

**j=strlen(str)-1;**

**for(i=0;i<j;i++,j--)**

**{ch=str[i];str[i]=str[j];str[j]=ch;}**

**cout<<"the invert of the string:"<<str<<endl;**

**}**

**//也可以这样实现**

**#include <iostream>**

**using namespace std;**

**// 输入一字符串，将其按反序重新存放输出**

**void main(void)**

**{**

**char str[50],ch;**

**int i,j;**

**cout<<"input string:";**

**cin>>str;**

**j=strlen(str);**

**for(i=0;i<j/2;i++)**

**{ch=str[i];str[i]=str[j-i-1];str[j-i-1]=ch;}**

**cout<<"the invert of the string:"<<str<<endl;**

**}**

**字符串类的应用：以 string s1,s2;为例，掌握 getline(cin,s1)、 s1.size()、s[i]以及 s1 与 s2 的比较操作；**

**1.对一串信息进行替换操作**

**#include <iostream>**

**#include<string>**

**using namespace std;**

**// 输入一字符串，将其中的数字用字符用其后三位代替即SCII码值加3，如字符‘0’，用字符‘3‘代替，**

**//字符’6‘换成’9‘，’7‘换成’0‘，’8‘换成’1‘，’9‘换成’2‘；**

**void main(void)**

**{**

**string str;**

**int i,j;**

**cout<<"input string:";**

**cin>>str;**

**j=str.size();**

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

**if(str[i]>='0'&&str[i]<='9') {**

**str[i]=str[i]+3;**

**if(str[i]>'9')str[i]=str[i]-10;**

**}**

**cout<<"The string after changing:"<<str<<endl;**

**}**

**2. 将一个字符串的特定字母复制到另一个字符串。**

**#include <iostream>**

**using namespace std;**

**// 将一个字符串的特定字母复制到另一个字符串**

**//将字符串a中的数字赋值到字符串b中**

**void main(void)**

**{**

**char str1[80],str2[40];**

**int i,j,k=0;**

**cout<<"input string:";**

**cin>>str1;**

**j=strlen(str1);**

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

**if(str1[i]>='0'&&str1[i]<='9')**

**str2[k++]=str1[i];**

**str2[k]='\0';**

**cout<<"The new string :"<<str2<<endl;**

**}**

**#include <iostream>**

**#include<string>**

**using namespace std;**

**// 将一个字符串的特定字母复制到另一个字符串**

**//将字符串a中的数字赋值到字符串b中**

**void main(void)**

**{**

**string str1,str2(" ");**

**int i,j,k=0;**

**cout<<"input string:";**

**cin>>str1;**

**j=str1.size()-1;**

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

**if(str1[i]>='0'&&str1[i]<='9')**

**str2[k++]=str1[i];**

**str2[k]='\0';**

**cout<<"The new string :"<<str2<<endl;**

**}**

**3. 以 string s1,s2;为例，掌握 getline(cin,s1)、 s1.size()、s[i]以及 s1 与 s2 的比较操作**

**#include <iostream>**

**#include<string>**

**using namespace std;**

**// 输入3个字符串，输出其中最大的字符串**

**void main(void)**

**{**

**string str1,str2,str3;**

**getline(cin,str1);**

**getline(cin,str2);**

**getline(cin,str3);**

**if(str1>str2&&str1>str3)**

**cout<<"the Big one is:"<<str1<<endl;**

**if(str2>str1&&str2>str3)**

**cout<<"the Big one is:"<<str2<<endl;**

**if(str3>str1&&str3>str2)**

**cout<<"the Big one is:"<<str3<<endl;**

**}**

**4.**

**#include <iostream>**

**#include<string>**

**using namespace std;**

**// 输入英文翻译成中文：monday,则输出中文的“星期一”**

**void main(void)**

**{**

**string str1;**

**cout<<"Please input a day to translate into Chinese,quit to exit"<<endl;**

**while(true)**

**{**

**getline(cin,str1);**

**if(str1=="quit") break;**

**if(str1=="Monday")**

**cout<<"今天是星期一"<<endl;**

**if(str1=="Wednesday")**

**cout<<"今天是星期三"<<endl;**

**if(str1=="Tuesday")**

**cout<<"今天是星期二"<<endl;**

**if(str1=="Thursday")**

**cout<<"今天是星期四"<<endl;**

**if(str1=="Friday")**

**cout<<"今天是星期五"<<endl;**

**if(str1=="Saturday")**

**cout<<"今天是星期六"<<endl;**

**if(str1=="Sunday")**

**cout<<"今天是星期日"<<endl;**

**}**

**}**

**3、函数的应用简单计算器的实现：加、减、乘、除的运算符由随机数来确定递归函数：（1）阶乘运算 n!、幂运算 ab、斐波那契数列；（2）给出递推关系，要求子函数实现（数值传递）。**

**函数参数的传递：**

**（1）子函数主要内容包括：求符合特定情况的数据和运算，两个数组的比较，把指定范围数组元素平移，对键盘输入的数据进行选取，对传递过来的数据进行求余等的特定操作，数组中找出符合特征值的数据组成新数组，删除数组中的特征值数据，删除有序数组中重复数据，统计具有某特征值的数据，统计字符数组中特征值数据出现的次数，将两个数据拆分并按规定重组为一个新数据，判断两个正整**

**数是否为互质数，将一个数据有序插入到一个有序的数列中，按规律生成一组数据并输出，字符数组中字符变换操作，字符串（string）中字符变换操作等。**

**（2）主函数与子函数之间的数据互交换：包括：数值传递、指针传递、数组传递、引用传递。**

**函数需要掌握的例题：**

**1）简单计算器的实现：加、减、乘、除的运算符由随机数来确定。//即产生一随机数在1—4之间，分别代表+，-，\*，/四种算法**

**#include <iostream>**

**#include<cstdlib>**

**#include<ctime>**

**using namespace std;**

**void main(void)**

**{**

**int ap;**

**double a,b,c;**

**cout<<"please input two number:"<<endl;**

**cin>>a>>b;**

**srand(time(NULL));**

**ap=1+rand()%(4-1+1);//产生一个1—4之间的随机数**

**switch(ap)**

**{case 1:c=a+b;cout<<a<<"+"<<b<<"="<<c<<endl; break;**

**case 2:c=a-b;cout<<a<<"-"<<b<<"="<<c<<endl; break;**

**case 3:c=a\*b;cout<<a<<"\*"<<b<<"="<<c<<endl; break;**

**case 4:if(b==0) cout<<"cannot devide!"<<endl;**

**else**

**{c=a/b;cout<<a<<"/"<<b<<"="<<c<<endl;}**

**break;**

**}**

**}**

**2. 递归函数：（1）阶乘运算 n!、幂运算 ab、斐波那契数列；**

**#include <iostream>**

**using namespace std;**

**//递归函数：（1）阶乘运算 n!、幂运算 ab、斐波那契数列；**

**double fun(int n)**

**{**

**double f;**

**if(n==0||n==1)f=1;**

**else f=fun(n-1)\*n;**

**return f;**

**}**

**//幂函数**

**int mihanshu(int x,int y)**

**{**

**int m;**

**if(y==0)m=1;**

**else m=x\*mihanshu(x,y-1);**

**return m;**

**}**

**//斐波那契数列函数**

**int fib(int n)**

**{ int f;**

**if(n==0||n==1) f=1;**

**else f=fib(n-1)+fib(n-2);**

**return f;**

**}**

**//以上函数的调用**

**void main(void)**

**{**

**cout<<fun(3)<<endl;**

**cout<<mihanshu(3,4)<<endl;**

**cout<<fib(5)<<endl;**

**}**

**3. 如何比较两个数组元素然后把不同的元素放到另一个数组里。**

**#include <iostream>**

**#include <iomanip>**

**using namespace std;**

**//函数，比较集合a，b，把不同的元素放入c集合中，**

**//d，e，f分别是a，b，c集合的元素个数，f必须采用**

**//地址引用，这样才可以把f的值传递给主程序**

**void compare(int a[],int b[],int c[],int d,int e,int &f)**

**{ int i,j,t=0;**

**bool mm;**

**//把a集合中与b集合不同的元素放入c集合中，注意t值的变化**

**for(i=0;i<=d-1;i++)**

**{ mm=true;**

**for(j=0;j<=e-1;j++)**

**if(a[i]==b[j]) mm=false;**

**if(mm) {c[t]=a[i];t++;}**

**}**

**//把b集合中不属于a集合的元素放入c集合中，注意t值的变化**

**for(i=0;i<=e-1;i++)**

**{ mm=true;**

**for(j=0;j<=d-1;j++)**

**if(b[i]==a[j]) mm=false;**

**if(mm) {c[t]=b[i];t++;}**

**}**

**//一定要把t值赋给f，然后传递给主程序**

**f=t;**

**}**

**void main(void)**

**{**

**const int N=20;**

**const int M=20;**

**int n,m;**

**int a[N],b[M],c[N],t=0;**

**cout<<"请输入数组A的实际长度："<<endl;**

**cin>>n;**

**cout<<"请输入数组A的所有元素："<<endl;**

**for(int i=0;i<n;i++)cin>>a[i];**

**cout<<"请输入数组B的实际长度："<<endl;**

**cin>>m;**

**cout<<"请输入数组B的所有元素："<<endl;**

**for(int i=0;i<m;i++)cin>>b[i];**

**compare(a,b,c,n,m,t);**

**cout<<"A和B中不同的元素为："<<endl;**

**for(int i=0;i<t;i++)**

**cout<<setw(6)<<c[i];**

**cout<<endl;**

**}**

**4. 把指定范围数组元素平移，一维数组中的数据为1,2,3,4,5,6,7,8,9,10，输入3，移动后一维数组中的值为5,6,7,8,9,10,1,2,3,4.**

**#include<iostream>**

**#include<iomanip>**

**using namespace std;**

**void shif(int a[],int n,int p)//p表示移动的元素位置，n表示数组元素的个数**

**{**

**int i,t;**

**while(p>0) //右移算法**

**{**

**t=a[n-1];**

**for(i=n-2;i>=0;i--) a[i+1]=a[i];**

**a[0]=t;**

**p--;**

**}**

**}**

**void main()**

**{**

**const int N=10;**

**int s[N];**

**int n,i,pos;**

**cout<<"请输入数组的实际长度："<<endl;**

**cin>>n;**

**cout<<"请输入数组的各个元素："<<endl;**

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

**cin>>s[i];**

**cout<<"请输入右移的位数："<<endl;**

**cin>>pos;**

**shif(s,n,pos);**

**cout<<"右移"<<pos<<"位的结果为："<<endl;**

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

**cout<<setw(6)<<s[i];**

**cout<<endl;**

**}**

**#include<iostream>**

**#include<iomanip>**

**using namespace std;**

**void shif(int a[],int n,int p)//p表示移动的元素位置，n表示数组元素的个数**

**{**

**int i,t;**

**for(;p>0;p--) //左移算法**

**{**

**t=a[0];**

**for(i=0;i<n-1;i++) a[i]=a[i+1];**

**a[n-1]=t;**

**}**

**}**

**void main()**

**{**

**const int N=10;**

**int s[N];**

**int n,i,pos;**

**cout<<"请输入数组的实际长度："<<endl;**

**cin>>n;**

**cout<<"请输入数组的各个元素："<<endl;**

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

**cin>>s[i];**

**cout<<"请输入左移的位数："<<endl;**

**cin>>pos;**

**shif(s,n,pos);**

**cout<<"左移"<<pos<<"位的结果为："<<endl;**

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

**cout<<setw(6)<<s[i];**

**cout<<endl;}**

**5. 数组中找出符合特征值的数据组成新数组；**

**把数组a中的偶数放入数组b中。**

**#include <iostream>**

**using namespace std;**

**//函数：把集合a中的偶数放入集合b中，因为集合b的个数不确定//且要返回给主程序，所以用地址引用e来实现，d代表a集合中元素的个数，**

**void oddb(int a[],int b[],int d,int &e)**

**{ int i;**

**for(i=0,e=0;i<=d-1;i++)**

**if(a[i]%2==0) b[e++]=a[i];**

**}**

**void main(void)**

**{**

**int a[10],b[10],t=0;**

**for(int i=0;i<=9;i++)cin>>a[i];**

**oddb(a,b,10,t);**

**for(int i=0;i<t;i++)**

**cout<<b[i]<<" "<<endl;**

**}**

**6. 删除数组中的特征值数据**

**#include <iostream>**

**using namespace std;**

**//函数：把数组a中的的值小于X的数组元素删掉,n代表数组元素的个数,数组**

**//元素的个数存在变化，所以要采用地址引用**

**void deletea(int a[],int x,int &n)**

**{ int t,i,j;**

**t=n;**

**i=0;**

**while(i<=t-1)**

**{if(a[i]==x) { for(j=i;j<=t-2;j++)a[j]=a[j+1];t--;}**

**else i++;}**

**n=t;**

**}**

**void main(void)**

**{**

**int a[10],x,nm=10,i;**

**for( i=0;i<=9;i++)cin>>a[i];**

**cout<<"input x:"<<endl;**

**cin>>x;**

**deletea(a,x,nm);**

**for(i=0;i<nm;i++)cout<<a[i]<<" ";}**

**7．删除有序数组中重复数据**

**#include <iostream>**

**using namespace std;**

**//函数：把有序数组a中的重复元素删掉,n代表数组元素的个数,数//组元素的个数存在变化，所以要采用地址引用**

**void deletesame(int a[],int &n)**

**{ int t,i,j;**

**t=n;**

**i=0;**

**while(i<=t-2)**

**{if(a[i]==a[i+1]) { for(j=i;j<=t-2;j++)a[j]=a[j+1];t--;}**

**else i++;}**

**n=t;**

**}**

**void main(void)**

**{**

**int a[10],x,nm=10,i;**

**for( i=0;i<=9;i++)cin>>a[i];**

**deletesame(a,nm);**

**for(i=0;i<nm;i++)cout<<a[i]<<" ";**

**}**

**8．统计具有某特征值的数据**

**//统计数组中偶数的被3整除数的个数。**

**#include <iostream>**

**using namespace std;**

**//函数：统计数据元素中被n整除的元素个数,m表示数组元素的个数**

**//函数返回值为数组中被n整除的元素个数。**

**int divedS(int a[],int m,int n)**

**{ int t,i;**

**t=0;**

**for(i=0;i<m;i++)**

**if(a[i]%n==0)t++;**

**return t;**

**}**

**void main(void)**

**{**

**int a[10],n=10,num=0,i;**

**for( i=0;i<=9;i++)cin>>a[i];**

**num=divedS(a,10,3);**

**cout<<"The data in array a has "<<num<<" number dived by 3"<<endl;**

**}**

**9.统计字符数组中特征值数据出现的次数**

**#include <iostream>**

**using namespace std;**

**//函数：统计字符数组a中，字符n出现的次数。**

**int chartime(char a[],char n)**

**{ int i=0,t=0;**

**while(a[i]!='\0')**

**{ if(a[i]==n) t++;**

**i++;**

**}**

**return t;**

**}**

**void main(void)**

**{**

**char a[40],x;**

**int time=0;**

**cout<<"请输入需要测试的字符串："<<endl;**

**cin.getline(a,40);**

**cout<<"请输入需要统计的字符："<<endl;**

**cin>>x;**

**time=chartime(a,x);**

**cout<<time<<endl;**

**}**

**10.将两个数据拆分并按规定重组为一个新数据**

**#include <iostream>**

**using namespace std;**

**//如果整数a=134，b=25，将其拆分组成一个整数c=12354。**

**int numdv(int a,int b)**

**{**

**int s=0,x,y,t=1;**

**while(a>0&&b>0)**

**{**

**x=a%10;**

**y=b%10;**

**x=x\*t;**

**y=y\*t\*10;**

**s=s+x+y;**

**a=a/10;**

**b=b/10;**

**t=t\*100;**

**}**

**if(a>0) s=s+a\*t;**

**else s=s+b\*t;**

**return s;**

**}**

**void main(void)**

**{**

**int a,b,c;**

**cin>>a>>b;**

**c=numdv(a,b);**

**cout<<c<<endl;**

**}**

**11.判断两个正整数是否为互质数**

**#include <iostream>**

**using namespace std;**

**//函数：判断两个正整数是否互为质数。**

**//返回值为ture表示互为质数，否则不是**

**bool nofactor(int a, int b)**

**{ int i,t;**

**if(a>b) t=b;**

**else t=a;**

**for(i=2;i<=t;i++)**

**if(a%i==0&&b%i==0) break;**

**if(i<=t) return false;**

**else return true;**

**}**

**void main(void)**

**{**

**int a,b;**

**cin>>a>>b;**

**cout<<a<<"and "<<b<<" "<<nofactor(a,b)<<endl;**

**}**

**12.将一个数据有序插入到一个有序的数列中，**

**#include <iostream>**

**using namespace std;**

**//函数：在有序(从小到大）数组a中插入一个元素x,参数n表示数组元素的个数**

**void insert(int a[], int x,int &n)**

**{**

**int i=0,j;**

**while(a[i]<x&&i<n) i++;**

**for(j=n;j>i;j--)a[j]=a[j-1];**

**a[i]=x;**

**n=n+1;**

**}**

**void main(void)**

**{**

**int s[10],x,n=9,i;**

**for( i=0;i<=n-1;i++)cin>>s[i];**

**cout<<"input x to be inserted!"<<endl;**

**cin>>x;**

**insert(s,x,n);**

**for(i=0;i<=n-1;i++) cout<<s[i]<<" ";**

**}**

**13.按规律生成一组数据并输出，**

**#include <iostream>**

**#include<cstdlib>**

**#include<ctime>**

**using namespace std;**

**//生成10个被3整除的数(概数在100以内），并放入数组a中，并输出。**

**void main(void)**

**{**

**int s[10],x,i;**

**srand(time(NULL));**

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

**{**

**while(true)**

**{ x=1+rand()%(100-1+1);**

**if(x%3==0){s[i]=x;break;}**

**}**

**}**

**for(i=0;i<=9;i++) cout<<s[i]<<" ";**

**}**

**14.字符数组中字符变换操作.**

**#include <iostream>**

**using namespace std;**

**//将字符串数组中大写字母转换为小写字母**

**void lower(char a[])**

**{int i;**

**i=0;**

**while(a[i]!='\0')**

**{ if(a[i]>='A'&&a[i]<='Z') a[i]=a[i]+32;**

**i++;**

**}**

**}**

**void main(void)**

**{**

**char st[30];**

**cin>>st;**

**lower(st);**

**cout<<st<<endl;**

**}**

**15.字符串（string）中字符变换操作等。**

**#include <iostream>**

**#include<string>**

**using namespace std;**

**//将字符串数组中数字改为'a'**

**void chang\_num\_char( string a)**

**{**

**int i,t;**

**t=a.size();**

**for(i=0;i<t;i++)**

**if(a[i]>='0'&&a[i]<='9')**

**a[i]='a';**

**cout<<"将数字用字母a替换后的结果串为："<<endl;**

**cout<<a<<endl;**

**}**

**void main(void)**

**{**

**string ss;**

**cout<<"请输入需要处理的字符串（其中包括数字）："<<endl;**

**getline(cin,ss);**

**chang\_num\_char(ss);**

**}**

**16. 求符合特定情况的数据和运算。**

**#include <iostream>**

**#include<string>**

**using namespace std;**

**//求数组元素中被3整除的数的乘积**

**int cji(int a[],int n)//n代表数组中元素的个数**

**{**

**int sum=1,i;**

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

**if(a[i]%3==0) sum=sum\*a[i];**

**return sum;**

**}**

**void main(void)**

**{**

**int s[9],sj,i;**

**for(i=0;i<9;i++) cin>>s[i];**

**sj=cji(s,9);**

**cout<<sj<<endl;**

**}**

**17.对键盘输入的数据进行选取.**

**#include <iostream>**

**#include<string>**

**using namespace std;**

**//从键盘输入一组数据，把偶数存放在数组中.**

**//输入-1时结束**

**void inputN(int a[],int &n)**

**{**

**int i=0,a1;**

**cin>>a1;**

**while(a1!=-1)**

**{ if(a1%2==0) {a[i]=a1,i++;}**

**cin>>a1;**

**if(i>=30) return;**

**}**

**n=i;**

**}**

**void main(void)**

**{**

**int a[30],n=30;**

**inputN(a,n);**

**for(int i=0;i<n;i++)**

**cout<<a[i]<<" ";**

**}**

**18，对传递过来的数据进行求余等的特定操作。**

**#include <iostream>**

**#include<string>**

**using namespace std;**

**//求数组元素对n求余数**

**void yji(int a[],int m,int n)//m代表数组中元素的个数**

**{**

**int i;**

**for(i=0;i<m;i++)**

**a[i]=a[i]%n;**

**return ;**

**}**

**void main(void)**

**{**

**int s[9],i;**

**for(i=0;i<9;i++) cin>>s[i];**

**yji(s,9,3);**

**for(i=0;i<9;i++) cout<<s[i]<<" ";**

**}**

**\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\***

**4、指针的应用利用指针在数组中完成：求数组中的特征值元素，数据的有序插入，计算并输出交集、并集，折半查找，使用交换排序法（起泡法、冒泡法）排序，Fibonacci 数列。**

**利用指针在字符数组中完成：指定位置插入子串，回文的判定，键盘输入并记录其中的字母。**

**函数参数传递完成：两个数据的交换，将数组中的数据按相反顺序存放。**

**指针这一章主要掌握两点：**

**1. 指向变量的指针（定义，赋值，引用）：**

**定义： int a，\*p；**

**对指针变量赋值：1）可以定义指针变量时候赋值：**

**如：int a，\*p=&a；**

**2）也可以在定义指针变量后赋值：**

**如：int a，\*p；**

**p=&a；**

**指针变量的引用：当对指针变量赋值后，可以用\*p 来代替以前的a变量。**

**指针变量只能的赋值只能是一个变量的地址或整数0（表示指针式空指针），也可以将一个指针赋给另外一个指针。如：int a，\*p，\*q；**

**P=&a，q=p；**

**2）指针访问一维数组：**

**如：int a[10]，\*p=a（或者\*p=&a[0]）都表示把数组的第一个元素的地址赋给指针p。**

**int a[10],\*p1,\*p2;**

**P1=a,p2=a+6 ;那么p2-p1表示指针p1和p2之间有多少个数组元素，p2>P1表示p2指针指向的是数组小标大的元素的地址，P1==p2表示p1和p2表示的统一数组元素的地址。**

**如果有\*p=a，则数组元素的引用有4种方式:a[i] p[i], \*(a+i),\*(p+i).而表达式a++是错误的（因为a是常量），p++是正确的（p是一个变量，表示是当前数组元素的下一个数组元素的地址）。**

**需要掌握的类似例题：**

**1. 指针完成数据的有序插入（P121,例5.9）**

**2. 求数组中的特征值元素（如一数组有n个元素，输出其中比平均值小的数）**

**#include <iostream>**

**using namespace std;**

**void main(void)**

**{**

**double a[6],sum,\*p;**

**p=a;**

**for(;p<=a+5;p++) cin>>\*p;**

**sum=0;**

**for(p=a;p<=a+5;p++)sum=sum+\*p;**

**sum=sum/6;**

**for(p=a;p<=a+5;p++) if(\*p<sum) cout<<\*p<<" ";**

**}**

**3. 计算并输出交集、并集。**

**//交集**

**#include <iostream>**

**using namespace std;**

**void main(void)**

**{**

**int a[6],b[5],jj[5],\*pa,\*pb,\*pc;//a,b,jj分别表示三个集合，其中jj表示交集**

**int f=0;//表示交集元素的个数**

**//输入集合a**

**pa=a;**

**for(;pa<=a+5;pa++) cin>>\*pa;**

**//输入集合b**

**for(pb=b;pb<=b+4;pb++)cin>>\*pb;**

**//求交集**

**pc=jj;**

**for(pa=a;pa<=a+5;pa++)**

**for(pb=b;pb<=b+4;pb++)**

**if(\*pa==\*pb){\*pc=\*pb;pc++;f++;}**

**//输出交集**

**for(pc=jj;pc<jj+f;pc++) cout<<\*pc<<" ";**

**}**

**并集**

**#include <iostream>**

**#include <string>**

**using namespace std;**

**void main(void)**

**{**

**int a[6],b[5],bj[11],\*pa,\*pb,\*pc;//a,b,jj分别表示三个集合，其中bj表示并集**

**int f=0;//表示并集元素的个数**

**//输入集合a**

**pa=a;**

**for(;pa<=a+5;pa++) cin>>\*pa;**

**//输入集合b**

**for(pb=b;pb<=b+4;pb++)cin>>\*pb;**

**//求并集，一定要注意f的值的变化，表示并集的个数**

**//1.先其中的一个集合放入并集，可以先把a集合的数据放入并集bj中**

**for(pa=a,pc=bj;pa<=a+5;pa++,pc++,f++)**

**\*pc=\*pa;**

**//2.把b集合中不在a集合中的元素放入并集bj中**

**for(pb=b;pb<=b+4;pb++)**

**{**

**for(pa=a;pa<=a+5;pa++)**

**if(\*pa==\*pb)break;**

**if(pa>a+5){\*pc=\*pb;pc++;f++;}**

**}**

**//输出交集**

**for(pc=bj;pc<bj+f;pc++) cout<<\*pc<<" ";**

**}**

**4. 折半查找**

**#include <iostream>**

**using namespace std;**

**void main(void)**

**{**

**int a[8],x,top=0,bot=7,mid,\*p;**

**//输入一个从小到大序列的8个数据**

**for(p=a;p<a+8;p++) cin>>\*p;**

**//输入要查找的数**

**cin>>x;**

**p=a;**

**while(top<=bot)**

**{**

**mid=(top+bot)/2;**

**if(\*(p+mid)==x) break;**

**else if(x>\*(p+mid)) top=mid+1;**

**else bot=mid-1;**

**}**

**if(top<=bot) cout<<"pos is:"<<mid<<endl;**

**else cout<<"This number is not in this array ";**

**}**

**5. 使用交换排序法（起泡法、冒泡法）排序**

**#include <iostream>**

**using namespace std;**

**void main(void)**

**{ //冒泡法排序**

**int a[8],tmp,\*p1,\*p2;**

**for(p1=a;p1<a+8;p1++) cin>>\*p1;**

**for(p1=a;p1<=a+6;p1++)**

**for(p2=a;p2<a+7-(p1-a);p2++)//p1-a表示什么？**

**if(\*p2>\*(p2+1)){tmp=\*p2;\*p2=\*(p2+1);\*(p2+1)=tmp;}**

**for(p1=a;p1<=a+7;p1++) cout<<\*p1<<" ";**

**}**

**#include <iostream>**

**using namespace std;**

**void main(void)**

**{ //选择法排序**

**int a[8],tmp,\*p1,\*p2;**

**for(p1=a;p1<=a+7;p1++) cin>>\*p1;**

**for(p1=a;p1<=a+6;p1++)**

**for(p2=p1+1;p2<=a+7;p2++)**

**if(\*p1>\*p2){tmp=\*p1;\*p1=\*p2;\*p2=tmp;}**

**for(p1=a;p1<=a+7;p1++) cout<<\*p1<<" ";**

**}**

**6. Fibonacci 数列。**

**#include <iostream>**

**using namespace std;**

**void main(void)**

**{ //选择法排序**

**int a[20],\*p=a,t=0;**

**\*p=1;\*(p+1)=1;**

**for(p=a+2;p<=a+19;p++) \*p=\*(p-1)+\*(p-2);**

**for(p=a;p<=a+19;p++,t++)**

**{if(t%5==0) cout<<endl;**

**cout<<\*p<<" ";**

**}**

**}**

**7. 利用指针在字符数组中完成：指定位置插入子串**

**输入一字符串，在最大的ASCII码最大的值前面插入子串“xyz”。**

**#include <iostream>**

**using namespace std;**

**void main(void)**

**{ //在一个字符串的ascii最大的字符前插入字符串“xyz”**

**char a[20];**

**char big,\*p,\*pos;**

**cin>>a;**

**p=&a[0],pos=p;**

**big=\*p;**

**//找ASCII码最大的字符所在的位置；**

**while(\*p!='\0')**

**{**

**if(big<\*p){big=\*p;pos=p;}**

**p++;**

**}**

**//移动字符串的位置从pos位置处往后移动3个字符的位置，注意字符串的结束标识也一起移动的的哦；**

**for(p=a+strlen(a);p>=pos;p--) \*(p+3)=\*p;**

**//插入子串**

**\*pos='x';**

**\*(pos+1)='y';**

**\*(pos+2)='z';**

**cout<<a;**

**}**

**8. 指针完成回文的判定（回文就是从左边读与从右边读是一样的；如： bookoob是回文）**

**#include <iostream>**

**using namespace std;**

**void main(void)**

**{ //在一个字符串的是否回文**

**char a[20],\*pa,\*pb;**

**cin.getline(a,19);**

**for(pa=a,pb=a+strlen(a)-1;pa<pb;pa++,pb--)**

**if(\*pa!=\*pb) break;**

**if(pa<pb) cout<<a<<" is not 回文"<<endl;**

**else cout<<a<<" is a 回文"<<endl;**

**}**

**9. 利用指针键盘输入并记录其中的字母。**

**#include <iostream>**

**using namespace std;**

**void main()**

**{ //在一个字符串的字母存放在字符数组中。**

**char a[20],\*pa,x;**

**pa=a;**

**cout<<"请输入一字符串以'#'结束"<<endl;**

**cin>>x;**

**while(x!='#')**

**{**

**if(x>='a'&&x<='z'||x>='A'&&x<='Z') {\*pa=x;pa++;}**

**cin>>x;**

**}**

**\*pa='\0';**

**cout<<a<<endl;**

**}**

**10.函数参数传递完成：两个数据的交换，将数组中的数据按相反顺序存放**

**#include <iostream>**

**using namespace std;**

**void swap(int \*p1,int \*p2)**

**{**

**int tmp;**

**tmp=\*p1;**

**\*p1=\*p2;**

**\*p2=tmp;**

**}**

**void main(void)**

**{**

**int a[10],i,\*p;**

**for(p=a;p<=a+9;p++) cin>>\*p;**

**for(p=a,i=0;i<=9/2;i++)**

**swap(\*(p+i),\*(p+9-i));**

**for(p=a;p<=a+9;p++)cout<<\*p<<" ";**

**}**

**//该函数也可以写成如下的引用传递**

**void swap(int &p1,int &p2)**

**{**

**int tmp;**

**tmp=p1;**

**p1=p2;**

**p2=tmp;**

**}**

**//调用时时候要注意**

**void main(void)**

**{**

**int a[10],i,\*p;**

**for(p=a;p<=a+9;p++) cin>>\*p;**

**for(p=a,i=0;i<=9/2;i++)**

**swap((p[i],p[9-i]);**

**for(p=a;p<=a+9;p++)cout<<\*p<<" ";**

**}**