Program 1

Get the sum of a double- dimension array.

Program code

#include <iostream>

using namespace std;

double sum(double a[20][20],int hang, int lie);

int main ()

{

double a[20][20];

int hang,lie;

cin>>hang>>lie;

for(int i =0;i<=hang-1;i++)

for (int j=0;j<=lie-1;j++)

cin>>a[i][j];

cout<<sum(a,hang,lie)<<endl;

return 0;

}

double sum(double a[20][20],int hang, int lie)

{

double total=0;

for(int i =0;i<=hang-1;i++)

for (int j=0;j<=lie-1;j++)

total=total+a[i][j];

return total;

}

Program analysis

Using the array to transfer the address of the result.

Program result

Program 2

Output the inverted order of a char array “hello world”

Program code

#include <iostream>

using namespace std;

void change(char s1[],char s2[]);

int main ()

{

char s1[]="hello world!";

char s2[20];

change(s1,s2);

cout<<s2<<endl;

return 0;

}

void change(char s1[],char s2[])

{

int l=strlen(s1)-1;

for(int i=0;s1[i]!='\0';i++)

s2[l-i]=s1[i];

s2[i]='\0';

}

Program analysis

Create a char array. And transfer the address of the original array and new array , then do the invert, output the new array.

Program result

Program 3

Calculate the sum of the number in the leading diagonal of a matrix.

Program code

#include <iostream>

using namespace std;

double sum(double a[30][30],int n);

int main ()

{

int n;

cin>>n;

double a[30][30];

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

for(int j=0;j<=n-1;j++)

cin>>a[i][j];

cout<<sum(a,n)<<endl;

return 0;

}

double sum(double a[30][30],int n)

{

double total=0;

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

total=total+a[i][i];

return total;

}

Program analysis

1. Pay attention to the two-dimension array in the function, you can’t only define as char a[], that’s wrong.
2. Using the address of the array to do the calculation.

Program result

Program 4

Sort a series of cities name

Program code

#include <iostream>

#include <string.h>

using namespace std;

void sort(char \*p[], int n);

int main ()

{

cout<<"输入10到20之间的一个数"<<endl;

int n;

cin>>n;

char a[20][20];

int i;

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

cin>>a[i];

char \*p[20];

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

p[i]=a[i];

sort(p,n);

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

cout<<p[i]<<endl;

return 0;

}

void sort(char \*p[], int n)

{

int i,j;

char \*temp=p[0];

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

for(j=0;j<=n-i-2;j++)

{

if (strcmp(p[j],p[j+1])>0)

{

temp=p[j];

p[j]=p[j+1];

p[j+1]=temp;

}

}

}

Program analysis

1. store the name in a two-dimension char array.
2. Create a new point char array to point to the original array.
3. Using strcmp to compare the ranking of the names
4. Change the point array to get the result.

Program result