C++語言上課資料

102年7月23日

**第二章 條件與迴圈**

// Vc312.cpp

// 巢狀迴圈練習 (九九乘法表)

#include <iostream.h>

int main()

{

int count1, count2;

for (count2=1; count2<=9; count2++) // 定義被乘數迴圈由1到9

{

for (count1=2; count1<=9; count1++) // 定義乘數迴圈由2到9

{

cout <<count1 <<'\*' <<count2 <<'=' // 輸出 count1 '\*' count2

<< count1\*count2 <<'\t'; // 輸出運算值跳至下一定位

}

cout << endl; // 輸出跳行

}

return 0;

}

// Vc313.cpp

// break 敘述練習 (搜尋第一個相符字元)

#include <iostream.h>

int main()

{

char string[25] = "Microsoft Visual C++ 5.0"; // 宣告與啟始字串

char letter; // 宣告字元變數

int count; // 宣告整數變數

for (count = 0; count <= 24; count++) // 利用迴圈顯示字串變數

cout << string[count];

cout << "\n請輸入要搜尋的字元："; // 顯示字串常數

cin >> letter; // 輸入字元

for (count = 0; count <= 24; count++) // 定義迴圈

{

if (string[count] == letter) // 找到相符字元

break; // 中斷迴圈

}

if (count <= 24) // 若計數值<=字串長度

cout << "第 " << ++count << " 個字元為 " // 則顯示字串位置

<< letter << '\n';

else

cout << "找不到相符字元\n"; // 否則顯示找不到

return 0;

}

// Vc314.cpp

// continue 敘述練習 (找尋 3 的倍數)

#include <iostream.h>

int main()

{

int count;

cout << "1 至 30 間 3 的倍數為：";

for (count = 1; count <= 30; count++) // 定義迴圈

{

if (count % 3 != 0) // 若不是3的倍數

continue; // 返回迴圈起點

cout << count << '\0'; // 顯示3的倍數並空格

}

cout << endl; // 輸出跳行

return 0;

}

**第四章 陣列與指標**

// Vc401.cpp

// 一維陣列練習 (找尋最大值)

#include <iostream.h>

int main()

{

int count, max = 0; // 宣告整數變數

int number[3] = {5, 9, 3}; // 宣告一維陣列變數

// 宣告 number[3] --> number[0], number[1], number[2]

for (count = 0; count <= 2; count++) // 找尋最大值迴圈

{

if (number[count] > max) // 若數值>緩衝器值

max = number[count]; // 則取代緩衝器值

}

cout << "5, 9, 3 三數的最大值 = " << max; // 顯示最大值

cout << endl << endl;

return 0;

}

// Vc402.cpp

// 一維陣列練習 (氣泡排序)

#include <iostream.h>

int main()

{

const int max = 5; // 宣告整數常數符號

int counti, countj, buffer = 0; // 宣告整數變數

int number[max] = {57, 19, 33, 92, 6}; // 宣告一維陣列變數

cout << "排序前："; // 顯示排序前資料

for (counti = 0; counti < max; counti++)

cout << number[counti] << '\0';

for (counti = 0; counti < max-1; counti++) // 排序外迴圈

for (countj = counti; countj < max; countj++) // 排序內迴圈

if (number[counti] > number[countj]) // 若須要則對調

{

buffer = number[counti];

number[counti] = number[countj];

number[countj] = buffer;

}

cout << "\n排序後："; // 顯示排序後資料

for (counti = 0; counti < max; counti++)

cout << number[counti] << '\0';

cout << endl << endl;

return 0;

}

// Vc403.cpp

// 一維陣列練習 (二分搜尋)

#include <iostream.h>

int main()

{

const int max = 5; // 宣告整數常數符號

int counti, countj, buffer = 0; // 宣告整數變數

int search, low = 0, high = max-1, middle; // 宣告整數變數

int number[max] = {57, 19, 33, 92, 6}; // 宣告整數陣列

cout << "排序前："; // 顯示排序前之值

for (counti = 0; counti < max; counti++)

cout << number[counti] << '\0';

cout << "\n請輸入要搜尋數值："; cin >> search; // 輸入搜尋值

for (counti = 0; counti < max-1; counti++) // 排序迴圈

for (countj = counti; countj < max; countj++)

if (number[counti] > number[countj])

{

buffer = number[counti];

number[counti] = number[countj];

number[countj] = buffer;

}

cout << "\n排序後："; // 顯示排序後之值

for (counti = 0; counti < max; counti++)

cout << number[counti] << '\0';

do // 搜尋迴圈

{

if ((low + high) % 2 > 0.5) // 計算搜尋位置

middle = (low + high) / 2 + 1;

else

middle = (low + high) / 2;

if (search < number[middle]) // 計算搜尋上限

high = middle - 1;

else if (search > number[middle]) // 計算搜尋下限

low = middle + 1;

else if (search = number[middle]) // 找到相符數值

break;

} while (low <= high);

if (low > high) // 顯示搜尋結果

cout << "\n找不到數值：" << search << endl;

else

cout << "\n找到數值：" << search << endl;

return 0;

}

練習題：

1、請用回圈做數字排列，輸出如下

1

12

123

1234

12345

123456

1234567

12345678

123456789

2、使用『一維陣列』、『if敘述』來求出所輸入5個數值之間的最大值。

3、氣泡排序，求出所輸入5個數值由小而大排列。

解答：

1、

#include <stdio.h>  
#include <stdlib.h>

int main()  
{  
    int a,b;  
    for(a=1; a<=9; a++)  
    {  
      for(b=1; b<=a; b++)  
      {  
       printf("%d",b);  
      }  
     printf("\n");  
    }  
    system("PAUSE");  
    return 0;  
}

邏輯：

 int a,b;                            宣告a,b為可變函數

 for(a=1; a<=9; a++)      (外迴圈)預設a值在1~9之間,控制縱軸  
 for(b=1; b<=a; b++)      (內迴圈)預設b值1~b<=a之間,控制橫軸

for(a=1; a<=9; a++)  
for(b=1; b<=a; b++)

a=1 ,b=1<=a                                                                印出"1"

a=2 ,b=1<=a ,b=2<=a                                                 印出"1" ,"2"

a=3 ,b=1<=a ,b=2<=a ,b=3<=a                                  印出"1" ,"2" ,"3"

2、

#include <iostream.h>

void main()

{

//練習陣列宣告

// dim a(59) as integer vb.net

int a[60];

//################3設定起始值-1

char b[5];

b[0]= 'h';

//################設定起始值--2

int c[5] = {45, 71, 91}; //沒有被設定到的值，為0

cout << c[0] << endl;

cout << c[3] << endl;

//##################設定起始值 -3 字串的 4 種設定方法

char a4[5] ={'h','a','p','p','y'};

char a5[] ={'h','a','p','p','y'}; //簡略寫法 -- 隱藏陣列大小

char a6[] = "happy";

char\* a7 = "happy";

///##################設定起始值--4 --設定字串值

char h[6] = "happy"; //注意字串設定除了使用{'h',...}外，也可用"happy"

//設定起始值為字串 ”…” 時，必須多保留一個位置

|  |
| --- |
| 說明：”ABCDE”共有5個字，可是必須宣告有6個元素。  原因：必須再多加上一個『空字元』=『\0』，其目的是在作為識別字元陣列結束的符號 |

//char h[] = "happy"; //也可使用隱藏陣列大小的宣告方法

//開始來比較5個數字的最大值

int k[5] ={35,67,94,63,20};

int max =0;

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

{

if(k[i]>max)

max=k[i];

}

cout << "最大值=" << max << '\n';

3、

#include <iostream.h>

void main()

{

int a[5] ={32,6,95,45,30};

int i;

//輸出排序前順序

cout << "排序前：";

for(i=0;i<=4;i++) //前面已經宣告過了 i，故在此不用宣告

cout << a[i] << '\0'; //使用\t，會造成間隔太大， \0代表一個空格

//開始進行排序

int temp=0;

int max=0;

for(i=4;i>0;i--) //注意，由大而小的遞減迴圈  
 { //注意，其號碼由3開始，乃是因為其將做到a[3]與a[4]的比較就停止了

for(int j=0;j<i;j++) //內層j迴圈，為一次兩兩相比的大循環

{ //第一次循環要做4次比較、第二次循環作3次比較  
 //第三次循環作2次比較、第四次循環作1次比較

//第五次作0次比較、故i的值由4--3--2--1--0

if(a[j]>a[j+1]) //若a[j]>a[j+1]，就將兩者的值互相交換

{

temp=a[j];

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

a[j+1]=temp;

}

}

}

//輸出排序後順序

cout << endl;

cout << "排序後：";

for(int k=0;k<=4;k++)

cout << a[k] << '\0';

cout << endl;

}