**TEST 10 指 针**

**测试 - 试题/答案**

【测试时间180分，自行批阅，计分并QQ告知 】

**学习和解题要点**

1. 内存空间和地址：C程序中定义的任何类型的变量，相当于在计算机内存中开辟存贮空间。

每个空间除了有名称（变量名）可放什么样的数值类型之外，还有一个地址的概念。如:

int i,j “为可放整型的名叫i和j的变量，则它们的地址分别为&i,&j。

一维数组的名称（数组名）即为地址，如int a[20]则a即为地址，也代表第一个元素a[0]的地址&a[0]，二维数组的int b[3][4]；数组名b也是地址，但它不能代表第0行第0列元素的地址。

对二维数组这里有二个层次的地址概念。二维数组b可以看成三个一维数组组成，它们的名称分别为b[0]，b[1]，b[2]。它们分别是&b[0][0]，&b[1][0] ，&b[2][0]三个第一个元素（每个是一个整数）的地址。

这时b可以看成大元素的（每个元素为4个整数）的数组的地址了，因此b就不是简单的一个整数的地址，也就是说b（也可写成b+0）代表了b[0]的地址，b+1代表了b[1]的地址。

|  |  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- | --- |
| b | b[0] |  |  |  |  |  |  |  |  |
|  | b[1] |  |  |  |  |  |  |  |  |
|  | b[2] |  |  |  |  |  |  |  |  |

以上讲的地址 &i，b[0]，b等都是常量，称为地址常量，一当定义就不可以再变的。

2．指针变量：简单地说就是放地址的变量，它是内存中一个空间，不放数据专放地址，不同时候可放不同地址，因此不是如前的地址常量而是可变的变量。

指针变量的定义如：有定义int i，\*p；p是一个变量的名称，但它不是一般的变量而是有个\* 特别标明以示与i的区别。因此这里 \* 是一个记号表示p与i不同，是指针变量。p可以放一个地址数但又不是任何的地址数，只可放整型数的地址，如p=&i；因此int \*p有三个意义：①说明p是变量名，②有 \* 说明p是指针变量名，③说明p只能放整型数的地址。

3．对指针变量的操作

赋值操作：通过求地址运算符 & 求出存储单元地址，赋给指针变量，或把一维数组名赋给指针变量如：

int i，a[10]，\*p，\*q；可以 p=&i 或 p=a；p=&a[0] 等，

二个指针变量之间赋值：q=p 表示q与p指向同一个地方，或该q跟着p指。

─→

&i

i

p

─→

p

a

q↑

当指针变量指向数组后，对指针变量可以比较和加减整数的运算。表示指针变量指向数组元素的前后和在数组元素上移动。

4．取指针所指的空间

指针运算符\*表示取出指针所指的空间（变量或数组元素）如int i=3，\*p；p=&i则 \*p即i，用 \*p输出时即为3。又如int a[4]={10,11,12,14},p=a，则\*p即为a[0]为10。

1. 指向动态分配的空间

C语言有一个函数malloc可在程序运行时动态的分配空间，其用法为：

p=(类型名 \*) malloc (sizeof（类型名）)。表示由sizeof（类型名）测出该类型所需的字节数，malloc申请这么多字节，同时把它强制类型转换成（类型名 \*）地址。p是指针变量，就指向这个地址。当然p应预先定义成这个类型的指针变量。

6．指向一维数组的指针的使用

当指针指向一维数组如：int \*p，a[10]；p=a；后，则p即与a等效，可用p[0],p[1],...,p[9]与a[0],a[1],...a[9]代表下标元素。也可用\*p ，\*(p+1)...\*(p+q)与\*a，\*(a+1)，...\*（a+9）代表下标元素。另外，与a不同的是 p可以为p=p+1或p++运算表示p在数组元素上“移动”。

1. 指向二维数组的指针的使用

变通的指针变量（我们称之为“小”指针），如int b[3][4]，\*p可以用来指向二维数组，这时把二维数组看成几个一维的数组一个一个的处理才可以。如p=b[0]或p=\*b，p=b[1]或p=\*(b+1)，这里不能写成p=b或p=b+1。 \*(b+1) 与b+1是地址相同概念不同的。

b+1代表第1行的地址，我们可称“行地址”，而\*(b+1)是第一行第0列元素的地址，我们可称“元素地址”，由于p是小指针因此只能用来指向元素地址 \*(b+1)，而不能指向b+1。这样取第j个元素可用 \*(p+j)，或 \*(p+i\*4+j)或 \*p++等。不能用\*(\*(p+i)+j)，p是小指针，+i只是走过i个元素而不是走i行。

专门指向数组的指针，我们称之为“大”指针，如int (\*p)[4]，int b[3][4];可以用来指向二维数组的整个一维如 p=b，p=b+1，而(\*p)[j]表示该行第j个元素了。另外还可以用p+i表示移i行，\*(p+i)表示该行首元素地址 \*\*(p+i)表示该行首元素值，\*(p+i)+j表示在该行位移j地址，\*(\*(p+i)+j)表示该行第j个元素值。

─→

p

b+i

─→

q

─→

p

i

1. 指针数组和多维指针

当定义成int \*p[4]；时，表示有4个指针，可以用来指向不同的4个整型变量或4个整型数组。使用时要一个一个的，如p[0]=&i，p[1]=&j等等。

当定义成int \*\*p，表示p是指针的指针，即可以指向指针变量，如int \*\*p,\*q,i q=&i,p=&q

1. 对于字符型的数组，用字符指针特别有意义。

当char str[20]，\*p；p=str；p就可以当成字符串使用，出现在任何字符串函数中。对于char \*p[4];这种指针数组就更加有意义，它表示4个字符指针，可以分别指向4个不同的字符数组（字符串）。

注意：char p[10]是10个字符的字符数组

char \*p[10]是10个指针的数组有10个字符型指针。

10．当指针变量指向一维数组时注意下面的区别：

\*p++ 等价于 \*(p++) 取 \*p的值，后p加1

(\*p)++ 取 \*p的值，后 \*p的值加1

\*(++p) 与 \*++p 等价，p自加，再取\*p值。

11. 指向结构的指针，有指向结构体的指针和指向结构数组的指针。

如：struct st { int sno; char sn[10]; } s, a[5], \*p;

指向结构体的指针相当于指针指向单个变量，用p=&s; 取结构体的成员用：p->sno

指向结构数组的指针相当于指针指向一维数组，用p=a; 可用p++或p+i取数组元素。

12．在使用指针变量中，要充分注意C语言并不给指针变量赋初值，因此程序中如果未给指针赋值，则称该指针变量为“无向指针”，使用这种指针是十分危险的。例如：int i=10,\*p； \*p=10,或\*p=i是错误的，因为p没有指向的空间。因此，10或i的值无法赋给\*p。同样如有：int a=10,b=20,\*p;交换a和b值写成：\*p=a;a=b;b=\*p;这里\*p也是没有指向空间的“无向指针”。同样，char \*str; scanf(“%s”,str); 也是错误的。

13. 指向数组的指针的例子

**例1:**

main()

{ static int a[10]={0,1,2,3,4,5,6,7,8,9};

int \*p, i;

printf("打印,移动指针:\n");

printf("数组 a 中的值:\n");

for(i=0;i<10;i++) printf("%d,",a[i]); printf("\n");

p=&a[4];

for(i=4;i<=6;i++,p++) printf("%d,",\*p); printf("\n");

printf("%d",\*p); printf("end p++\n");

printf("数组 a 中的值:\n");

for(i=0;i<10;i++) printf("%d,",a[i]); printf("\n");

p=&a[4];

for(i=4;i<=6;i++,p--) printf("%d,",\*p); printf("\n");

printf("%d",\*p); printf("end p--\n");

printf("边打印,边移动指针,不加括号:\n");

printf("数组 a 中的值:\n");

for(i=0;i<10;i++) printf("%d,",a[i]); printf("\n");

p=a;

for(i=0;i<10;i++) printf("%d,",\*p++); printf(" end \*p++\n");

printf("数组 a 中的值:\n");

for(i=0;i<10;i++) printf("%d,",a[i]); printf("\n");

p=&a[9];

for(i=0;i<10;i++) printf("%d,",\*p--); printf(" end \*p--\n");

printf("数组 a 中的值:\n");

for(i=0;i<10;i++) printf("%d,",a[i]); printf("\n");

p=a;

for(i=0;i<9;i++) printf("%d,",\*++p); printf(" end \*++p\n");

printf("数组 a 中的值:\n");

for(i=0;i<10;i++) printf("%d,",a[i]); printf("\n");

p=&a[9];

for(i=0;i<9;i++) printf("%d,",\*--p); printf(" end \*--p\n");

printf("边打印,边移动指针,加括号:\n");

printf("数组 a 中的值:\n");

for(i=0;i<10;i++) printf("%d,",a[i]); printf("\n");

p=a;

for(i=0;i<10;i++) printf("%d,",\*(p++)); printf(" end \*(p++)\n");

printf("数组 a 中的值:\n");

for(i=0;i<10;i++) printf("%d,",a[i]); printf("\n");

p=&a[9];

for(i=0;i<10;i++) printf("%d,",\*(p--)); printf(" end \*(p--)\n");

printf("数组 a 中的值:\n");

for(i=0;i<10;i++) printf("%d,",a[i]); printf("\n");

p=a;

for(i=0;i<9;i++) printf("%d,",\*(++p)); printf(" end \*(++p)\n");

printf("数组 a 中的值:\n");

for(i=0;i<10;i++) printf("%d,",a[i]); printf("\n");

p=&a[9];

for(i=0;i<9;i++) printf("%d,",\*(--p)); printf(" end \*(--p)\n");

printf("边打印,边使值变化:\n");

printf("数组 a 中的值:\n");

for(i=0;i<10;i++) printf("%d,",a[i]); printf("\n");

p=a;

for(i=0;i<10;i++,p++) printf("%d,",(\*p)++); printf(" end (\*p)++)\n");

printf("数组 a 中的值:\n");

for(i=0;i<10;i++) printf("%d,",a[i]); printf("\n");

p=a;

for(i=0;i<10;i++,p++) printf("%d,",(\*p)--); printf(" end (\*p)--\n");

printf("数组 a 中的值:\n");

for(i=0;i<10;i++) printf("%d,",a[i]); printf("\n");

p=a;

for(i=0;i<10;i++,p++) printf("%d,",++(\*p)); printf(" end ++(\*p)\n");

printf("数组 a 中的值:\n");

for(i=0;i<10;i++) printf("%d,",a[i]); printf("\n");

p=a;

for(i=0;i<10;i++,p++) printf("%d,",--(\*p)); printf(" end --(\*p)\n");

}

**例2:**

#include <stdio.h>

void main()

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

int i,j;

int (\*p)[4],\*q;//p是“大”指针，q是“小”指针

p=a; //p指向二维数组a中的一维

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

{ p=a+i; //p移动即移过一行

for(j=0;j<4;j++) printf("%d,", (\*p)[j]); //取这行的j元素

printf("\n");

}

p=a;

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

{ //p+i 第i行，\*(p+i)第i行首元素地址, \*(p+i)+j 位移j, \*(\*(p+i)+j)取元素

for(j=0;j<4;j++) printf("%d,", \*(\*(p+i)+j));

printf("\n");

}

q=a[0]; //q是“小”指针，指向首元素

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

{ q=\*(a+i); //q指向i行首元素

for(j=0;j<4;j++) printf("%d,", \*(q+j)); //移过q+j, 取元素\*(q+j)

printf("\n");

}

q=a[0]; //q指向首元素

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

{for(j=0;j<4;j++) printf("%d,", \*(q+i\*4+j)); //移过q+i\*4+j, 取元素\*(q+i\*4+j)

printf("\n");

}

q=a[0]; //q指向首元素

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

{ for(j=0;j<4;j++) printf("%d,", \*q++);//行列一个一个的移

printf("\n");

}

q=a[0];

for(i=0;i<12;i++) printf("%d,", \*q++); //一个一个的移，移12次

printf("\n");

for(q=a[0];q<a[0]+12;q++) printf("%d,", \*q); //用a[0] 到 a[0]+12控制

printf("\n");

}

**【试题** 总分109 **】**

1. 单项选择题 【25 X 1= 25 】
2. 对于同类型的指针变量，不可能进行的运算是（ ）。

A．- B．= C．+ D．==

2. 下列不正确的定义是（ ）。

A．int \*p=&i, i; B．int \*p, i; C．int i, \*p=&i; D．int i,\*p;

3. 下列正确的定义是（ ）。

A．int x=3,\*p=1; B．int x=3,\*P; C．int x=3,\*p=x; D．int \*p=&x=3;

4．设有定义“a=3，b，\*p=&a”，则下列语句中使b不为3的语句是（ ）。

A．b=\*&a; B．b=\*p; C．b=a; D．b=\*a;

5．设指针x指向的整型变量值为25，则printf(“%d\n”,++\*x);的输出是（ ）。

A．23 B．24 C．25 D．26

6．若有说明：“int i，j=7，\*p=&i；”则与i=j；等价的语句是（ ）。

A．i=\*p; B．\*p=\*&j; C．i=&j; D．i=\*\*p;

7．若有说明：“char ch，\*p；”则不能正确获得值的语句是（ ）。

A．\*p=getchar(); B．p=&ch; scanf(“%c”,p);

C．p=(char \*)malloc(1);\*p=getchar(); D．p=&ch; \*p=getchar();

1. 若有以下定义，且0<=i<5，则对数组元素错误的引用是（ ）。

int a[]={1,2,3,4,5},\*p=a,i;

A．\*(a+i) B．a[p-a] C．p+i D．\*(&a[i])

9．若有说明语句“int a[5]，\*p=a；”，对数组元素的正确引用是（ ）。

A．\*&a[5] B．\*p+2 C．\*(a+2) D．\*a++

10．若有说明语句“int a[5]，\*p=a；”，对数组元素的正确引用是（ ）。

A．a[p] B．p[a] C．\*(p+2) D．p+2

11. 若有以下定义，且0<=i<5，则对数组元素地址的正确表示是（ ）。

int a[]={1,2,3,4,5},\*p=a,i;

A．&(a+i) B．a++ C．&p D. &p[i]

12．下面各语句中，能正确进行赋字符串操作的语句是（ ）。

A．char s[5]={”ABCDE”}; B．char s[5]={‘A’, ‘B’, ‘C’, ‘D’, ‘E’};

C．char \*s; s=”ABCDE”; D．char \*s; scanf(“%s”,&s);

13. 若有以下定义，则\*(p+5)的值为（ ）。

char s[]=”Hello”, \*p=s;

A．’0’ B．’\0’ C．’0’的地址 D．不确定的值

14．若有以下定义，则不能表示a数组元素的表达式是（ ）。

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

A．\*p B．a[10] C．\*a D．a[p-a]

15．若有以下的定义，则值为3的表达式是（ ）。

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

A．p+=2，\*(p++) B．p+=2，\*++p C．p+=3，\*p++ D．p+=2，++\*p;

16．执行语句“char a[10]={“abcd”}，\*p=a”后，\*(p+4)的值是（ ）。

A．”abcd” B．’d’ C．’\0’ D．不能确定

17．若有定义“int a[3][4];”，则对数组元素a[i][j](0<=i<3,0<=j<4) 正确的引用是（ ）。

A．\*(a+4\*i+j) B．\*(\*(a+i)+j) C．\*(a+i)[j] D．a[i]+j

18．若有定义“int a[3][4],\*p;”，则对数组元素a[i][j](0<=i<3,0<=j<4) 正确的引用是（ ）。

A．p=a B．\*(\*(p+i)+j) C．\*(p+i\*4+j) D. \*(\*(a+i)+j)

19．若有定义“int a[3][4],(\*p)[3];”，则对数组元素a[i][j](0<=i<3,0<=j<4) 正确的引用是（ ）。

A．p=a[i] B．p=&a[i][j] C．\*(p+i\*4+j) D. \*(\*(p+i)+j)

20. 若有定义“int a[3][4],\*p,\*q[3];”，且0<=i<3,则错误的赋值语句是（ ）。

A．p=a B．q[i]=a[i] C．p=a[i] D. q[i]=&a[2][0]

21．设有定义语句“int (\*ptr)[10];”，其中的ptr是（ ）。

A. 10个指向整型变量的函数指针

B. 指向10个整型变量的函数指针

1. C. 一个指向具有10个元素的一维数组的指针

D. 具有10个指针元素的一维数组

22．若有以下定义，则数值为4的表达式是（ ）。

int w[3][4]={{0,1},{2,4},{0,1},{0,1}},(\*p)[4]=w;

A．\*w[1]+1 B．p++,\*(p+1) C．w[2][2] D．p[1][1]

23．若有下面的程序片段，则对数组元素的错误引用是（ ）。

int a[12]={0}, \*p[3], \*\*p, i;

for(i=0;i<3;i++) p[i]=&a[i\*4];

pp=p;

A．pp[0][1] B．a[10] C．p[3][1] D．\*(\*(+2)+2)

24．若有如下定义和语句，则输出结果是（ ）。

int \*\*pp, \*p, a=10, b=20;

pp=&p; p=&a; p=&b; printf(“%d,%d\n”,\*p,\*\*pp);

A．10,20 B．10,10 C．20,10 D．20,20

25．若有以下定义和语句，则对w数组元素的非法引用是（ ）。

int w[2][3], (\*pw)[3]; pw=w;

A．\*(w[0]+2) B．\*pw[2] C．pw[0][0] D．\*(pw[1]+2)

二、填充题【15 X 1= 15 】

1．C语言中，数组名是一个 常量，不能对它进行 和 运算。由 代表它的首地址。

1. 在C程序中，指针变量只能赋 值和 值。
2. 在C程序中，指针变量可以通过 ， ， 三种方式赋值。

4．在C程序中，可以对指针变量进行 和 来移动指针。

5．“\*”称为 运算符，“&”称为 运算符。

6．若两个指针变量指向同一个数组的不同元素，可以进行减法运算和 运算。

7．设int a[10],\*p=a；则对a[3]的引用可以是p[ ]和\*(p )。

8．若d是已定义的双精度变量，再定义一个指向d的指针变量p的语句是 。

9．&后跟变量名，表示该变量的 ，\*后跟指针变量名，表示该指计变量 ，&后指针变量名，表示该指针变量的 。

10．设有 int sz[4],\*p=sz; 有 ， ， 和 四种不同的引用数组元素的方法。

11．设有char \*a=”ABCD”; ，则printf(“%s”,a)的输出是 ；而printf(“%c”,\*a)的输出是 。

12．设有以下定义和语句，则的值为 。

int a[3][2]={10,20,30,40,50,60},(\*p)[2];

p=a;

13. 以下程序用指针指向三个整型存储单元，输入三个整数，选出其中最小值并输出，请填空。

#include <stdio.h>

main()

{ int ;

a=(int \*)malloc(sizeof(int));

b=(int \*)malloc(sizeof(int));

c=(int \*)malloc(sizeof(int));

min=(int \*)malloc(sizeof(int));

printf(“输入三个整数：\n”);scanf(“%d%d%d”, );

printf(“输出这三个整数：%d %d %d \n”, );

\*min=\*a;

if(\*a>\*b) ;

if( > \*c) ;

printf(“输出最小整数：%d\n”, );

}

14．以下程序的功能是从键盘上输入若干个字符（以回车键作为结束）组成一个字符串存入一个字符数组，然后输出该字符数组中的字符串。请填空。

# include “stdio.h”

#include “ctype.h”

main()

{ char str[81],\*sptr;

int i;

for(i=0;i<80;i++)

{ str[i]=getchar(); if (str[i]==’\n’) break; }

str[i]= ;

sptr=str;

while(\*sptr) putchar(\*sptr );

}

15. 以下程序从输入的10 个字符串中找出最长的那个串及长度。请填空。

#include <stdio.h>

#include <string.h>

main()

{ char str[10][80],\*sp; int i;

for(i=0;i<10;i++) gets(str[i]);

sp= str[0] ;

for(i=0;i<10;i++) if(strlen(sp)<strlen(str[i])) ;

printf(“输出最长的那个串: %s\n”, );

printf(“输出最长的那个串的长度: %d\n”, );

}

三、程序分析题【8 X 3= 24 】

1．阅读下列程序，写出程序的输出结果。

main()

{ char \*a[6]={“AB”, “CD”, “EF”, “GH”, “IJ”, “KL”};

int i;

for(i=0;i<4;i++) printf(“%s”,a[i]);

printf(“\n”);

}

2．阅读程序，写出程序的主要功能。

main()

{ int i, a[10], \*p=&a[9];

for(i=0;i<10;i++) scanf(“%d”,&a[i]);

for(;p>=a;p--) printf(“%d\n”,\*p);

}

3．阅读下列程序，写出程序运行的输出结果。

char s[]=”ABCD”;

main()

{ char \*p;

for(p=s;p<s+4;p++) printf(“%c %s\n”,\*p, p);

}

4．设有下列程序，试写出运行的结果。

main()

{ int i,b,c,a[]={1,10,-3,-21,7,13}, \*p\_b, \*p\_c;

b=c=1; p\_b=p\_c=a;

for(i=0;i<6;i++)

{ if(b<\*(a+i)) { b=\*(a+i); p\_b=&a[i]; }

if(c>\*(a+i)) { c=\*(a+i); p\_c=&a[i]; }

}

i=\*a; \*a=\*p\_b; \*p\_b=i; i=\*(a+5); \*(a+5)=\*p\_c; \*p\_c=i;

printf(“%d,%d,%d,%d,%d,%d\n”,a[0], a[1], a[2], a[3], a[4], a[5]);

}

5．设有下列程序，当输入字符串“LEVEL”和“LEVAL”时，试写出运行的结果。

#include <stdio.h>

#include <string.h>

void main()

{ char s[81],\*pi,\*pj;

int n;

gets(s); n=strlen(s);

pi=s; pj=s+n-1;

while(\*pi==' ') pi++;

while(\*pj==' ') pj--;

while((pi<pj)&&(\*pi==\*pj)) { pi++; pj--; }

if(pi<pj) printf("NO\n");

else printf("YES\n");

}

6. 阅读下列程序，写出程序运行的输出结果。

#include <stdio.h>

void main()

{ char \*alpha[4]={"ABCD","EFGH","IJKL","MNOP"};

char \*p; int i;

p=alpha[0];

for(i=0;i<4;p=alpha[++i]) printf("%c",\*(p));

printf("\n");

}

7. 阅读下列程序，写出程序运行的输出结果。

#include <stdio.h>

void main()

{ int s[4][4],i,j;

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

for(j=0;j<4;j++) \*(\*(s+i)+j)=i-j;

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

{ for(i=0;i<4;i++) printf("%4d", \*(\*(s+i)+j));

printf("\n");

}

}

8. 阅读下列程序，写出程序运行的输出结果。

#include <stdio.h>

void main()

{ static int a[4][4];

int \*p[4],i,j;

for(i=0; i<4; i++) p[i]=&a[i][0];

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

{ \*(p[i]+i)=1; \*(p[i]+4-(i+1))=1; }

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

{ for(j=0;j<4;j++) printf("%2d", p[i][j]);

printf("\n");

}

}

四、程序设计题（全部题目均要求用指针方法实现）【15 X 3= 45 】

1. 输入3个整数，按从大到小的次序输出。
2. 编一个程序，输入15个整数存入一维数组，再按逆序重新存放后再输出。
3. 输入一个字符串，按相反次序输出其中的所有字符。
4. 输入一个一维实型数组，输出其中的最大值、最小值和平均值。
5. 输入一个3×6的二维整型数组，输出其中最大值、最小值及其所在的行列下标。
6. 输入3个字符串，输出其中最大的字符串。
7. 输入2个字符串，将其连接后输出。
8. 比较2个字符串是否相等。
9. 输入10个整数，将其中最大数和最后一个数交换，最小数和第1个数交换。
10. 有10个整数，编一程序使它循环后移4个位置，再输出该整数。
11. 写一程序，将一个3×3的整数矩阵转置，并输出。
12. 写一程序，输入一字符串，输出其长度。
13. 写一程序，输入一行文字，找出其中大写字母、小写字母、空格、数字以及其他字符各有多少？
14. 将输入的一个字符串从第m个字符开始的全部字符复制到另一字符串并输出。
15. 把10个字符串从小到大排序并输出。

【答案 总分109 】

1. 单项选择题 【25 X 1= 25 】

|  |  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- | --- |
| 1 | 2 | 3 | 4 | 5 | 6 | 7 | 8 | 9 | 10 |
| C | A | B | D | D | B | A | C | C | C |
| 11 | 12 | 13 | 14 | 15 | 16 | 17 | 18 | 19 | 20 |
| D | C | B | B | A | C | B | C | D | A |
| 21 | 22 | 23 | 24 | 25 |  |  |  |  |  |
| C | D | C | D | C |  |  |  |  |  |

二、填充题【15 X 1= 15 】

1. 地址 加减 赋值 数组名

2. 地址 NULL

3. 取地址运算符 ， 相互赋值 ， 赋NULL

4. 加减整数 ++，--

5. 指针（取值） 取地址

6. 比较

7. 3 +3

8. double \*p=&a

9. 地址 指的变量 地址

10. sz[i] ， p[i] ， \*(sz+i) 和 \*(p+i)

11. ABCD A

12. 50

13. \*min,\*a,\*b,\*c a , b , c \*a, \*b, \*c \*min=\*b \*min \*min=\*c \*min

14. ’\0’ ++

15. sp=str[i] sp strlen(sp)

三、程序分析题【8 X 3= 24 】

1. 程序输出：ABCDEFGH

2. 程序的主要功能：把输入的10个整数反序一行一个的输出。

3. 程序运行的输出结果：A ABCD

B BCD

C CD

D D

4. 该程序找出数组中最大和最小的元素，分别与第一和最后的元素交换，运行结果输出：

13，10，-3，1，7，-21

5. 程序用pi和指针pj指向字符串s的头尾，然后逐步向中比较字符，到不等时终止。若都相等，则是“回文”，回答“YES”，否则回答“NO”。因此当输入字符串“LEVEL”时，输出“YES”， 当输入字符串 “LEVAL”时输出 “NO”。

6. 四个字符指针指向四个字符串，程序每次循环p指向一个字符串，输出第一的字符\*p，运行的输出结果: AEIM

7. 程序运行输出：0 1 2 3

-1 0 1 2

-2 -1 0 1

-3 -2 -1 0

8. 程序运行输出：1 0 0 1

0 1 1 0

0 1 1 0

1 0 0 1

四、程序设计题【15 X 3= 45 】

1. #include <stdio.h>

main()

{ int a, b, c, \*pa=&a, \*pb=&b, \*pc=&c;

scanf(“%d,%d,%d”,pa,pb,pc);

if(\*pa>=\*pb)

if(\*pb>\*pc) printf(“%d, %d, %d\n“,\*pa,\*pb,\*pc);

else if(\*pa>\*pc) printf(“%d, %d, %d\n“,\*pa,\*pc,\*pb);

else printf(“%d, %d, %d\n“,\*pc,\*pa,\*pb);

else if(\*pc>\*pb) printf(“%d, %d, %d\n“,\*pc,\*pb,\*pa);

else if(\*pa>\*pc) printf(“%d, %d, %d\n“,\*pb,\*pa,\*pc);

else printf(“%d, %d, %d\n“,\*pb,\*pc,\*pa);

}

2. #include <stdio.h>

#define N 15

main()

{ int a[N], b, \*p, \*q;

for(p=a;p<a+N;p++) scanf(“%d”,p);

for(p=a,q=a+N-1;p<a+N/2;p++,q--)

{ b=\*p; \*p=\*q; \*q=b; }

for(p=a;p<a+N;p++) printf(“%3d ”,\*p);

printf(“\n”);

}

3. #include <stdio.h>

#include <string.h>

main()

{ char str[81], \*sptr;

gets(str);

sptr=str+strlen(str)-1;

for(;sptr>=str;sptr--) printf(“%c”,\*sptr);

printf(“\n”);

}

4. #include <stdio.h>

#define N 10

main()

{ float a[N], avg, \*pm, \*ps, \*p;

for(p=a;p<a+N;p++) scanf(“%f”,p);

pm=ps=a; avg=\*a;

for(p=a+1;p<a+N;p++)

{ if(\*p>\*pm) pm=p;

if(\*p<\*ps) ps=p;

avg+=\*p;

}

printf(“一维实型数组最大值=%f\n”,\*pm);

printf(“一维实型数组最小值=%f\n”,\*ps);

printf(“一维实型数组平均值=%f\n”,avg/N);

}

5. #include <stdio.h>

main()

{ int a[3][6], (\*p)[6],i,j,maxh=0,maxl=0,minh=0,minl=0,max,min;

for(p=a,i=0;i<3;i++)

for(j=0;j<6;j++) scanf(“%d”,\*(p+i)+j);

printf(“二维数组是：\n”);

for(p=a,i=0;i<3;i++,p++)

{ for(j=0;j<6;j++) printf(“%3d ”,(\*p)[j]);

printf(“\n”);

}

max=min=a[0][0];

for(p=a,i=0;i<3;i++)

for(j=0;j<6;j++)

{ if(\*(\*(p+i)+j)>max) { maxh=i; maxl=j; max=\*(\*(p+i)+j); }

if(\*(\*(p+i)+j)<min) { minh=i; minl=j; min=\*(\*(p+i)+j); }

}

printf(“最大值是：%d 所在的行：%d 所在的列：%d\n”,a[maxh][maxl],maxh,maxl);

printf(“最小值是：%d 所在的行：%d 所在的列：%d\n”,a[minh][minl],minh,minl);

}

6. #include <stdio.h>

#include <string.h>

main()  
{ char s[3][81], \*p[3]={s[0],s[1],s[2]}, \*tp;

int i;

printf(“输入3个字符串: \n”);

for(i=0;i<3;i++) gets(p[i]);

tp=p[0];

for(i=1;i<3;i++) if(strcmp(p[i],tp)>0) tp=p[i];

printf(“其中最大的字符串是：%s\n”,tp);

}

7. #include <stdio.h>

#include <string.h>

main()

{ char a[40],b[40],c[80],\*s,\*t=c;

gets(a); gets(b);

s=a;

for(;\*s;) \*t++=\*s++;

s=b;

for(;\*s;) \*t++=\*s++;

\*t=’\0’;

puts(c);

}

8. #include <stdio.h>

#include <string.h>

main()

{ char a[81],b[81],\*s=a,\*t=b;

gets(a); gets(b);

whilw(\*s&&\*t) if(\*s++!=\*t++) break;

if(!\*s&&!\*t) printf(“%s 与 %s 相等\n”,a,b);

elas printf(“%s 与 %s 不相等\n”,a,b);

}

9. #include <stdio.h>

#define N 10

main()

{ int x[N],\*p=x, \*min, \*max, t1,t2;

printf("输入十个整数(用空格分开):\n");

for(p=a;p<a+N;p++) scanf("%d",p);

min=&a[0]; max=&a[N-1];

for(p=a;p<a+N;p++)

{ if (\*p<\*min) min=p;

if (\*p>\*max) max=p;

}

t1=\*min; \*min=a[0]; a[0]=t1;

t2=\*max; \*max=a[N-1];a[N-1]=t2;

for(p=a;p<a+N;p++) printf("%d ",\*p);

}

10. #include <stdio.h>

void main()

{ int x[100],i,n,m,end\_num,\*p;

printf("输入整数个数n(少于100个): ");scanf("%d",&n);

printf("输入%d个整数(用空格分开):\n",n);

for(p=x;p<x+n;p++) scanf("%d",p);

printf("这%d个整数原来是:\n",n);

for(p=x;p<x+n;p++) printf("%d ",\*p);

printf("\n");

printf("输入移动的位数m: \n"); scanf("%d",&m);

for(i=1;i<=m;i++){

end\_num=x[n-1];

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

x[0]=end\_num;

}

printf("这%d个整数移动后%d是:\n",n,m);

for(p=x;p<x+n;p++) printf("%d ",\*p);

printf("\n");

}

11. #include <stdio.h>

void main()

{ int a[3][3],i,j,t,\*p,(\*q)[3];

p=a[0];

printf("输入3×3个整数:\n");

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

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

scanf("%d",p++);

p=a[0];

printf("3×3的整数矩阵:\n");

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

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

printf("%3d ",\*(p+i\*3+j));

printf("\n");

}

q=a;

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

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

{ t=\*(\*(q+i)+j); \*(\*(q+i)+j)=\*(\*(q+j)+i); \*(\*(q+j)+i)=t;}

printf("转置后3×3的整数矩阵:\n");

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

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

printf("%3d ",\*(\*(q+i)+j));

printf("\n");

}

}

12. #include <stdio.h>

#include <string.h>

void main()

{ char a[80],\*p;

int count=0;

printf("输入一个字符串:\n"); gets(a);

p=a;

for(;\*p!='\0';p++) count++;

printf("这个的长度是：%d\n",count);

}

13. #include <stdio.h>

#include <string.h>

void main()

{ char a[80],\*p;

int up=0,lp=0,k=0,s=0,q=0;

printf("输入一个字符串:\n"); gets(a);

p=a;

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

if('A'<=\*p&&\*p<='Z') up++;

else if('a'<=\*p&&\*p<='z') lp++;

else if('0'<=\*p&&\*p<='9') s++;

else if(\*p==' ') k++;

else q++;

p++;

}

printf("大写字母个数是: %d\n",up);

printf("小写字母个数是: %d\n",lp);

printf("空格个数是: %d\n",k);

printf("数字个数是: %d\n",s);

printf("其他字符个数是: %d\n",q);

}

14. #include <stdio.h>

#include <string.h>

void main()

{ char a[80], b[80],\*p,\*q;

int m;

printf("输入一个字符串:\n"); gets(a);

printf("输入第几个字符开始复制: "); scanf("%d",&m);

p=a; q=b;

for(p=p+m-1; \*p!='\0'; p++,q++) \*q=\*p;

\*q='\0';

printf("被复制的字符串是：%s\n",b);

}

15. #include <stdio.h>

#include <string.h>

void main()

{ char a[10][80],t[80],\*p[10],\*k;

int i,j;

printf("输入10个字符串:\n");

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

{ p[i]=a[i]; gets(p[i]); }

printf("这10个字符串原来是:\n");

for(i=0;i<10;i++) puts(p[i]);

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

{ k=p[i];

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

if(strcmp(k,p[j])>0) k=p[j];

if(k!=p[i]) { strcpy(t,k); strcpy(k,p[i]); strcpy(p[i],t); }

}

printf("这10个字符串交换后:\n");

for(i=0;i<10;i++) puts(p[i]);

}