**1.** 在16位编译系统上 sizeof(int) =2，若有定义int a[]={10,20,30}, \*p=a;，当执行p++;后，下列说法错误的是（）

A）p向高地址移了一个字节 B）p向高地址移了一个存储单元

C）p向高地址移了两个字节 D）p与a+1等价

**2.**有以下程序段   
int a[10]={1,2,3,4,5,6,7,8,9,10},\*p=&a[3],b;  
b=p[5];  
b中的值是（）  
 A)5 B)6 C)8 D)9

**3.**若有以下定义，则对a数组元素的正确引用是\_\_\_\_\_\_\_\_\_.

int a[5],\*p=a;

a)\*&a[5] b)a+2 c)\*(p+5) d)\*(a+2)

**4.**若有以下定义，则p+5表示\_\_\_\_\_\_\_.

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

a)元素a[5]的地址 b)元素a[5]的值

c)元素a[6]的地址 d)元素a[6]的值

**5.**设已有定义: int a[10]={15,12,7,31,47,20,16,28,13,19}，\*p; 下列语句中正确的是（）

A) for(p=a; a<(p+10); a++);

B) for(p=a;p<(a+10);p++);

C) for(p=a,a=a+10;p<a;p++);

D) for(p=a;a<p+10; ++a);

**6.**有以下程序段   
#include <stdio.h>

int main()

{ int x[] = {10, 20, 30};

int \*px = x;

printf("%d,", ++\*px); printf("%d,", \*px);

px = x;

printf("%d,", (\*px)++); printf("%d,", \*px);

px = x;

printf("%d,", \*px++); printf("%d,", \*px);

px = x;

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

return 0;

}

程序运行后的输出结果是( )

A）11,11,11,12,12,20,20,20 B）20,10,11,10,11,10,11,10

C）11,11,11,12,12,13,20,20 D）20,10,11,20,11,12,20,20

**7.**设有如下定义：

int arr[]={6,7,8,9,10};

int \*ptr;

ptr=arr;

\*(ptr+2)+=2; 10

printf ("%d,%d\n",\*ptr,\*(ptr+2));

则程序段的输出结果为

A)8,10 B)6,8 C)7,9 D)6,10

**8.**若有定义:int a[]={2,4,6,8,10,12},\*p=a;则\*(p+1)的值是\_\_\_\_\_\_. \*(a+5)的值是\_\_\_\_\_\_\_\_\_.

**9.**若有以下说明和语句，int c[4][5],(\*p)[5];p=c;能正确引用c数组元素的是\_\_\_\_\_\_.

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

**10．**若有定义：int a[2][3],则对a数组的第i行j列元素地址的正确引用为\_\_\_\_\_\_.

a)\*(a[i]+j) b)(a+i) c)\*(a+j) d)a[i]+j

**11．**若有以下定义:int a[2][3]={2,4,6,8,10,12};则a[1][0]的值是\_\_\_\_\_. \*(\*(a+1)+0)的值是\_\_\_\_\_\_\_\_.

**12.**有以下定义  
char a[10],\*b=a;  
不能给数组a输入字符串的语句是（）  
A)gets(a) B)gets(a[0]) C)gets(&a[0]); D)gets(b);