**一、选择题**

1. 变量的指针，其含义是指该变量的\_\_\_b\_\_\_\_\_\_.

a)值 b)地址

c)名 d)一个标志

2．若有语句int \*point,a=4;和point=&a;下面均代表地址的一组选项是\_\_\_d\_\_.

a)a,point,\*&a b)&\*a,&a,\*point

c)\*&point,\*point,&a d)&a,&\*point ,point

3．若有说明;int \*p,m=5,n;以下正确的程序段的是\_\_\_\_c\_\_\_\_.

a)p=&n; b)p=&n;

scanf("%d",&p); scanf("%d",\*p);



c)scanf("%d",&n); d)p=&n;

\*p=n; \*p=m;



4. 以下程序中调用scanf函数给变量a输入数值的方法是错误的，其错误原因是\_\_\_b\_\_\_\_\_.

main()

{

int \*p,\*q,a,b;

p=&a;

printf(“input a:”);

scanf(“%d”,\*p); //a



……

}

a)\*p表示的是指针变量p的地址

b)\*p表示的是变量a的值，而不是变量a的地址

c)\*p表示的是指针变量p的值

d)\*p只能用来说明p是一个指针变量

5.有以下程序

#include<stdio.h>

main()

{ int m=1,n=2,\*p=&m,\*q=&n,\*r;

r=p;p=q;q=r;

printf("%d,%d,%d,%d\n",m,n,\*p,\*q);

}

程序运行后的输出结果是



A）1，2，1，2 B）1，2，2，1

C）2，1，2，1 D）2，1，1，2

6. 有以下程序

main()

{ int a=1, b=3, c=5;

int \*p1=&a, \*p2=&b, \*p=&c;



\*p =\*p1\*(\*p2);



printf("%d\n",c);

}

执行后的输出结果是

A）1 B）2 C）3 D）4



7. 有以下程序

main()

{ int a,k=4,m=4,\*p1=&k,\*p2=&m;



a=p1==&m;



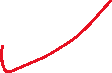
printf("%d\n",a);



}

程序运行后的输出结果是（）

A）4 B）1 C）0 D）运行时出错，无定值



8．设p1和p2是指向同一个字符串的指针变量，c为字符变量，则以下不能正确执行的赋值语句是\_\_\_\_\_\_\_\_.

a)c=\*p1+\*p2; b)p2=c c)p1=p2 d)c=\*p1\*(\*p2);



9. 某32位系统下，请计算sizeof 的值.



char str[] = “http://www.ibegroup.com/”



char \*p = str ;



int n = 10;

请计算

sizeof (str ) = ？ （1）



sizeof ( p ) = ？ （2）

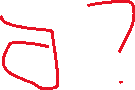
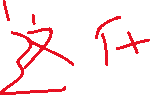
sizeof ( n ) = ？ （3）



10. int main()

{

int x=258; 256+2



char \*p = (char \*)&x;

printf("%d ",\*p);

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

}

程序运行结果是\_\_\_\_258 \_\_\_\_\_\_\_\_\_\_\_\_\_\_

1.程序分析

1.

int \*ptr;

printf(“%d\n”,\*ptr);

指针变量 ptr 没有初始化，它的值是不确定的

2.

int \*ptr=null;

printf(“%d\n”,\*ptr);

null 是一个空指针常量，指针变量 ptr 被赋值为 null在解引用空指针时，会产生运行时错误，因为空指针没有指向有效的内存地址。

3.

int \*ptr;

\*ptr=10;

printf(“%d\n”,\*ptr);

指针变量 ptr 没有分配内存空间，它的值是不确定的。在尝试修改未分配内存的地址时，会产生运行时错误，因为没有有效的内存来存储值。

4.

char \*s1 = “abc”; char \*s2=”xyz”; \*s1 = \*s2；

上述各程序中，各存在什么问题？

字符串字面量是只读的，尝试修改字符串字面量的值是非法的。指针 s1 和 s2 分别指向字符串常量 "abc" 和 "xyz"，它们不能被修改。因此，对 \*s1 赋值 \*s2 会导致运行时错误。 等号前不能是常量

2.局部变量与全局变量同名时，如何在局部访问到全局变量的值。(运用指针)

在函数 内部中，我们通过定义一个指针int \* ptr，并将其初始化为全局变量的地址 &全局变量，来访问全局变量的值。在 main() 函数中，我们调用函数来在局部访问到全局变量的值。

3.思考？

const int \*p;

int const \*p; const int \*p; 或 int const \*p;

这两个声明都表示 p 是一个指向整型常量的指针。意味着 p 所指向的值是不可修改的，但指针本身可以改变。这种声明称为指向常量的指针。

int \* const p;

int \* const p;

这个声明表示 p 是一个指向整型的常量指针。意味着 p 的指针值不可更改，但可以通过 p 修改所指向的值。这种声明称为常指针。

void \*p; 这个声明表示 p 是一个指向 void 类型的指针即通用指针。通用指针可以指向任何类型的数据，但不能直接解引用。需要进行类型转换才能操作。

分别是什么意思？动手测试一下。