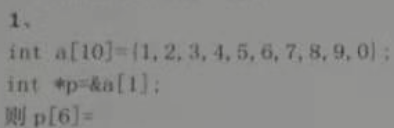
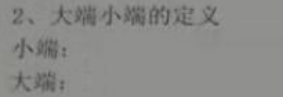
1、



8

2、



大端：数据存储时数据高位对应低地址，数据低位对应高地址

小端：数据存储时数据高位对应高地址，数据低位对应低地址

3、





int a;



int \*a;



int \*\*a;



int a[10] ;



int \*a[10];



Int (\*a)[10];

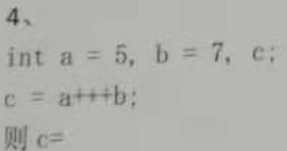


int (\*a)(int );



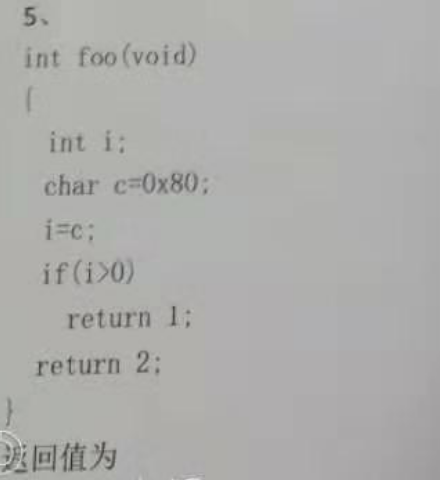
void (\*a[10])(int);

4、



12（c=a+++b等价于c=(a++)+b）

5、



返回值为2

6、

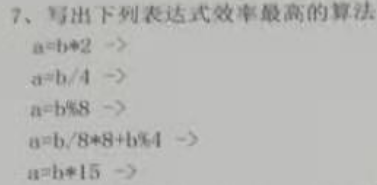


int \*p = NULL;

P = (int \*)0x67a9;

\*p = 0xaa66;

7、



a=b\*2 -> a=b<<1;

a=b/4 -> a = b>>2;

a=b%8 -> a=b&0b111;

a=b/8\*8+b%4 -> a = (b>>3)<<3 + (b&0b11)

a=b\*15 ->

8、



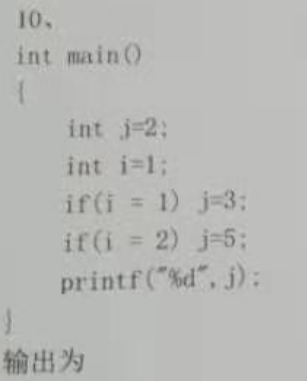
#define MIN((A),(B)) (A)<(b)?(A):(B)

9、



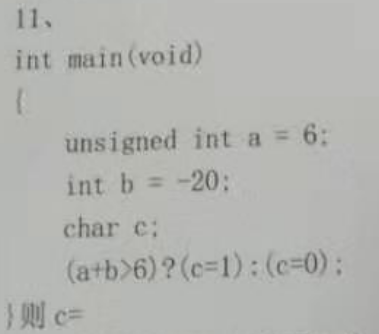
#define NUM (60\*60\*24\*365)UL

10、



5

11、



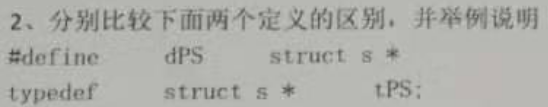
1



1、



2、



#define dPS struct s \* //宏定义（单纯替换）

typedef struct s \* tPS //重定义

dPS a,b;

struct s \* a,b;

a->struct s \*

b->struct s

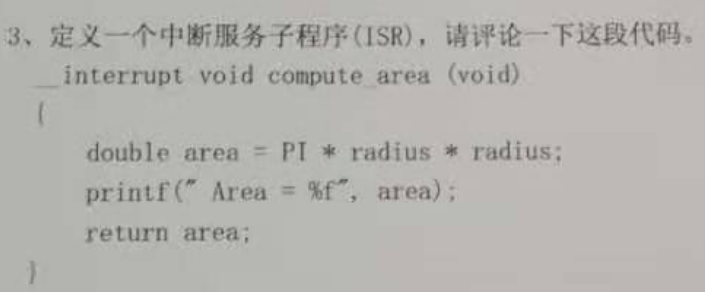
tPS c,d; ->

struct s \* c,d;

c->struct s \*

d->struct s \*

3、



4、



static：

①static本质是延长变量或函数的生命周期，并且限制其作用域。

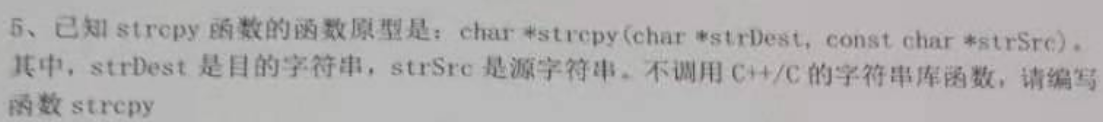
②static声名的全区变量、函数，仅当前文件内可用，其他文件不能引用。

③static在函数内部声名的内部静态变量，只需初始化一次。

④变量存储在全局数据段中，而不是栈中，其生命周期持续到程序退出。

volatile：

5、



char \*strcpy(char \*strDest, const char \*strSrc)

{

char \*src = strSrc;

char \*dest = strDest;

while(\*src != '\0')

{

if(\*dest!=\*src)

\*dest=\*src;

dest++;

src++;

}

\*dest='\0';

return strDest;

}

6、

