

下列程序段的运行结果是

```
int i=1,a[]={1,5,10,9,13,7};
while(a[i]<=10)
    a[i++]+=2;
for (i=0;i<6;i++)
    printf("%d",a[i]);
```

对字符数组 s 赋值,不合法的一个是

- A) char s[]="Beijing"; B) char s[20]={"beijing"};
C) char s[20]; s="Beijing"; D) char s[20]={'B','e','i','j','i','n','g'};

若有 typedef float AR[5];AR a;则以下叙述正确的是

- A) a 是一个新类型名
B) AR 是一个实型数组
C) AR 是一个实型变量
D) a 是一个实型数组
-

下面程序的功能是：从键盘上输入一行字符，存入一个字符数组中，然后输出该字符串。

```
char str[81];
int i;
for(i=0; i<80; i++)
{
    str[i]=getchar( );
    if(str[i]=='\n')
        break;
}
str[i]= ①;
```

```
for(i = 0; ② i++)  
    putchar(str[i]);
```

下面程序的输出结果是

```
char str[100] = "How do you do";  
strcpy( str + strlen(str)/2, "es she");  
printf("%s\n", str);
```

A) How do you do B) es she C) How are you D) How does she

下面程序段的运行结果是 () 。

```
int i,j;  
char str[ ]="1234567",k;  
for ( i=1, j=strlen(str)-2; i<j; i++, j--)  
{  
    k=str[i]; str[i]= str[j]; str[j]=k;  
}  
puts(str);
```

下面程序段的运行结果是 () 。

```
char c[5]={ 'a','b','\0','c','\0'}; printf("%s",c);  
A) 'a''b'    B) ab    C) ab c
```

以下语句把字符串“abcde”赋初值给字符数组，不正确的语句是

- A) char s[]="abcde";
- B) char s[]={ 'a','b','c','d','e','\0'};
- C) char s[]={ "abcde"};
- D) char s[5]= "abcde";

下面程序段的运行结果是

```
char s[]="ABC";
```

```
int i;  
for(i=0;i<3;i++)  
    printf("%s",&s[i]);
```

已知: `char s[100]; int i = [0,89];`则引用数组元素的错误的形式是

A) `s[i+10]` B) `*(s+i)` C) `*(i+s)` D) `*((s++)+i)`

地址进行不能自增自减运算

若已定义 `char s[2][40]={“china”,“English”};` 则在下面表达式中能表示字符‘E’的地址的是 ()。

A) `&s[2][1]` B) `&(s[1]+1)` C) `s[1]` D) `&s[1]`

若有定义 `int a[3][5], i, j;` (且 $0 \leq i < 3, 0 \leq j < 5$) ,则 `a[i][j]`的地址不正确表示是 A)

`&a[i][j]`

B) `a[i]+j`

C) `*(a+i)+j`

D) `*(*(a+i)+j)`

若有以下的定义: `int t[3][2];`能正确表示 `t` 数组元素地址的表达式是

A) `&t[3][2]` B) `t[3]` C) `t[1]` D) `t[2]`