1. 下面的代码是反转一个字符串，指出其中的错误。 (7.14)

```
char s[] = "ABCD";
for (int i=0, j=sizeof(s)-1; i<sizeof(s); i++, j--)
{
    char temp = s[i];
    s[i] = s[j];
    s[j] = temp;
}
cout << "The reversed string is " << s << endl;
```

2. 下面代码中的 s1 和 s2 有什么区别? (7.18)

```
char s1[] = {'a', 'b', 'c'};
char s2[] = "abc";
```

3. 已有语句：char s1[]="abc"; char s2[]="efg"; 下面的表达式是否正确？ (7.19)

```
a) s1 = "good"
b) s1 < s2
c) s1[0]
d) s1[0] < s2[0]
e) strcpy(s1, s2)
f) strcmp(s1, s2)
g) strlen(s1)
```

4. 下面的代码有什么错误? (11.2)

```
int x = 30;
int* pX = x;
cout << "x is " << x << endl;
cout << "x is " << pX << endl;
```

5. 下面代码的输出结果是什么？ (11.3)

```
int x = 30;
int* p = &x;
cout << *p << endl;
int y = 40;
p = &y;
cout << *p << endl;
```

6. 下面的代码有什么错误? (11.6)

```
double x = 3.0;
int* pX = &x;
```

7. 下面代码的输出结果是什么？ (11.5)

```
char s[] = "ABCD";
char* p = s;
cout << p << endl;
cout << *p << endl;
```

8. 下面代码中声明的 p1 和 p2 都是指针吗？ (11.7)

```
double* p1, p2;
double* p1, *p2;
```

9. 下面的代码有什么错误？ (11.9)

```
int x;
int* const p = &x;
int y;
p = &y;
```

10. 下面的代码有什么错误? (11.10)

```
int x;
const int* p = &x;
int y;
p = &y;
*p = 5;
```

11. 已有语句：int* p，且 p 的值为 100，则 p + 1 的值是什么？ (11.11)

12. 已有语句：int* p，则 p++, *p++ 和 (*p)++ 的区别是什么？ (11.12)

13. 已有语句：int p[4]={1,2,3,4}，则 *p, *(p+1), p[0] 和 p[1] 的值分别是什么？ (11.13)

14. 下面的代码有什么错误？ (11.14)

```
char* p;
cin >> p;
```

15. 下面语句的输出结果是什么？ (11.15)

```
char* const pCity = "Dallas";
cout << pCity << endl;
cout << *pCity << endl;
cout << *(pCity + 1) << endl;
cout << *(pCity + 2) << endl;
cout << *(pCity + 3) << endl;
```

16. 下面语句的输出结果是什么？ (11.16)

```
char* city = "Dallas";
cout << city[0] << endl;
char* cities[] = {"Dallas", "Atlanta", "Houston"};
cout << cities[0] << endl;
cout << cities[0][0] << endl;
```

17. 已知一个数组，数组名为 x，试用一条语句算出该数组的元素个数。
（提示：使用 sizeof 函数）

18. 阅读下面的代码

```
int a[]={6,21,12,34,55};
int * pa = a, * pb;
*pa = 5;
*(pa+2) = 8;
*(pa++) = *a + 42;
pb = pa;
*(++pb) = 45;
```

指出最后 a，*pa 和 *pb 的值。