

# 实验7

## 1 (10分)

利用字符指针将输入的一个字符串中的大小写字母相互转换，并输出转换后的字符串的内容。

如，假设输入的字符串的内容为“How are you”，则转换后的内容为“hOWARE YOU”

```
1 #include "iostream"
2 using namespace std;
3
4 int main() {
5     char str[100]; // 定义一个字符数组存储输入的字符串
6     char *p = str; // 指向字符串的指针
7
8     cout << "请输入一个字符串: ";
9     cin.getline(str, 100); // 获取输入的字符串
10
11     // 遍历字符串，进行大小写转换
12     while (*p != '\0') { // 当指针没有指向字符串的结尾'\0'时
13         // 判断字符是否为大写字母
14         if (*p >= 'A' && *p <= 'Z') {
15             *p = *p + 32; // 将大写字母转换为小写
16         }
17         // 判断字符是否为小写字母
18         else if (*p >= 'a' && *p <= 'z') {
19             *p = *p - 32; // 将小写字母转换为大写
20         }
21         p++; // 移动指针到下一个字符
22     }
23
24     // 输出转换后的字符串
25     cout << "转换后的字符串: " << str << endl;
26
27     system("pause");
28     return 0;
29 }
```

代码说明：

### 1. 字符判断：

- 使用条件判断 `if (*p >= 'A' && *p <= 'Z')` 判断当前字符是否为大写字母。

- 使用条件判断 `else if (*p >= 'a' && *p <= 'z')` 判断当前字符是否为小写字母。

## 2. 大小写转换：

- 如果字符是大写字母（ASCII 码在 65 到 90 之间），则通过加上 32 将其转换为小写字母。
- 如果字符是小写字母（ASCII 码在 97 到 122 之间），则通过减去 32 将其转换为大写字母。

## 3. 指针操作： `p++` 用来指向下一个字符，直到遇到字符串结尾符 `\0`。

D:\2024\课内\C++\第七章\7.1\Debug\7.1.exe

```
请输入一个字符串: how are you
转换后的字符串: HOW ARE YOU
请按任意键继续. . .
```

## 2 （10分） 利用字符指针将字符串s中从第n个字符开始的内容复制到字符串t中

```
1 #include "iostream"
2 using namespace std;
3
4 int main() {
5     char s[100], t[100]; // 定义字符串数组 s 和 t
6     char *ps = s, *pt = t; // 字符指针，分别指向 s 和 t
7     int n;
8
9     // 输入字符串 s 和要复制的起始位置 n
10    cout << "请输入字符串s: ";
11    cin.getline(s, 100); // 输入字符串 s
12    cout << "请输入起始位置n(从1开始): ";
13    cin >> n;
14
15    // 将指针 ps 移动到第 n 个字符（注意数组从 0 开始）
16    ps = s + (n - 1);
17
18    // 从第 n 个字符开始复制到字符串 t
19    while (*ps != '\0') { // 遍历直到字符串结尾
20        *pt = *ps; // 将当前字符复制到 t 中
21        ps++; // 移动到下一个字符
22        pt++; // 移动到 t 中的下一个位置
23    }
24    *pt = '\0'; // 给字符串 t 加上结束符
25
26    // 输出复制后的字符串 t
```

```

27     cout << "复制到字符串t的内容: " << t << endl;
28
29     system("pause");
30     return 0;
31 }

```

## 代码说明：

### 1. 输入字符串和位置：

- 使用 `cin.getline()` 输入字符串 `s`。
- 输入要复制的起始位置 `n`（假设从 1 开始计数）。

### 2. 移动指针到第 `n` 个字符：

- `ps = s + (n - 1)` 将指针 `ps` 移动到 `s` 中的第 `n` 个字符位置。
- 注意：数组是从下标 `0` 开始的，所以要用 `n-1` 来定位第 `n` 个字符。

### 3. 字符复制：

- 使用 `while (*ps != '\0')` 遍历字符串，从第 `n` 个字符开始逐个复制到目标字符串 `t`。
- `*pt = *ps` 完成字符复制，然后分别移动源指针 `ps` 和目标指针 `pt`。

### 4. 添加字符串结束符：

- 在复制结束后，通过 `*pt = '\0'` 标记字符串 `t` 的结束位置。

### 5. 输出结果：

- 最终输出字符串 `t`。

```

D:\2024\课内\C++\第七章\7.2\Debug\7.2.exe
请输入字符串s: dbubo
请输入起始位置n(从1开始): 2
复制到字符串t的内容: bubo
请按任意键继续. . .

```

## 3 (10分)

利用指针将一个包含10个整数的数组中的最大最小元素进行交换，并输出交换后的内容。10个整数为随机产生的0~100之间的数。（若最大最小元素不唯一，则交换任意两个都算正确，即可以不用特殊考虑不唯一的情况）

```

1 #include "iostream"
2 #include "ctime"    // 用于随机数生成
3 using namespace std;
4
5 int main() {
6     int arr[10];      // 定义一个包含10个整数的数组
7     int *p = arr;     // 定义指针指向数组
8     int *pMax = arr, *pMin = arr; // 定义指针用于指向最大和最小值
9
10    // 随机数种子
11    srand(time(0));
12
13    // 随机生成数组的10个元素
14    cout << "原数组内容: ";
15    for (int i = 0; i < 10; i++) {
16        *(p + i) = rand() % 101; // 生成 0~100 的随机数
17        cout << *(p + i) << " ";
18    }
19    cout << endl;
20
21    // 找到最大和最小值的指针
22    for (int i = 1; i < 10; i++) {
23        if (*(p + i) > *pMax) { // 找最大值
24            pMax = p + i;
25        }
26        * if (*(p + i) < *pMin) { // 找最小值
27            pMin = p + i;
28        }
29    }
30
31    // 交换最大值和最小值
32    int temp = *pMax;
33    *pMax = *pMin;
34    *pMin = temp;
35
36    // 输出交换后的数组内容
37    cout << "交换后数组内容: ";
38    for (int i = 0; i < 10; i++) {
39        cout << *(p + i) << " ";
40    }
41    cout << endl;
42
43    system("pause");
44    return 0;
45 }

```

## 代码说明：

### 1. 随机数生成：

- 使用 `srand(time(0))` 设置随机数种子。
- 使用 `rand() % 101` 生成 `[0, 100]` 范围内的随机整数。

### 2. 指针遍历：


- 使用指针 `p` 遍历数组，比较找到最大值和最小值的指针位置 `pMax` 和 `pMin`。

### 3. 交换元素：

- 交换两个元素的值，使用一个临时变量 `temp`。

### 4. 输出结果：

- 输出数组内容，交换前后打印两次。



```
D:\2024\课内\C++\第七章\7.3\Debug\7.3.exe
原数组内容: 76 60 74 27 76 29 97 71 34 33
交换后数组内容: 76 60 74 97 76 29 27 71 34 33
请按任意键继续. . .
```