

课程名称	软硬件接口程序设计	课程编号	A2130731
实验地点	重庆邮电大学综合实验楼	实验时间	2022. 10. 13
实验名称	实验二 字符串与字符		

一、 实验目的

练习 C 语言中字符串数组和指针的使用，学习 C 语言中递归的使用，学习 C 语言中 I/O 流。

二、实验内容

Task1:

1. 根据 example1 给出的代码回答以下问题：

(1) 为什么代码中的 scanf 不需要&？

(2) 将给出的 strings 输入后，将得到什么输出？

2. 如果将给出的 Strings 输入 example2 后，将得到什么输出？

3. 根据 example3 回答以下问题：

(1) 为什么需要小心地使用 gets()

(2) 将给出的 strings 输入后，将得到什么输出？

Task2: 学习如何使用递归的方法实现字符串反转。讨论递归的利与弊，并使用指针运算重写给出的程序。

Task3:使用 scanf 完成输入后，完成以下任务

1. 给出一个用户输入的字符串，在不使用标准函数的情况下计算该字符串中大写和小写字母的数量。不使用标准函数并打印结果。

2. 给出两个用户输入的字符串，在不使用库函数的情况下将两个字符串连接起来，并显示结果。

3. 给出一个用户输入的字符串，用指针计算字符串的长度，并显示结果。

### 三、实验过程原始记录(数据、图表等)

#### Task1:

```
root@h9-virtual-machine:/home/h9/ICSI333/Lab/Lab2# ./task1-1
Enter name: Michael Jordan
Your name is Michael.
root@h9-virtual-machine:/home/h9/ICSI333/Lab/Lab2# ./task1-1
Enter name: Michael Jordan Jordan
Your name is Michael.
root@h9-virtual-machine:/home/h9/ICSI333/Lab/Lab2# ./task1-1
Enter name: MichaelJordan
Your name is MichaelJordan.
```

1:

```
root@h9-virtual-machine:/home/h9/ICSI333/Lab/Lab2# ./task1-2
Enter name: Michael Jordan
Name: Michael Jordan
root@h9-virtual-machine:/home/h9/ICSI333/Lab/Lab2# ./task1-2
Enter name: Michael Jordan Jordan
Name: Michael Jordan Jordan
root@h9-virtual-machine:/home/h9/ICSI333/Lab/Lab2# ./task1-2
Enter name: MichaelJordan
Name: MichaelJordan
```

2:

```
root@h9-virtual-machine:/home/h9/ICSI333/Lab/Lab2# ./task1-3
Enter name: Michael Jordan
Name: Michael Jordan
root@h9-virtual-machine:/home/h9/ICSI333/Lab/Lab2# ./task1-3
Enter name: Michael Jordan Jordan
Name: Michael Jordan Jordan
root@h9-virtual-machine:/home/h9/ICSI333/Lab/Lab2# ./task1-3
Enter name: MichaelJordan
Name: MichaelJordan
```

3:

#### Task2:

```
root@h9-virtual-machine:/home/h9/ICSI333/Lab/Lab2# ./task2

Enter the string :Michael Jordan Jordan

Reverse string is :nadroJ nadroJ leahciM
```

#### Task3:

```
root@h9-virtual-machine:/home/h9/ICSI333/Lab/Lab2# ./task3

Enter the string to calculate uppercases and lowercases :Hello
The number of uppercase and lowercase are 1 and 4

Enter two strings to concat :Hello
World!
After concating:HelloWorld!

Enter string to calculate length :HelloWorld!
The length of string is: 11
```

### 四、实验结果及分析

实验结果可见原始数据记录。相应分析如下:

### Task1:

1. (1) 不需要`&`，因为 `char` 类型数组本身就是指针，指向了首元素的地址。

1. (2) 结果为：

Michael

Michael

MichaelJordan

符合运行记录，其原因为 `scanf` 遇到空格自动停止。

2. 结果为：

Michael Jordan

Michael Jordan Jordan

MichaelJordan

符合运行记录。因为程序不再使用 `scan` 进行输入，而使用 `while` 循环和 `getchar()` 进行读取，直到遇到 `'\n'` 换行符终止。

3. (1) `gets` 不会进行边界检查，若字符串长度大于剩余空间，会造成越界。

3. (2) 结果为：

Michael Jordan

Michael Jordan Jordan

MichaelJordan

符合运行记录。因为 `gets()` 遇到空格不会自动停止，故能实现完整读入。

**Task2:** 构建头指针和尾指针，通过使用指针运算来同时移动头指针和尾指针，并交换指针指向地址的内容，直到二者位置重合。

### Task3:

1. 构建 `UpperLowerLen()` 函数，使用 `for` 循环遍历字符串，通过比较 ASCII 码大

小，来统计大小写字母个数。如 $\geq a$  且  $\leq z$  时，小写字母个数加一，当 $\geq A$  且  $\leq Z$  时大写字母个数加一。

2. 构建 `ConcatStr()`函数，通过遍历字符串 2 的内容，将其赋值在字符串 1 后对应的位置，实现连接字符串。

3. 构建 `my_strlen()`函数，使用指针以及指针运算遍历字符串直到结束标识，来统计字符串长度。

## 五、实验心得体会

1.熟悉并掌握了 C 语言中的字符数组、字符、指针、递归方法。

2.熟悉并掌握了 C 语言中 `Scanf`，`gets` 等不同 I/O 流方法的区别。