

对于传参类型 `char*`，直接放入字符数组的数组名即可。

接下来我们通过例 4.3.2 来具体学习 `str` 系列字符串操作函数，掌握每个函数的内部实现。

【例 4.3.2】`str` 系列字符串操作函数的使用。

```
#include <stdio.h>
#include <stdlib.h>
#include <string.h>

int mystrlen(char c[])
{
    int i=0;
    while(c[i++]);
    return i-1;
}
//strlen 统计字符串长度
int main()
{
    int len; //用于存储字符串长度
    char c[20];
    char d[100]="world";
    while(gets(c)!=NULL)
    {
        puts(c);
        len=strlen(c);
        printf("len=%d\n",len);
        len=mystrlen(c);
        printf("mystrlen len=%d\n",len);
        strcat(c,d);
        strcpy(d,c); //c 中的字符串复制给 d
        puts(d);
        printf("c?d %d\n",strcmp(c,d));
        puts(c);
    }
    system("pause");
    return 0;
}
```

通过 `gets` 函数循环读取字符串的目的是，方便大家不断地输入不同的字符串并查看程序的执行效果，并在修改程序中的某部分后能够以多种输入进行测试。如果要结束循环，那么可以按组合键 `Ctrl+Z`。图 4.3.5 所示为我们输入“hello”后的执行结果，通过 `strlen` 函数计算的字符串长度为 5，我们自己写的函数就是 `strlen` 函数的计算原理，即通过判断结束符来确定字符串的长度。

`strcpy` 函数用来将字符串中的字符逐个地赋值给目标字符数组。例中我们将 `c` 复制给 `d`，就是将 `c` 中的每个字符依次赋值给 `d`，也会将结束符赋值给 `d`。注意，目标数组一定要大于字符串大小，即 `sizeof(d)>strlen(c)`，否则会造成访问越界。