2019/3/17 C 字符串 | 菜鸟教程

◆ C 指针

C 结构体 →

C字符串

在 C 语言中,字符串实际上是使用 **null** 字符 '\0' 终止的一维字符数组。因此,一个以 null 结尾的字符串,包含了组成字符串的字符。

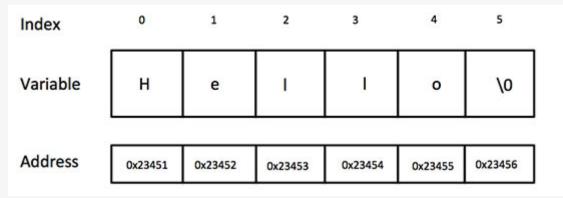
下面的声明和初始化创建了一个 "Hello" 字符串。由于在数组的末尾存储了空字符,所以字符数组的大小比单词 "Hello" 的字符数多一个。

```
char greeting[6] = {'H', 'e', 'l', 'l', 'o', '\0'};
```

依据数组初始化规则,您可以把上面的语句写成以下语句:

```
char greeting[] = "Hello";
```

以下是 C/C++ 中定义的字符串的内存表示:



其实,您不需要把 *null* 字符放在字符串常量的末尾。C 编译器会在初始化数组时,自动把 '\0' 放在字符串的末尾。让我们尝试输出上面的字符串:

实例

```
#include <stdio.h>
int main ()
{
    char greeting[6] = {'H', 'e', 'l', 'o', '\0'};
    printf("Greeting message: %s\n", greeting );
    return 0;
}
```

当上面的代码被编译和执行时,它会产生下列结果:

```
Greeting message: Hello
```

C 中有大量操作字符串的函数:

```
序号 函数 & 目的
```

```
1
    strcpy(s1, s2);
    复制字符串 s2 到字符串 s1。
2
    strcat(s1, s2);
    连接字符串 s2 到字符串 s1 的末尾。
3
    strlen(s1);
    返回字符串 s1 的长度。
4
    strcmp(s1, s2);
    如果 s1 和 s2 是相同的,则返回 0;如果 s1<s2 则返回小于 0;如果 s1>s2 则返回大于 0。
5
    strchr(s1, ch);
    返回一个指针,指向字符串 s1 中字符 ch 的第一次出现的位置。
6
    strstr(s1, s2);
    返回一个指针,指向字符串 s1 中字符串 s2 的第一次出现的位置。
```

下面的实例使用了上述的一些函数:

```
实例
#include <stdio.h>
#include <string.h>
int main ()
char str1[12] = "Hello";
char str2[12] = "World";
char str3[12];
int len ;
/* 复制 str1 到 str3 */
strcpy(str3, str1);
printf("strcpy( str3, str1) : %s\n", str3 );
/* 连接 str1 和 str2 */
strcat( str1, str2);
printf("strcat( str1, str2): %s\n", str1 );
/* 连接后, str1 的总长度 */
len = strlen(str1);
printf("strlen(str1) : %d\n", len );
return 0;
}
```

当上面的代码被编译和执行时,它会产生下列结果:

```
strcpy( str3, str1) : Hello
strcat( str1, str2): HelloWorld
strlen(str1) : 10
```

2019/3/17 C 字符串 | 菜鸟教程