

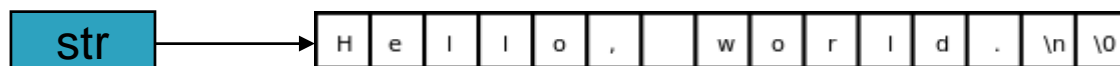
# 千锋嵌入式学院C语言培训

## ——字符串专题

# 字符串定义和初始化

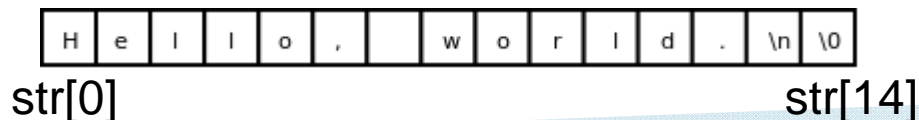
## ▶ 字符串常量

- `char *str = "Hello, world.";` //str指向'H', 以NULL结尾



## ▶ 字符数组

- `char str[ ] = "Hello, world.";` //数组大小15,最后一个字符是NULL
- `char str[10] = "Hello, world.";` //数组大小15,最后一个字符是NULL
- `char str[ ] = { 'H', 'e', 'l', 'l', 'o', ' ', 'w', 'o', 'r', 'l', 'd', '.', '\0' };`



## 字符串用法和常见错误

- ▶ `char* str = "Hello, world.\n";`
  - `str = "Hello QianFeng!"` // 正确, 指针str指向另一个字符串
  - `*(str+3) = 'z';` // 错误, 常量不可修改
- ▶ `char str[ ] = "Hello, world.\n";`
  - `str = "Hello QianFeng!";` // 错误, 数组不可整体赋值
  - `str[3] = 'z';` // 正确, 改变第4个字符为 'z'
- ▶ 常见错误
  - 为数组分配的空间不够, 无法容纳字符串
  - 忘记添加结束符NULL
  - 把 "a" 与 'a' 互相写错

# 字符判断函数

- int isdigit (int c) // 是否数字
- int isalpha (int c) // 是否字母
- int isalnum (int c) // 数字或字母
- int isxdigit (int c) // 十六进制数字字符
- int islower (int c) // 小写字母
- int isupper (int c) // 大写字母
- int tolower (int c) // 转换成小写字母
- int toupper (int c) // 转换成大写字母
- int isspace (int c) // 是否空格
- int isctrl (int c) // 是否控制符
- int ispunct (int c) // 除空格，数字和字母外的可打印字符
- int isprint (int c) // 可打印字符
- int isgraph (int c) // 除空格外的可打印字符

# 字符判断函数

- ▶ 练习：单词计数，统计字符串中用空格分开的单词数（常考题目）



```
4 #define TRUE 1
5 #define FALSE 0
6
7 int getwordnum(const char* str)
8 {
9     int cnt=0;
10
11     int out=TRUE;
12
13     int i=0;
14
15     while( str[i]!='\0' )
16     {
17         if(' '==str[i])
18         {
19             if(!out)
20             {
21                 out=TRUE;
22             }
23         }
24         else
25         {
26             if(out)
27             {
28                 cnt++;
29                 out=FALSE;
30             }
31         }
32         i++;
33     }
34     return cnt;
35 }
36 }
```



## 字符串转换函数

- ▶ `double atof (const char* nPtr) //字符串转double`
- ▶ `int atoi (const char* nPtr) //字符串转int`
- ▶ `long stol (const char* nPtr) //字符串转long`
- ▶ `double strtod (const char *nPtr, char **endPtr)`  
`//字符串转double`
- ▶ `long strtol (const char *nPtr, char **endPtr, int base)`  
`//字符串转long`
- ▶ `unsigned long strtoul (const char *nPtr, char **endPtr, int base) //字符串转 unsigned long`

## 字符串转换函数

- ▶ 练习：节假日很多商品都会打折，给定商品价格\$112.50，打八折后输出。

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

double discount(const char *str)
{
    double price = atof(str+1);
    price *= 0.8f;
    return price;
}

int main()
{
    char *pri = "$112.50";
    printf("%.2f\n", discount(pri));
    return 0;
}
```



# 字符串操作函数

## ▶ 字符串复制函数

- `char * strcpy (char *s1, const char *s2)`
- `char * strncpy (char *s1, const char *s2, size_t n)`

Q:strncpy会自动添加'\0'不?

## ▶ 字符串拼接函数

- `char * strcat (char *s1, const char *s2);`
- `char * strncat (char *s1, const char *s2, size_t n)`

## ▶ 字符串长度计算

- `size_t strlen( const char *s)`

# 字符串操作函数

- ▶ 练习：自己实现 strlen(常考题目)

```
int mystrlen(const char* str)
{
    int len = 0;
    char *index = (char*)str;
    while(*index++ != '\0') // not NULL
        len++;             // increase len
    return len;
}
```

# 字符串查找函数

- ▶ 在字符串中搜索字符
  - `char * strchr (const char *s, int c)`
  - `char * strrchr (const char *s, int c)`
- ▶ 查找特定字符串长度
  - `size_t strcspn (const char *s1, const char *s2)`
  - `size_t strspn (const char *s1, const char *s2)`
- ▶ 搜索子串
  - `char * strpbrk (const char *s1, const char *s2)`
  - `char * strstr (const char *s1, const char *s2)`
- ▶ 字符串分割
  - `char * strtok (char *s1, const char *s2)`

## 字符串比较函数

- ▶ `int strcmp (const char *s1, const char *s2)`
- ▶ `int strncmp (const char *s1, const char *s2, size_t n)`
- ▶ 练习：输出 “abcdcabcdcefcddcdcg” 中 “cdc” 出现的次数
  - cdccdc 算作出现两次
  - cddcdc 算作出现一次

## 字符串输入输出函数

- ▶ `int sprintf( char *s, const char* format, ...)`
- ▶ `int sscanf( const char *s, const char* format, ...)`
- ▶ 练习：向字符数组里写入整数，浮点数
- ▶ 练习：从” \$123.34” 中提取出数字



## 课后练习

- ▶ 练习：打印出一行句子中的单词，单词可以用空格，逗号(,)，句号(.)，问号(?)，叹号(!)，分号(;)分开。
- ▶ `char * substr (char *src, int start, int end)`
  - 从字符串src中取出从start到end的子串，返回子串指针，请实现。
- ▶ `char * insert (char *src, char *substr, int pos)`
  - 把substr插入到src的pos处，若pos大于最大下标，就插入到尾部，返回新串的指针，请实现。
- ▶ `char * remove( char *src, int start, int len)`
  - 从src的start处删除长度为len的字串，返回新串指针，请实现。

# 其他问题？

源自清华 值得信赖