

MD5库的使用

- 本库是 RFC1321 提供的标准 C 语言库
- 如果使用32位编译器，global.h中的UINT4的定义要改成：

```
typedef unsigned long int UINT4;
```

- 如果使用64位编译器，global.h中中的UINT4的定义要改成：

```
typedef unsigned int UINT4;
```

- 部分编译器对 long int 实际占用的字节数定义不一样，请检查：

```
printf("sizeof(UINT2)=%d, sizeof(UNIT4)=%d\n", sizeof(UINT2),  
sizeof(UINT4));
```

- 在项目中加入 md5.c 即可

代码示例

- 输入要计算的字符串
- 初始化上下文、更新数据、完成计算
- 按十六进制输出MD5摘要

In []:

```
#include <stdio.h>
#include <string.h>
#include "global.h"
#include "md5.h"

void md5String(unsigned char str[], unsigned int len, unsigned char md5[16])
{
    MD5_CTX ctx;          // 定义上下文
    MD5Init(&ctx);         // 初始化上下文

    MD5Update(&ctx, (unsigned char *)str, strlen(str)); // 计算散列摘要

    MD5Final(md5, &ctx); // 完成计算
}

int main(int argc, char *argv[])
{
    char str[4096];        // 要计算的字符串
    printf("Input: ");
    gets(str);

    unsigned char md5[16]; // 保存散列摘要结果
    md5String(str, strlen(str), md5);

    printf("sizeof(UINT2)=%d, sizeof(UNIT4)=%d\n", sizeof(UINT2), sizeof(UINT4));

    // 以十六进制形式输出结果
    int i;
    printf("MD5: ");
    for (i=0; i<16; i++) printf("%02X", (unsigned)md5[i]);
    printf("\n");

    return 0;
}
```