<https://blog.csdn.net/smf0504/article/details/52055971>

将两个char类型链接。

char d[20]="GoldenGlobal"; char \*s="View"; strcat(d,s);

结果放在d中

printf（"%s",d）；

输出 d 为 GoldenGlobalView （中间无空格）

d和s所指内存区域不可以重叠且d必须有足够的空间来容纳s的字符串。

返回指向d的[指针](http://baike.baidu.com/view/159417.htm" \t "_blank)。

## C函数

编辑

### 原型

extern char \*strcat(char \*dest,char \*src);

### 用法

#include <string.h>

在C++中，则存在于<cstring>头文件中。

### 功能

把src所指字符串添加到dest结尾处(覆盖dest结尾处的'\0')。

### 说明

src和dest所指内存区域不可以重叠且dest必须有足够的空间来容纳src的字符串。

返回指向dest的[指针](http://baike.baidu.com/view/159417.htm" \t "_blank)。

### 举例

|  |  |
| --- | --- |
| 1  2  3  4  5  6  7  8  9  10  11  12  13  14  15  16  17  18  19  20  21  22  23  24  25  26  27  28  29 | // strcat.c  #include <syslib.h>  #include <string.h>  main()  {      char d[20] = "GoldenGlobal";      char\* s = "View";      clrscr();      strcat(d,s);      printf("%s",d);      getchar();      return 0;  }    // strcat.cpp  #include <iostream>  #include <cstring>  #include <cstdlib>  int main()  {      using namespace std;      char d[20] = "GoldenGlobal";      char\* s = "View";      system("cls");      strcat(d,s);      cout << d << endl;      system("pause");      return 0;  } |

程序执行结果为：

GoldenGlobalView

Strcat函数原型如下（以下代码为错误代码，想要通过char \*指针修改字符串常量中的字符会导致Segment fault错误）：

|  |  |
| --- | --- |
| 1  2  3  4  5  6  7  8  9  10  11  12  13  14  15  16  17  18  19  20  21  22  23  24  25  26  27  28  29  30  31  32  33  34 | /\*   \* 注意以下代码有问题:   \* 1. 指针strDest被修改了，实际在使用中并不会去调用返回值来重新获取strDest原来的值   \* 2. 代码注释不该这么写，函数注释只需要写使用方法，无需写实现过程[所以实现过程尽量保证正确]   \*/    //将源字符串加const，表明其为输入参数  char\* strcat(char\* strDest , const char\* strSrc)  {      //后文return address，故不能放在assert断言之后声明address      char\* address=strDest;      assert( (strDest!=NULL)&&(strSrc!=NULL) );//对源地址和目的地址加非0断言      while(\*strDest)//是while(\*strDest!=’\0’)的简化形式      {          //若使用while(\*strDest++)，则会出错，因为循环结束后strDest还会执行一次++,          //那么strDest将指向'\0'的下一个位置。/所以要在循环体内++；因为要使\*strDest最后指          //向该字符串的结束标志’\0’。          strDest++;      }        while(\*strDest++=\*strSrc++)      {          NULL;//该循环条件内可以用++，      }//此处可以加语句\*strDest=’\0’;无必要      return address;//为了实现链式操作，将目的地址返回  }    4 char \*mystrcat(char \*dst,const char \*src) //用自己的方式实现strcat函数功能    5 {    6     char \*p=dst;  //下面的操作会改变目的指针指向，先定义一个指针记录dst    7     while(\*dst!='\0')dst++;    8     while(\*src!='\0')\*dst++=\*src++;    9     \*dst='\0';   10 return p;  //dst现在指向拼接后的最后一位字符，在这里返回dst，会出现错误   11 } |

版权声明：本文为博主原创文章，未经博主允许不得转载。 https://blog.csdn.net/SMF0504/article/details/52055971