



第10讲 字符指针和字符数组

教材10.1~10.5节

MOOC第10周

哈尔滨工业大学

苏小红

sxh@hit.edu.cn

凯撒密码与藏头诗

BhvLoryhbrx

一叶轻舟向东流，
帆梢轻握杨柳手，
风纤碧波微起舞，
顺水任从雅客悠。



如何表示与存储一个字符串？

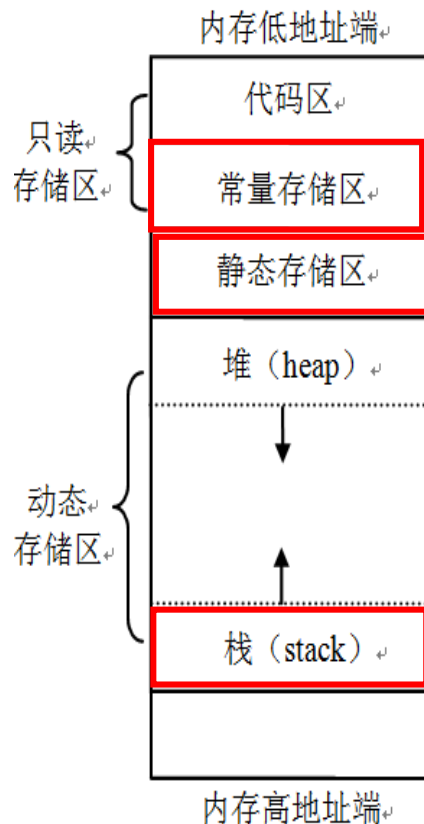
```
printf("Hello world");
```

用双引号括起的一串字符是字符串字面量(string literal)，系统自动为其添加空字符'\0'

'\0' 不是 '0' 哦！



用空字符'\0'标志字符串的末尾——字符串结束标志
哨兵字符——why?



下面定义和初始化字符数组正确的是（）

A `char str[6] = {'C', 'h', 'i', 'n', 'a', '\0'};`

B `char str[6] = {"China"};`

C `char str[6] = "China";`

D `char str[10] = "China";`

E `char str[] = "China";`

F `char str[4] = "China";`

G `char str[6] = {'C', 'h', 'i', 'n', 'a'};`

提交

如何保存和修改一个字符串？

```
char str[] = "Hello World";
```

```
str = "Hello China";
```

```
strcpy(str, "Hello China");
```

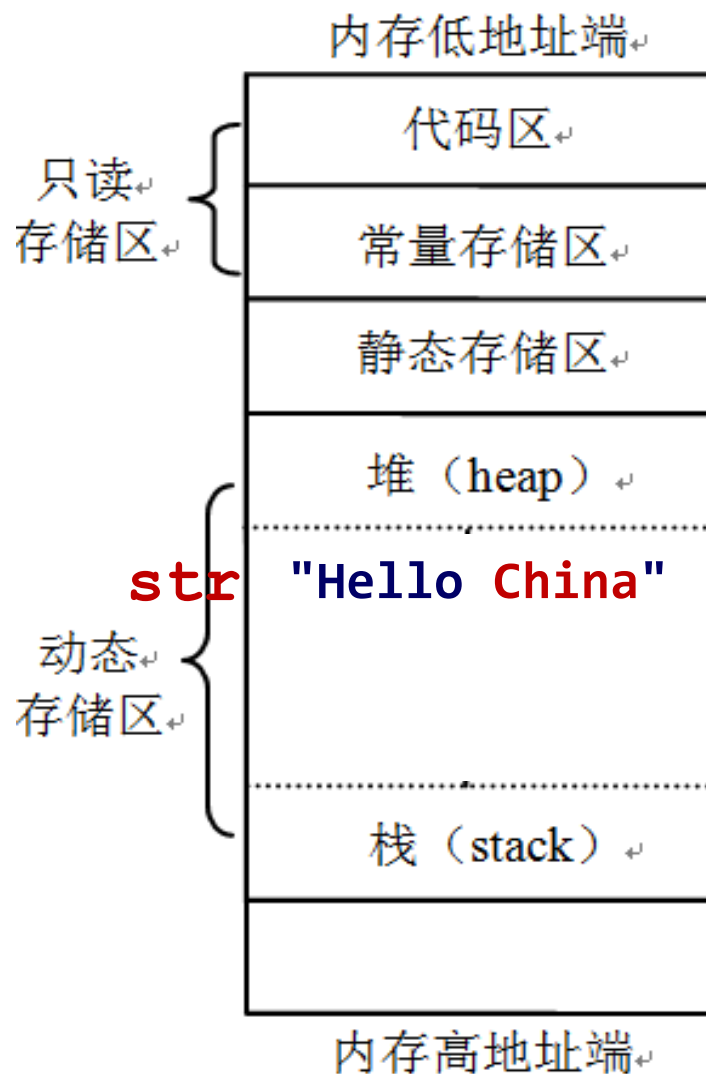
```
str[6] = 'C';
```

```
str[7] = 'h';
```

```
str[8] = 'i';
```

```
str[9] = 'n';
```

```
str[10] = 'a';
```



多选题

已知 `char str[] = "Hello World";`
下列能正确修改字符串的是 ()

☐ A `str = "Hello China";`

☒ B `strcpy(str, "Hello China");`

☒ C `str[6] = 'C';`
`str[7] = 'h';`
`str[8] = 'i';`
`str[9] = 'n';`
`str[10] = 'a';`

提交

如何保存和修改一个字符串？

```
char str[] = "Hello World";
```

```
str = "Hello China"; ❌
```

```
strcpy(str, "Hello China"); ✓
```

```
str[6] = 'C';
```

```
str[7] = 'h';
```

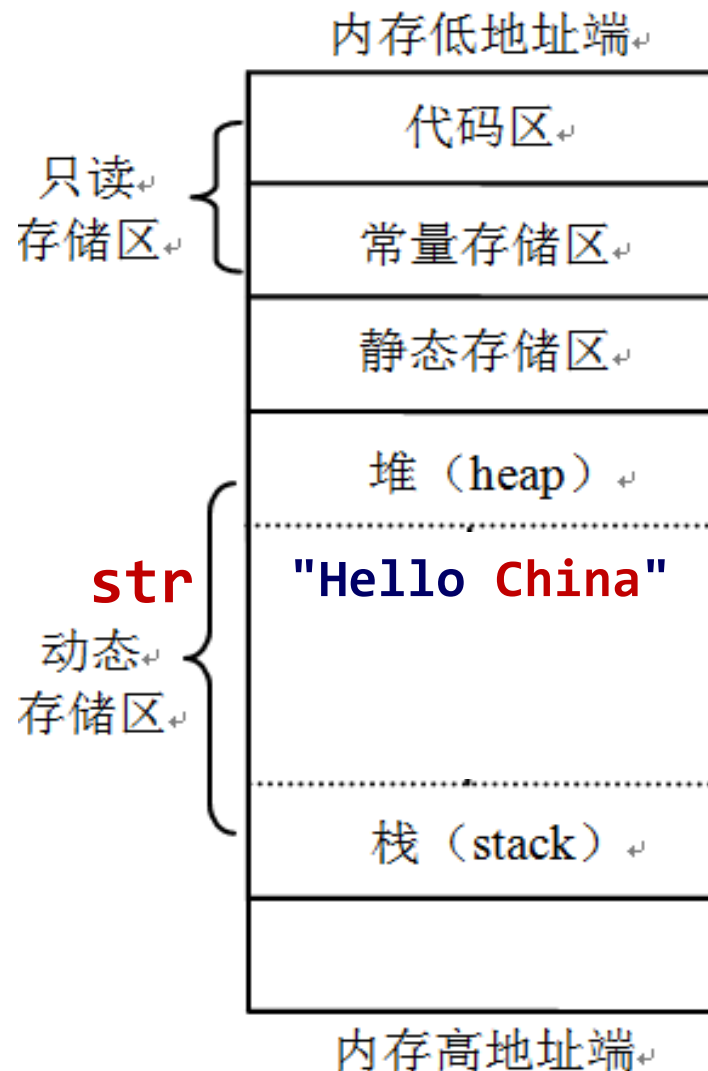
```
str[8] = 'i';
```

```
str[9] = 'n';
```

```
str[10] = 'a';
```

数组名 **str** 是地址常量，不可修改

数组元素(内容)可被修改



字符数组 vs 字符指针

```
char str[] = "Hello World";
```

```
char *pStr = str;
```

```
pStr = "Hello China";    ?
```

```
strcpy(pStr, "Hello China");
```

```
pStr[6] = 'C';
```

```
pStr[7] = 'h';
```

```
pStr[8] = 'i';
```

```
pStr[9] = 'n';
```

```
pStr[10] = 'a';
```

```
pStr = &str[6];
```

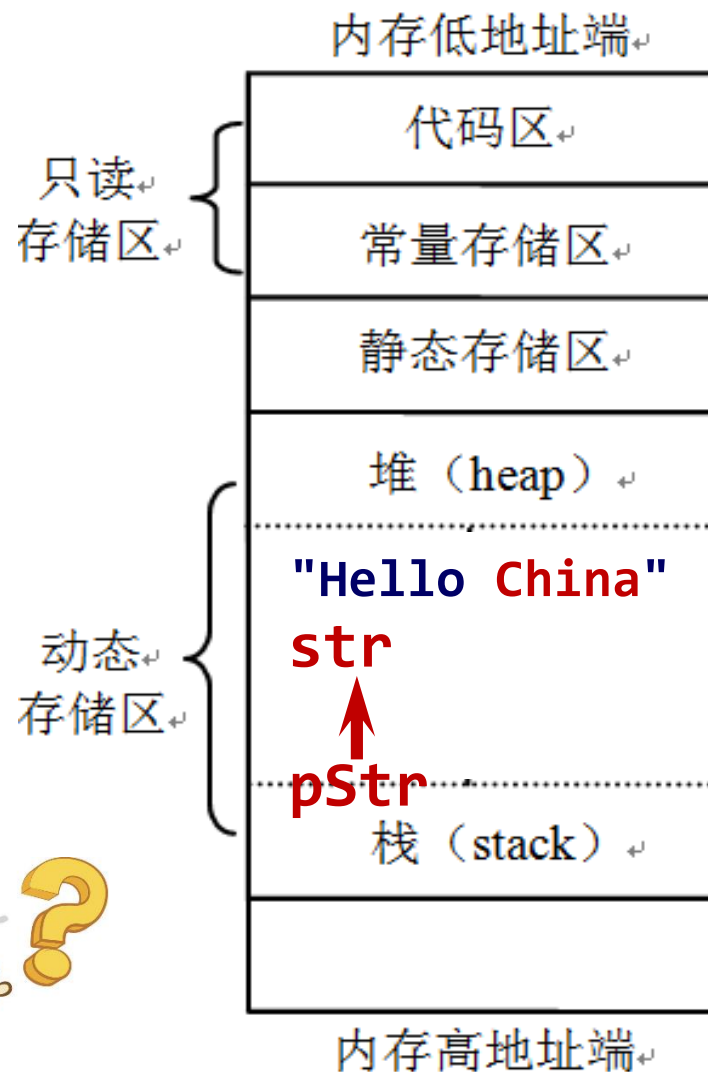
```
*pStr++ = 'C';
```

```
*pStr++ = 'h';
```

```
*pStr++ = 'i';
```

```
*pStr++ = 'n';
```

```
*pStr = 'a';
```



下列能正确修改数组str中的字符串的是（）

A

```
pStr = "Hello China";
```

B

```
strcpy(pStr, "Hello China");
```

C

```
pStr[6] = 'C';  
pStr[7] = 'h';  
pStr[8] = 'i';  
pStr[9] = 'n';  
pStr[10] = 'a';
```

D

```
pStr = &str[6];  
*pStr++ = 'C';  
*pStr++ = 'h';  
*pStr++ = 'i';  
*pStr++ = 'n';  
*pStr = 'a';
```

提交

字符数组 vs 字符指针

```
char str[] = "Hello World";
```

```
char *pStr = str;
```

```
pStr = "Hello China";    ?
```

```
strcpy(pStr, "Hello China");
```

```
pStr[6] = 'C';
```

```
pStr[7] = 'h';
```

```
pStr[8] = 'i';
```

```
pStr[9] = 'n';
```

```
pStr[10] = 'a';
```

```
pStr = &str[6];
```

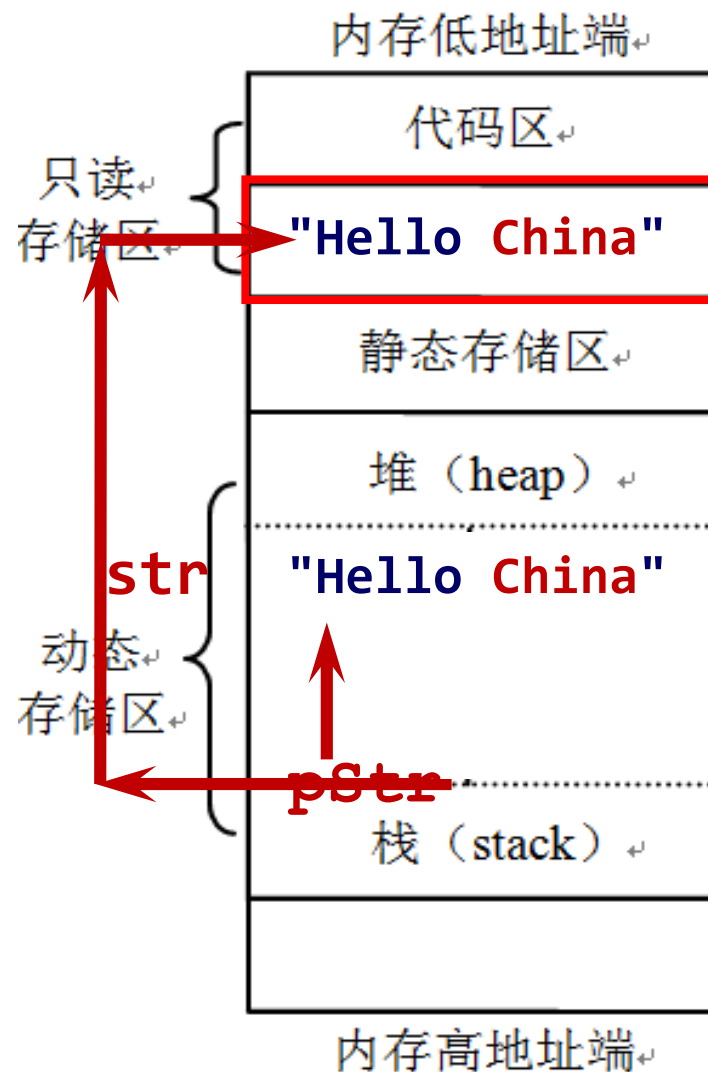
```
*pStr++ = 'C';
```

```
*pStr++ = 'h';
```

```
*pStr++ = 'i';
```

```
*pStr++ = 'n';
```

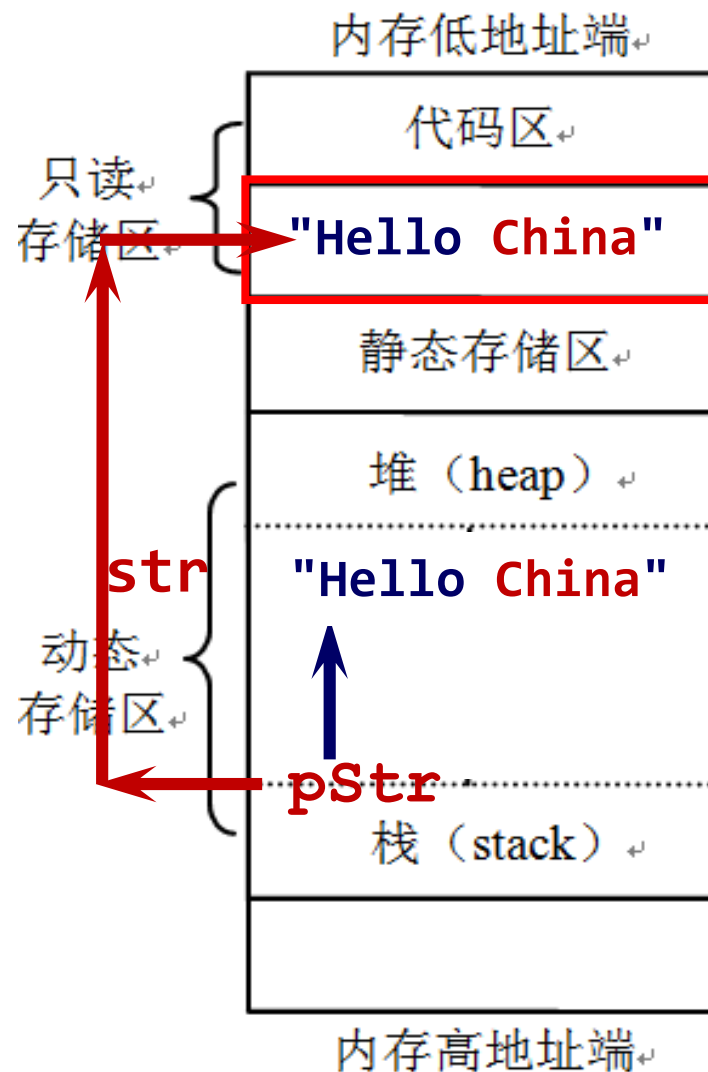
```
*pStr = 'a';
```



字符数组 vs 字符指针

```
char str[] = "Hello World";  
char *pStr = str;  
char *pStr = "Hello China";  
const char *pStr = "Hello China";
```

- 正确使用字符指针须牢记以下基本原则：
 - * 明确字符串被保存在哪里——指向只读存储区的指针内容不可被修改
 - * 明确字符指针指向了哪里——指针必须初始化



小组讨论：5分钟推理剧——神奇子弹案件

- **Alice**的**男盆友Jimmy**是当地一家星巴克的服务生，他的业余爱好是收集动物标本，他梦想着有一天能制作一头大象标本，可惜他交友不慎。
- 那天下午，**蒙面劫匪Masked Raider**去喝咖啡时遇到了Jimmy，那时他们还都活着：

```
char maskedRaider[] = "Alive";  
char *jimmy = maskedRaider;  
printf("Masked Raider is %s\n", maskedRaider);  
printf("Jimmy is %s\n", jimmy);
```



C:\Users\sxh\Desktop\c\same\bin
Masked Raider is Alive
Jimmy is Alive



小组讨论：5分钟推理剧——神奇子弹案件

- 那天晚上，蒙面劫匪动身去抢劫酒吧。过去从未失手，但这次当他踹开酒吧门时，发现联邦调查局的人正在玩王者欢度周末，枪声响起，歹徒倒地
- 奇怪的是，Alice去咖啡店找他男盆友时，却被告知他昨晚调的那杯摩卡冰乐已成绝唱。
- 为什么这颗神奇的子弹同时杀死了Jimmy和蒙面劫匪？

```
char maskedRaider[] = "Alive";  
char *jimmy = maskedRaider;  
printf("Masked Raider is %s\n", maskedRaider);  
printf("Jimmy is %s\n", jimmy);  
maskedRaider[0] = 'D';  
maskedRaider[1] = 'E';  
maskedRaider[2] = 'A';  
maskedRaider[3] = 'D';  
maskedRaider[4] = '!';  
printf("Masked Raider is %s\n", maskedRaider);  
printf("Jimmy is %s\n", jimmy);
```

C:\Users\sxh\Desktop\c\same\bin\

```
Masked Raider is Alive  
Jimmy is Alive  
Masked Raider is DEAD!  
Jimmy is DEAD!
```

如何输入一个字符串？

```
#define LEN 80  
char str[LEN+1];  
scanf("%s", str);
```

```
gets(str);
```

第1回合：能输入多个
字段吗？

第2回合：能输入带空
格的字符串吗？

第3回合：限制用户输
入的字符数吗？



拳王争霸赛

1 : 0

1 : 1

2- : 1

如何输入一个字符串？

```
#define LEN 80  
char str[LEN+1];  
scanf("%s", str);
```

```
fgets(str, sizeof(str), stdin);
```

第1回合：能输入多个
字段吗？

第2回合：能输入带空
格的字符串吗？

第3回合：限制用户输
入的字符数吗？



拳王争霸赛

1 : 0

1 : 1

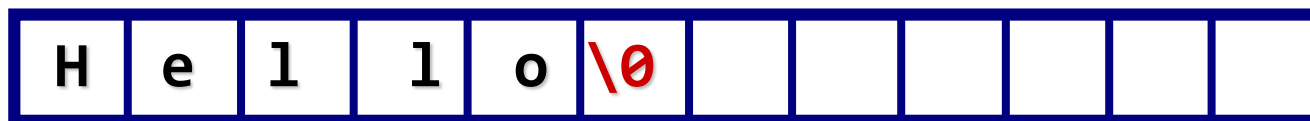
2- : 2+

如何输入一个字符串？

```
char str[80];
```

Hello kids ✓

```
scanf("%s", str);
```



遇空白字符结束，不读走 '\n'

```
gets(str);
```



遇回车结束，读走 '\n'，换成 '\0'

```
fgets(str, sizeof(str), stdin);
```



读走 '\n'
插在末尾

假设有：char str[80];

char *pStr;

则下面哪条语句可能导致程序崩溃？

A

pStr[0] = 'a';

B

*pStr = 'a';

C

scanf("%s", pStr);

D

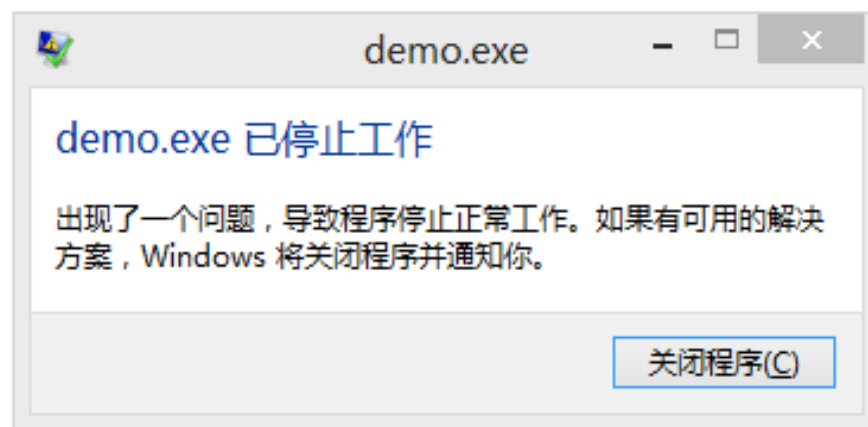
gets(pStr);

E

scanf("%s", str);

F

gets(str);



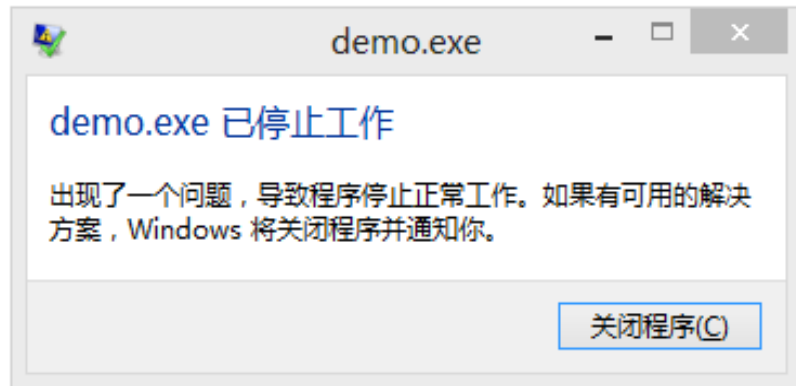
提交

如何正确使用字符指针输入一个字符串？

```
#include <stdio.h>

#define LEN 80

int main()
{
    char str[LEN+1];
    char *ptr = str;
    printf("Input a string:");
    gets(ptr);
    printf("%s\n", ptr);
    return 0;
}
```

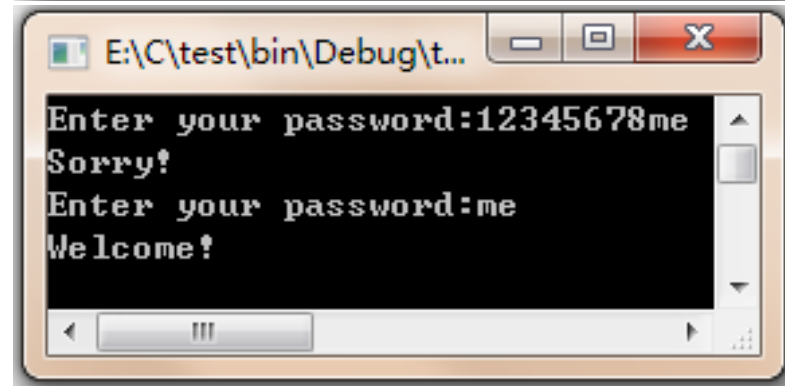
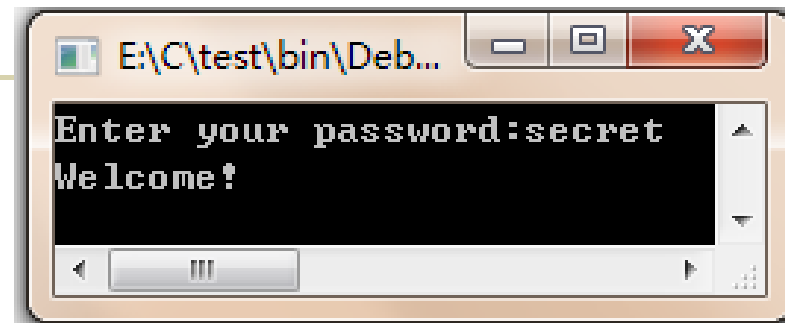


- 几乎全是由指针和数组引起的**非法内存访问**导致的
- 黑客攻击服务器利用的bug绝大部分是指针和数组造成的



缓冲区溢出攻击实例

```
#include <stdio.h>
#include <string.h>
int main()
{
    char password[8]="secret", input[8];
    while (1)
    {
        printf("Enter your password:");
        gets(input);
        if (strcmp(input, password) == 0)
        {
            printf("Welcome!\n");
            break;
        }
        else
        {
            printf("Sorry!\n");
        }
    }
    return 0;
}
```



考试攻略

- 5月2日考字符串和结构体
 - * 必做题是复杂编程题，没有普通编程题
 - * 普通编程题，千万不要使用指针和结构体编写程序
 - * 附加改错题，可以使用指针和结构体
 - * 能用字符数组编写的程序，不要使用字符指针
- 直接拷贝粘贴题目给的格式信息，避免输入输出格式错误
- 借助编译提示，尽量消灭程序中的语法错误，否则分数很低
- 熟记常用基本操作的算法和范例程序
- 不仅测试题目给的用例，还要考虑边界条件的测试

