



# 第12章 文件操作

## ——按行读写文件



哈尔滨工业大学

苏小红

sxh@hit.edu.cn

# 按行写文件

- `int puts(const char *s);`

- \* 将字符串s写入标准输出流stdout，并在其后添加一个换行符

- 按字符串（行）写文件

- `int fputs(const char *s, FILE *fp);`

- \* 将字符串s写入fp所指的文件中

- \* 与puts()相同的是：若出现写入错误，则返回EOF，否则返回一个非负数

- \* 与puts()不同的是：它不会自动添加换行符，除非字符串本身含有换行符

# 按行读文件

- `char *gets(char *s);`

- \* 从标准输入流`stdin`读取字符串，读到换行符时停止

- 按字符串（行）**读**文件

- `char *fgets(char *s, int n, FILE *fp);`

- \* 从`fp`所指的文件中读取字符串，最多读`n-1`个字符

- \* 当读到回车换行符、文件末尾，或读满`n-1`个字符时，函数返回该字符串的首地址

- \* 用`fgets(buf, sizeof(buf), stdin)`代替`gets(buf)`更安全

# 按行读文件

- `char *gets(char *s);`
- `char *fgets(char *s, int n, FILE *fp);`

## 相同点

- 都会在字符串末尾添加'\0'
- 读取失败或者到达文件尾都返回NULL

## 不同点

- 对换行符的处理不同
- gets不保留'\n'替换为'\0'
- fgets保留并存储'\n'

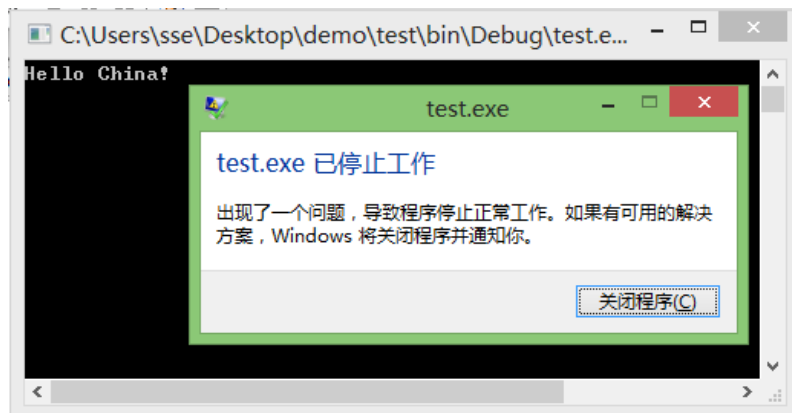
# 按行读写文件

从键盘键入一串字符，  
转存到磁盘文件上

```
#include <stdio.h>
#include <stdlib.h>
int main()
{
    FILE *fp;
    char str[80];
    if ((fp = fopen("demo.txt", "w")) == NULL)
    {
        printf("Failure to open demo.txt!\n");
        exit(0);
    }
    gets(str);
    fputs(str, fp);
    fclose(fp);
    return 0;
}
```

# 按行读写文件

从键盘键入一串字符，  
转存到磁盘文件上



```
#include <stdio.h>
#include <stdlib.h>
int main()
{
    FILE *fp;
    char str[8];
    if ((fp = fopen("demo.txt", "w")) == NULL)
    {
        printf("Failure to open demo.txt!\n");
        exit(0);
    }
    gets(str);
    fputs(str, fp);
    fclose(fp);
    return 0;
}
```

# 按行读写文件

从键盘键入一串字符，  
转存到磁盘文件上

```
#include <stdio.h>
#include <stdlib.h>
int main()
{
    FILE *fp;
    char str[8];
    if ((fp = fopen("demo.txt", "w")) == NULL)
    {
        printf("Failure to open demo.txt!\n");
        exit(0);
    }
    fgets(str, sizeof(str), stdin);
    fputs(str, fp);
    fclose(fp);
    return 0;
}
```

# 按行读写文件

从文件读入一串字符，  
输出到屏幕上



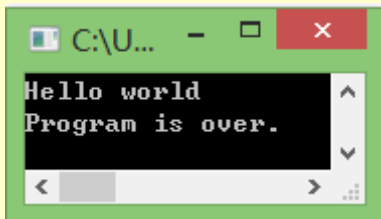
```
#include <stdio.h>
#include <stdlib.h>
int main()
{
    FILE *fp;
    char str[8];
    if ((fp = fopen("demo.txt", "r")) == NULL)
    {
        printf("Failure to open demo.txt!\n");
        exit(0);
    }
    fgets(str, sizeof(str), fp);
    puts(str);
    fclose(fp);
    return 0;
}
```



# 讨论

- 下面左侧的程序中将`gets(str)`修改为`fgets(str, sizeof(str), fp)`后, 再重新运行右侧的程序, 则输出结果将会在输出的Hello world和program is over.之间增加一个空行, 这是为什么? 请说明其原因。

```
#include <stdio.h>
#include <stdlib.h>
int main()
{
    FILE *fp;
    char str[80];
    if ((fp = fopen("demo.txt", "w")) == NULL)
    {
        printf("Failure to open demo.txt!\n");
        exit(0);
    }
    gets(str);
    fputs(str, fp);
    fclose(fp);
    return 0;
}
```



```
#include <stdio.h>
#include <stdlib.h>
int main()
{
    FILE *fp;
    char str[80];
    if ((fp = fopen("demo.txt", "r")) == NULL)
    {
        printf("Failure to open demo.txt!\n");
        exit(0);
    }
    fgets(str, sizeof(str), fp);
    puts(str);
    printf("Program is over.\n");
    fclose(fp);
    return 0;
}
```

