标准I/O（二）

主讲：华清远见 老师

课程目标

按字符输入（熟练）

按字符输出（熟练）

小结

以 函数内接收或送的变量，一般是数组形式，当然
malloc 也行但麻烦点吧。
XF：当然，觉能首先反应就首先反应，反应不了再按常规来，

标准I/O – 读写流

流支持不同的读写方式:

读写一个字符：fgetc()/fputc()一次读/写一个字符

读写一行：fgets()和fputs()一次读/写一行

读写若干个对象：fread()/fwrite() 每次读/写若干个对象，而每个对象具有相同的长度

标准I/O – 按字符输入

下列函数用来输入一个字符:

```
#include  <stdio.h>
int  fgetc(FILE *stream);
int  getc(FILE *stream);   //宏
int  getchar(void);
```

- 成功时返回读取的字符；若到文件末尾或出错时返回EOF（-1），
- getchar()等同于fgetc(stdin)
- getc和fgetc区别是一个是宏一个是函数

标准I/O – fgetc – 示例

```
int  ch;
ch = fgetc(stdin);
printf( “%c\n” , ch);

FILE  *fp;
int  ch, count = 0;
if ((fp  =  fopen(argv[1],  “r” )) == NULL)  {
  perror( “fopen” );  return  -1;
}
while ((ch  =  fgetc(fp)) != EOF) {
  count++;
}
printf( “total  %d bytes\n” , count);
```

→ 键盘入，打印。

→ 不断读字符，直到失败（始open，有字；里 count++，还打印）

标准I/O – 按字符输出

下列函数用来输出一个字符:

```
#include  <stdio.h>
int  fputc(int c, FILE *stream);
int  putc(int c, FILE *stream);
int  putchar(int c);
```

- 成功时返回写入的字符；出错时返回EOF
- putchar(c)等同于fputc(c, stdout)

标准I/O – fputc – 示例

```
fputc( ‘a’ , stdout);
putchar( ‘\n’ );

FILE *fp;
int ch;
if ((fp = fopen(argv[1], “w” )) == NULL) {
   perror( “fopen” ); return -1;
}
for(ch = ‘a’ ; ch <= ‘z’ ; ch++) {
   fputc(ch, fp);
}
```

把a-z写入参数2

open argv

fputc

for

封装

XF 核心 辅助 周围 声明.

标准I/O – 小结

fopen

fgetc

fputc

标准I/O – 思考和练习

如何利用fgetc / fputc实现文件的复制?

1. 通过命令行参数传递源文件和目标文件名

2. 通过fgetc返回值判断是否读到文件末尾