

Programming and Problem Solving through C Language O Level / A Level

Chapter - 11 : File Processing

The **ftell()** and **rewind()** Functions

- To set the position indicator to the beginning of the file, use the library function **rewind()**.
- Its prototype, in **STDIO.H**, is:
void rewind(FILE *fp);
- The argument **fp** is the **FILE** pointer associated with the stream.
- After **rewind()** is called, the file's position indicator is set to the beginning of the file (byte 0).
- The **rewind()** is used to read some data from the beginning of the file again without closing and reopening the file.

The **ftell()** and **rewind()** Functions

- To determine the value of a file's position indicator, use **ftell()**.
- This function's prototype, is located in **STDIO.H** and reads as follows:
long ftell(FILE *fp);
- The argument **fp** is the **FILE** pointer returned by **fopen()** when the file was opened.
- The function **ftell()** returns a type **long** that gives the current file position in bytes from the start of the file (the first byte is at position 0).
- If an error occurs, **ftell()** returns 1(of type **long**).

fseek() Function

- More precise control over a stream's position indicator is possible with the **fseek()** library function.
- By using **fseek()**, one can set the position indicator anywhere in the file.
- The function prototype , in **STDIO.H**, is as follows:
int fseek(FILE *fp, long offset, int origin);
- The argument **fp** is the **FILE** pointer associated with the file.
- The distance that the position indicator is to be moved is given by its offset in bytes.
- The argument **origin** specifies the moves relative starting point.
- There can be three values for **origin**, with symbolic constants defined in **io.h**.

SEEK_SET 0

SEEK_CUR 1

SEEK_END 2

fseek(fp, 0 , 0) to move the pointer to the beginning of file.

fseek(fp, 0, 2) to move the pointer to the end of the file.

fseek(fp, 0 , 1) to keep the pointer at the same point.

fseek(fp, 10,1) moves the pointer 10 byte in forward direction.

fseek(fp, -10,1) moves the pointer 10 byte in backward direction.

```
void main ( ) {  
    FILE *fp;  
  
    fp = fopen("file.txt","w+");  
    fputs("This is a C Language", fp);  
  
    fseek( fp, 7, SEEK_SET );  
    fputs(" C is High Level Language", fp);  
    fclose(fp);  
}
```

feof() Function

- With a binary mode stream, one can't detect the end-of-file by looking for 1, because a byte of data from a binary stream could have that value, which would result in a premature end of input.
- Instead the library can be used. Function feof(), which can be used for both binary and text mode files as shown below:
int feof(FILE *fp);
- The argument fp is the FILE pointer returned by fopen() when the file was opened.
- The function feof() returns 0 if the end of file fp hasn't been reached, or a nonzero value if end-of-file has been reached.
- If a call to feof() detects end-of-file, no further read operations are permitted until a rewind() has been done, fseek() is called, or the file is closed and reopened.