Description

The C library function **int fseek(FILE \*stream, long int offset, int whence)** sets the file position of the **stream** to the given **offset**.

Declaration

Following is the declaration for fseek() function.

int fseek(FILE \*stream, long int offset, int whence)

Parameters

* **stream** − This is the pointer to a FILE object that identifies the stream.
* **offset** − This is the number of bytes to offset from whence.
* **whence** − This is the position from where offset is added. It is specified by one of the following constants −

|  |  |
| --- | --- |
| **Sr.No.** | **Constant & Description** |
| 1 | **SEEK\_SET**  Beginning of file |
| 2 | **SEEK\_CUR**  Current position of the file pointer |
| 3 | **SEEK\_END**  End of file |

Return Value

This function returns zero if successful, or else it returns a non-zero value.

Example

The following example shows the usage of fseek() function.

#include <stdio.h>

int main () {

FILE \*fp;

fp = fopen("file.txt","w+");

fputs("This is tutorialspoint.com", fp);

fseek( fp, 7, SEEK\_SET );

fputs(" C Programming Language", fp);

fclose(fp);

return(0);

}

Let us compile and run the above program that will create a file **file.txt** with the following content. Initially program creates the file and writes *This is tutorialspoint.com* but later we had reset the write pointer at 7th position from the beginning and used puts() statement which over-write the file with the following content −

This is C Programming Language

Now let's see the content of the above file using the following program −

#include <stdio.h>

int main () {

FILE \*fp;

int c;

fp = fopen("file.txt","r");

while(1) {

c = fgetc(fp);

if( feof(fp) ) {

break;

}

printf("%c", c);

}

fclose(fp);

return(0);

}

Let us compile and run the above program to produce the following result −

This is C Programming Language

fseek() in C/C++ with example

* Difficulty Level : [Medium](https://www.geeksforgeeks.org/medium)
* Last Updated : 02 Jun, 2017

fseek() is used to move file pointer associated with a given file to a specific position.  
**Syntax:**

**int fseek(FILE \*pointer, long int offset, int position)**

**pointer:** pointer to a FILE object that identifies the stream.

**offset:** number of bytes to offset from position

**position:** position from where offset is added.

**returns:**

zero if successful, or else it returns a non-zero value

position defines the point with respect to which the file pointer needs to be moved. It has three values:  
SEEK\_END : It denotes end of the file.  
SEEK\_SET : It denotes starting of the file.  
SEEK\_CUR : It denotes file pointer’s current position.

filter\_none

edit

play\_arrow

brightness\_4

|  |
| --- |
| // C Program to demonstrate the use of fseek()  #include <stdio.h>    int main()  {      FILE \*fp;      fp = fopen("test.txt", "r");        // Moving pointer to end      fseek(fp, 0, SEEK\_END);        // Printing position of pointer      printf("%ld", ftell(fp));        return 0;  } |

Output:

81