fsetpos

Defined in header <stdio.h>

```
int fsetpos( FILE *stream, const fpos_t *pos );
```

Sets the file position indicator and the multibyte parsing state (if any) for the file stream stream according to the value pointed to by pos.

Besides establishing new parse state and position, a call to this function undoes the effects of ungetc and clears the end-of-file state, if it is set.

If a read or write error occurs, the error indicator (ferror) for the stream is set.

Parameters

stream - file stream to modify

pos - pointer to a fpos_t object to use as new value of file position indicator

Return value

0 upon success, nonzero value otherwise.

Notes

After seeking to a non-end position in a wide stream, the next call to any output function may render the remainder of the file undefined, e.g. by outputting a multibyte sequence of a different length.

Example

fsetpos with error checking

Run this code

```
#include <stdio.h>
#include <stdlib.h>

int main(void)
{
    /* Prepare an array of f-p values. */
    #define SIZE 5
    double A[SIZE] = {1.,2.,3.,4.,5.};
    /* Write array to a file. */
    FILE * fp = fopen("test.bin", "wb");
    fwrite(A,sizeof(double),SIZE,fp);
    fclose (fp);

    /* Read the f-p values into array B. */
    double B[SIZE];
    fp = fopen("test.bin","rb");
    fpos_t pos;
    if (fgetpos(fp,&pos) != 0)      /* current position: start of file */
    {
       perror("fgetpos()");
       fprintf(stderr,"fgetpos() failed in file %s at line # %d\n", __FILE__,__LINE__-3);
       exit(EXIT_FAILURE);
    }

    int ret_code = fread(B,sizeof(double),1,fp);   /* read one f-p value */
    /* current position: after reading one f-p value */
    printf("%.1f; read count = %d\n", B[0], ret_code);   /* print one f-p value and ret_code

    if (fsetpos(fp,&pos) != 0)   /* reset current position to start of file */
    {
       if (ferror(fp))
       {
          perror("fsetpos()");
          fprintf(stderr,"fsetpos() failed in file %s at line # %d\n", __FILE__,__LINE__-5);
          exit(EXIT_FAILURE);
       }
```

```
    }

    ret_code = fread(B,sizeof(double),1,fp);   /* reread first f-p value */
    printf("%.1f; read count = %d\n", B[0], ret_code);   /* print one f-p value and ret_code
    fclose(fp);

    return EXIT_SUCCESS;
}
```

Output:

```
1.0; read count = 1
1.0; read count = 1
```

References

- C11 standard (ISO/IEC 9899:2011):
  - 7.21.9.3 The fsetpos function (p: 337)
- C99 standard (ISO/IEC 9899:1999):
  - 7.19.9.3 The fsetpos function (p: 303)
- C89/C90 standard (ISO/IEC 9899:1990):
  - 4.9.9.3 The fsetpos function

See also

fgetpos	gets the file position indicator (function)
ftell	returns the current file position indicator (function)
fseek	moves the file position indicator to a specific location in a file (function)

C++ documentation for **fsetpos**