

Programme : Bachelor of Science (First Year)

Subject : Computer Science

Semester : I

Course Code : CSC101

Course Title : Programming Fundamentals Using C

Title of the Unit : File Handling

Module Name : Read and Write

File management functions

Reading File : fgetc() function

- The fgetc() function returns a single character from the file.
- It gets a character from the stream.
- It returns EOF at the end of file.

Syntax:

int fgetc(FILE *stream) ;

Example: Display contents of a file on screen.

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

```
main( )
```

```
{ FILE *fp ;
```

```
char ch ;
```

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

```
while ( 1 )
```

```
{ ch = fgetc ( fp ) ;
```

```
if ( ch == EOF )
```

```
break ;
```

```
printf ( "%c", ch ) ;
```

```
}
```

```
fclose ( fp ) ;
```

```
}
```

Writing File : fputc() function

- The fputc() function is used to write a single character into file.
- It outputs a character to a stream

Syntax:

```
int fputc(int c, FILE *stream);
```

Example: Write a single character into a file

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

```
main()
```

```
{
```

```
    FILE *fp;
```

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

```
    fputc('a',fp);
```

```
    fclose(fp);
```

```
}
```

Writing File :fprintf() function

- The fprintf() function is used to write set of characters into file.

Syntax:

```
int fprintf(FILE *stream, const char *format [, argument, ...]);
```

Example : Write to a text file

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

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

```
int main()
```

```
{ int num;
```

```
    FILE *fp;
```

```
    fp = fopen("test.txt", "w");
```

```
    if(fp == NULL)
```

```
{ printf("Error!");
```

```

    exit(1);
}

printf("Enter num: ");
scanf("%d",&num);
fprintf(fp,"%d",num);
fclose(fp);
return 0; }

```

Reading File : fscanf() function

- The fscanf() function is used to read set of characters from file.
- It reads a word from the file and returns EOF at the end of file.

Syntax:

int fscanf(**FILE** *stream, **const char** *format [, argument, ...]);

Example : Read from a text file

```

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

int main()
{ int num;
FILE *fp;
if ((fp = fopen("test.txt","r")) == NULL)
{ printf("Error! opening file");
  exit(1);
}

fscanf(fp,"%d", &num);
printf("Value of n=%d", num);
fclose(fp);
return 0; }

```

String I/O functions in C

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

This function reads a character from the file stream pointed by `fptr` and stores it in the character array 'str' until a new line character (`\n`) is read or end of file (EOF) is reached.

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

This function writes data to the stream pointed to by `fp`, the content of the string stored in 'str'

Example:

Receives strings from keyboard and writes them to file

```
#include<stdio.h>

main( )
{
    FILE *fp ;
    char s[80] ;
    fp = fopen ( "test.TXT", "w" ) ;
    if ( fp == NULL )
    {
        puts ( "Cannot open file" ) ;
    }
    printf ( "\nEnter a few lines of text:\n" ) ;
    while ( strlen ( gets ( s ) ) > 0 )
    {
        fputs ( s, fp ) ;
        fputs ( "\n", fp ) ;
    }
    fclose ( fp ) ;
}
```

Example :

Reads strings from the file and displays them on screen

```
#include <stdio.h>

main( )
```

```

{
FILE *fp ;
char s[80] ;
fp = fopen ( "test.TXT", "r" ) ;
if ( fp == NULL )
{
puts ( "Cannot open file" ) ;
}
while ( fgets ( s, 79, fp ) != NULL )
printf ( "%s" , s ) ;
fclose ( fp ) ;
}

```

fseek() function

- The fseek() function is used to set the file pointer to the specified offset.
- It is used to write data into file at desired location.

Syntax:

```
int fseek(FILE *stream, long int offset, int whence);
```

There are 3 constants used in the fseek() function for whence: SEEK_SET, SEEK_CUR and SEEK_END.

Example:

```

#include <stdio.h>

main()
{
    FILE *fp;
    fp = fopen("myfile.txt","w+");
    fputs("This is C Programming", fp);
    fseek( fp, 8, SEEK_SET );
    fputs("FSEEK() function", fp);
}

```

```
fclose(fp);  
}
```

rewind() function

- The rewind() function sets the file pointer at the beginning of the stream. It is useful if you have to use stream many times.

Syntax:

```
void rewind(FILE *stream);
```

Example:

```
#include<stdio.h>  
  
main(){  
FILE *fp;  
char c;  
fp=fopen("file.txt","r");  
while((c=fgetc(fp))!=EOF){  
printf("%c",c); }  
rewind(fp);  
while((c=fgetc(fp))!=EOF){  
printf("%c",c); }  
fclose(fp);  
}
```

ftell() function

- The ftell() function returns the current file position of the specified stream.
- Can be used to get the total size of a file after moving file pointer at the end of file.

Syntax:

```
int ftell(FILE *stream);
```

Example:

```
#include <stdio.h>  
  
main (){
```

```
FILE *fp;
int length;
fp = fopen("file.txt", "r");
fseek(fp, 0, SEEK_END);
length = ftell(fp);
fclose(fp);
printf("Size of file: %d bytes", length);
}
```

Example: Program to copy contents from one file to another and display content of second file.

```
#include <stdio.h>
main()
{ FILE *fp1, *fp2;
  char ch;
  fp1 = fopen("file1.txt", "r");
  fp2 = fopen("file2.txt", "w");
  while((ch = fgetc(fp1)) != EOF)
  { fputc(ch, fp2);
    printf("%c",ch);}
  fclose(fp1);
  fclose(fp2); }
```