FILING

Week 12

Sumaiyah Zahid

Filing

- It saves your data even if the program terminates.
- You can read a large amount of data using files.
- You can easily move your data from one computer to another without any changes.

File Operations

1. Creating a new file
2. Opening an existing file
3. Closing a file
4. Reading from and writing information to a file

Creating A File / Opening An Existing File

```
FILE *fptr;  // FILE is datatype

//fptr = fopen("fileopen","mode");
fptr = fopen("C:\\program.txt","w");

fclose(fptr);
```

Sr.No.	Mode & Description
1	**r** Opens an existing text file for reading purpose.
2	**w** Opens a text file for writing. If it does not exist, then a new file is created. Here your program will start writing content from the beginning of the file.
3	**a** Opens a text file for writing in appending mode. If it does not exist, then a new file is created. Here your program will start appending content in the existing file content.
4	**r+** Opens a text file for both reading and writing.
5	**w+** Opens a text file for both reading and writing. It first truncates the file to zero length if it exists, otherwise creates a file if it does not exist.
6	**a+** Opens a text file for both reading and writing. It creates the file if it does not exist. The reading will start from the beginning but writing can only be appended.

```c
#include <stdio.h>
int main()
{
  int num;
   FILE *fptr;

   fptr = fopen("C:\\program.txt","w");

   if(fptr == NULL)
    {
       printf("Error!");
    }
    fclose(fptr);

   return 0;
}
```

Writing To A File

```
fputc(char, file_pointer)

fputs(str, file_pointer)

fprintf(file_pointer, str, variable_lists)
```

```c
#include <stdio.h>
int main()
{
  int num;
  FILE *fptr;
  fptr = fopen("C:\\program.txt","w");
  if(fptr == NULL)
   {
      printf("Error!");
   }
  else
   {
      printf("Enter num: ");
      scanf("%d",&num);
      fprintf(fptr,"You entered %d\n Happy Coding",num);}
   fclose(fptr);
  return 0;   }
```

Reading From A File

```
fgetc(file_pointer)

fgets(buffer, count, file_pointer)

fscanf(file_pointer, str, variable_lists)
```

```c
#include <stdio.h>
int main()
{
  int num; FILE *fptr; char c;
   fptr = fopen("program.txt","r");
   if(fptr == NULL)
    {
       printf("Error!");
    }
   else
    {
       while ((c = fgetc(fptr)) != EOF)
           printf("%c", c);
    }
    fclose(fptr);
   return 0;
}
```

```c
#include <stdio.h>
int main()
{
  int num; FILE *fptr; char buffer[50];
   fptr = fopen("program.txt","r");
   if(fptr == NULL)
    {
       printf("Error!");
    }
   else
    {
       fgets(buffer,50,fptr); // It reads a single line
       printf("%s", buffer);
    }
    fclose(fptr);
   return 0;
}
```

```c
#include <stdio.h>
int main()
{
  int num; FILE *fptr; char c[100], d[100];
  fptr = fopen("program.txt","r");
  if(fptr == NULL)
   {
      printf("Error!");
   }
  else
   {
      fscanf(fptr,"%s %s %d",&c, &d, &num);
      printf("%s %s %d", c,d,num);
   }
   fclose(fptr);
  return 0;
}
```

FSEEK

whence defines the point with respect to where the file pointer needs to be moved. It is specified by one of the following constants:

- SEEK_END: End of the file.
- SEEK_SET: Beginning of the file.
- SEEK_CUR: Current position of the file pointer.

```
fseek(fptr, 0, SEEK_END);
fseek(fptr, 10, SEEK_SET);
```

```c
#include <stdio.h>
int main()
{
  int num; FILE *fptr; char c[100], d[100];
  fptr = fopen("program.txt","r+");
  if(fptr == NULL)
      printf("Error!");
  else
   {
      fscanf(fptr,"%s %s %d",&c, &d, &num);
      printf("%s %s %d", c,d,num);
      fseek(fptr, 0, SEEK END);
      fprintf(fptr,"\nI am the new last line :)");
   }
   fclose(fptr);
  return 0;
}
```

Home Assignment

Write a program to read a file and display contents with its line numbers.

Make a factorial program which takes input from a file input.txt and saves the output in another file result.txt.

Make a program which reads all the text from a file but saves only count of vowels to another file.

References

https://www.guru99.com/c-file-input-output.html

https://www.educative.io/edpresso/what-is-fseek-in-c