FILE

Q.Describe different types of file operation in data file?Marks:3 Exam-ACCE-2014

Ans:

A file represents a sequence of bytes, does not matter if it is a text file or binary file. C programming language provides access on high level functions as well as low level (OS level) calls to handle file on your storage devices.

In C programming, file is a place on disk where a group of related data is stored.

File Operations

(i) Creating a new file.
(ii) Writing into a file.
(iii) Opening an existing file.
(iv) Reading a file.
(v) Closing a file.

Opening Files

You can use the fopen( ) function to create a new file or to open an existing file, this call will initialize an object of the type FILE, which contains all the information necessary to control the stream. Following is the prototype of this function call:

```
FILE *fopen("file_name", "Mode );
```

for example:

```
FILE *fp;
fp = fopen("MYABC.C", "r");
```

Above code will open a file MYABC.C in r mode (read only mode). The address of the first character is stored in pointer fp.

Writing a File

Following is the simplest function to write individual characters to a stream:

```
int fputc( int c, FILE *fp );
int fputs( const char *s, FILE *fp );
#include <stdio.h>
main()
{
FILE *fp;
}
```

Closing a File

To close a file, use the fclose( ) function.

The prototype of this function is:

```
int fclose( FILE *fp );
```

Here fclose() function closes the file and returns zero on success, or EOF if there is an error in closing the file. This **EOF** is a constant defined in the header file stdio.h.

No, it is a structure defined in stdio.h.

Q.Briefly discuss Opening Modes in Standard I/O?
Ans;
File can be opened in basic 3 modes : Reading Mode, Writing Mode, Appending Mode
- If File is not present on the path specified then New File can be created using Write and Append Mode.

File Opening Mode Chart :

Mode	Meaning	fopen Returns if FILE-	
		Exists	Not Exists
r	Reading	–	NULL
w	Writing	Over write on Existing	Create New File
a	Append	–	Create New File
r+	Reading + Writing	New data is written at the beginning overwriting existing data	Create New File
w+	Reading + Writing	Over write on Existing	Create New File
a+	Reading + Appending	New data is appended at the end of file	Create New File

Q.How can we working with file? Given with example? Exam:

Working with file
While working with file, you need to declare a pointer of type file. This declaration is needed for communication between file and program.

```
FILE *ptr;
```

Opening a file
Opening a file is performed using library function fopen(). The syntax for opening a file in standard I/O is:

```
ptr=fopen("fileopen","mode")
```

For Example:.

```
fopen("E:\\cprogram\program.txt","w");

/* ---------------------------------------------------- */
E:\\cprogram\program.txt is the location to create file.
"w" represents the mode for writing.
/* ---------------------------------------------------- */
```

Q. Why files are needed?
Ans:

When the program is terminated, the entire data is lost in C programming. If you want to keep large volume of data, it is time consuming to enter the entire data. But, if file is created, these information can be accessed using few commands.

There are large numbers of functions to handle file I/O in C language. In this tutorial, you will learn to handle standard I/O(High level file I/O functions) in C.

High level file I/O functions can be categorized as:

1. Text file
2. Binary file

Q. write a c program that will read a text file and count the total number of vowels within the text file. The file only contains uppercase latter? Exam:ACCE-2011

```
#include <stdio.h>
void main()
{
    FILE *fp;
int vowel=0,consonant=0;
char ch;
fp=fopen("name.txt","r");
if(fp==NULL)
{
printf("Source can't be opened");
exit(0);
}
ch=fgetc(fp);
while(ch!=EOF)
        {
if((ch=='a')||(ch=='A')||(ch=='e')||(ch=='E')||(ch=='i')||(ch=='I')||(ch=='o')
||(ch=='O')||(ch=='u')||(ch=='U'))
{ vowel++;
}
else
{
consonant++;
}  ch=fgetc(fp);
}
printf("\n Number of vowels are = %d",vowel);

return 0;
}
```

Q.Discuss you we can use File Read operation ?
Ans:

```
fp = fopen("test.txt", "w+");
fprintf(fp, "This is testing for fprintf...\n");
fputs("This is testing for fputs...\n", fp);
fclose(fp);
}
```

Q.Discuss fprintf() and fscanf() functions.
Ans:

The functions fprintf() and fscanf() are the file version of printf() and fscanf(). The only difference while using fprintf() and fscanf() is that, the first argument is a pointer to the structure FILE fprintf() can be used to write data in file.

Writing to a file

```
#include <stdio.h>
int main()
{
  int n;
  FILE *fptr;
  fptr=fopen("C:\\program.txt","w");
  if(fptr==NULL){
    printf("Error!");
    exit(1);
  }
  printf("Enter n: ");
  scanf("%d",&n);
  fprintf(fptr,"%d",n);
  fclose(fptr);
  return 0;
}
```

Reading a File

fscanf() can be used to read data from file.

```
#include <stdio.h>
int main()
{
  int n;
  FILE *fptr;
  if ((fptr=fopen("C:\\program.txt","r"))==NULL){
     printf("Error! opening file");
     exit(1);         /* Program exits if file pointer returns NULL. */
  }
  fscanf(fptr,"%d",&n);
  printf("Value of n=%d",n);
  fclose(fptr);
  return 0;
}
```

Q.Discuss about the preprocessor?
Ans:

Preprocessor directives are actually the instructions to the compiler itself. They are not translated but are operated directly by the compiler. Note that preprocessor statements begin with a #symbol, and are NOT terminated by a semicolon. Traditionally, preprocessor statements are listed at the beginning of the source file.They are also termed as macros.

The most common preprocessor directives are

- include directive
- define directive

1.iuclude directive:

The include directive is used to include files like as we include header files in the beginning of the program using #include directive like
#include<stdio.h>
#include<conio.h>

2.define directive:

It is used to assign names to different constants or statements which are to be used repeatedly in a program. These defined values or statement can be used by main or in the user defined functions as well.
They are used for
a. defining a constant
b. defining a statement
c. defining a mathematical expression

For example
#define PI 3.141593
#define TRUE 1
#define floatingpointno float

Q. Write a c function to evaluate the series or sine series? Exam:ACCE-2013

$$\sin(x) = x - \left(\frac{x^3}{3}\right) + \frac{x^5}{5} - \left(\frac{x^7}{7}\right)$$

Ans:

```
include<stdio.h>
#include<math.h>
#define ACC 0.000001
#define PI 3.14
Int main()
{      int x,i;
       float x1,x2,sine=0,term;
       printf("Enter the X value : ");
       scanf("%d",&x);
       x1=(x*PI)/180;
       x2=x1*x1;
       term=x1;
       sine=term;
       for(i=3;fabs(term)>ACC;i=i+2)
         {
            term = (x2*(-term))/(i*(i-1));
            sine = sine+term;
         }
            printf("\n%f",sine);
return 0;
}
Output :
Enter the X value : 100
0.984961
```