MODULE 5 SOLUTIONS

PART A

1- Define file and list basic operations of a file?

ANSWER-

A file is an abstract data type. To define a file properly, we need to consider the operations that can be performed on files.There are 6 types of basic operations of file

i) to create

ii) to write

iii) to read

Iv) to reposition

v) to delete

vi) to truncate files

2-Explain various text file opening modes?

ANSWER-

- ★ r (Opens an existing text file)
- ★ w (Opens a text file for writing if the file doesn't exist then a new file is created)
- ★ a (Opens a text file for appending(writing at the end of existing file) and create the file if it does not exist)
- ★ r+ (Opens a text file for reading and writing)
- ★ a+ (Open for reading and appending and create the file if it does not exist. The reading will start from the beginning, but writing can only be appended)
- ★ w+(Open for reading and writing and create the file if it does not exist. If the file exists then make it blank)

3- State the various types of status enquiry library functions in C?

ANSWER-

<assert.h> Program assertion functions
<ctype.h> Character type functions
<locale.h> Localization functions
<math.h> Mathematics functions
<setjmp.h> Jump functions
<signal.h> Signal handling functions
<stdarg.h> Variable arguments handling functions
<stdio.h> Standard Input/Output functions
<stdlib.h> Standard Utility functions
<string.h> String handling functions
<time.h> Date time functions

4-Explain ftell() function with an example?

ANSWER-

ftell() returns the current file position of the specified stream with respect to the starting of the file. This function is used to get the total size of a file after moving the file pointer at the end of the file.

EXAMPLE-

```
long int ftell(FILE *stream)          //(syntax of ftell)
```

5- Write the purpose of fseek() with an example?

ANSWER-

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

EXAMPLE-

```
#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

6- Write the syntax and usage of rewind( )?

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

SYNTAX-

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

7- Explain a file opening mode with an example?

ANSWER-

- r (open a file in read mode)
  `Ex: fp=fopen(“file.txt”,”r”)`
- w (opens or create a text file in write mode)
  `Ex: fp=fopen(“file.txt”,”w”)`
- a (opens a file in append mode)
  `Ex: fp=fopen(“file.txt”,”a”)`
- r+ (opens a file in both read and write mode)
  `Ex: fp=fopen(“file.txt”,”r+”)`
- a+ (opens a file in both read and write mode)
  `Ex: fp=fopen(“file.txt”,”a+”)`

- w+(opens a file in both read and write mode)
  `Ex: fp=fopen(“file.txt”,”w+”)`

8-List the different types of files?

ANSWER-

i) **Source files**: These files contain function definitions, and have names which end in .c

Example - foo.c

ii) **Header files:** These files contain function prototypes and various pre - processor statements and it ends with .h

Example - foo.h

iii) **Object files:** These files are produced as the output of the compiler and it ends with .o

Example - foo.o

iv) **Binary executables:** These are produced as the output of a program called a "linker". The linker links together a number of object files to produce a binary file which can be directly executed and it ends with .exe

Example - foo.exe

v) **Libraries:** A library is a compiled binary but is not in itself an an executable

9-List the application of files.

Applications of Files:

- Used in Computer Design
- Used in Networking Devices
- Used for making GUI Apps
- Used for designing Databases

10- predict the output of the following code?

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

```
{
char *str = "ZOHO";
while (*str)
{
putc(*str, stdout);
fputchar(*str);
printf("%c", *str);
str++;
}
return 0;
}
```

OUTPUT-
ZZZOOOHHHOOO

11- Predict the output of the following code?

```
#include <stdio.h>
int main()
{
FILE *fp = stdout;
stderr = fp;
fprintf(stderr, "%s", "hello");
}
```

OUTPUT-
hello

12- Find the output of this code.

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

```
FILE *fp = stdout;
int n;
fprintf(fp, "%d", 45);
}
```

OUTPUT-
45

13-Explain the error handling function for files in C?

ANSWER-

The header file “error. h” is used to print the errors using the return statement function. It returns -1 or NULL in case of any error and errno variable is set with the error code.It returns a value zero if no error has occurred and a non-zero value if there is an error.

14-Predict the output of this code.

```
#include <stdio.h>
#include <string.h>
int main()
{
char line[3];
fgets(line, 3, stdin);
printf("%d\n", strlen(line));
return 0;
}
```

OUTPUT-
IF INPUT IS “Hello”, OUTPUT IS “2”

15-Predict the content of 'file.c' after executing the following program?

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

```c
int main()
{
FILE *fp1, *fp2;
fp1=fopen("file.c", "w");
fp2=fopen("file.c", "w");
fputc('A', fp1); fputc('B', fp2);
fclose(fp1);
fclose(fp2);
return 0;
}
```

OUTPUT-

B

16-f the file 'source.txt' contains the line "Be my friend", predict the output of the below program?

```c
#include <stdio.h>
int main()
{
FILE *fs, *ft;
char c[10];
fs = fopen("source.txt", "r");
c[0] = getc(fs);
fseek(fs, 0, SEEK_END);
fseek(fs,-3L, SEEK_CUR);
fgets(c, 5, fs);
puts(c);
return 0;
}
```

OUTPUT-

Returns END string

17-Identify the error in the program.

```
#include<stdio.h>
#include<stdlib.h>
int main() {
unsigned char;
FILE *fp;
fp=fopen("trial", "r");
if(!fp) {
printf("Unable to open file");
exit(1);
}
fclose(fp);
return 0;
}
```

OUTPUT-
Char is not declared in the program

18-Explain why fseek() should be preferred over rewind()?

ANSWER-
fseek() can be used to check if the specified operations are done successfully. So, fseek() is preferred over rewind().

19- Differentiate between file opening mode r+ and w+?

ANSWER-
w+ truncate the file to zero length if it exists or create a new file whereas, r+ neither deletes the content nor creates a new file.

20- Predict the output of the following code?

```
#include<stdio.h>
int main()
{
char *str = "IARE";
while (*str)
```

```
{
putc(*str, stdout);
putchar(*str);
printf("%c", *str);
str++;
}
return 0;
}
```

OUTPUT-

IIIAAARRREEE

PART B

1- Write a C program to read a text file containing some paragraphs. Use fseek() function and read the text after skipping n characters from the beginning of the file.

```
#include <stdio.h>
 void main() {
    FILE *fp;
    fp = fopen("sample.txt", "r");
    char ch; int n;
    scanf("%d", &n);// Input value
    fseek(fp, n, SEEK_SET);
    while(1) {
        ch=fgetc(fp);
        printf("%c", ch);
        if (ch == EOF) {
            break;
        }
    }
fclose(fp);
}
```

Input:
This is a sample file for testing out C Programs n = 8
Output:
a sample file for testing out C Programs

2- Explain the following functions through a sample program that reads a file "test.txt".
a) ftell()
b) fseek()
c) rewind().

ANSWER-

fseek() - It is used to move the reading control to different positions using the fseek function.

ftell() - It tells the byte location of the current position in the file pointer.

rewind() - It moves the control to the beginning of a file.

```
#include <stdio.h>
void main() {
    FILE *fp;
    fp = fopen("sample.txt", "r");
    char ch;
    fseek(fp, 1, SEEK_SET);
    printf("%ld ", ftell(fp));
    rewind(fp);
    printf("%ld", ftell(fp));
    fclose(fp);
}
```

Input:

This is a sample file for testing out C Programs

Output: 1 0

3. Write a C program to read a text file "sample.txt" and print the following.

a) Substring of N characters from position i.

b) Reverse order of substring of N characters produced in a.

```
#include <stdio.h>
#define FILENAME "sample.txt"
int main(){
FILE *fptr;

fptr = fopen(FILENAME,"r");
int to_pos,n,counter=0;
scanf("%d",&n);
check:
  scanf("%d",&to_pos);
  int status = fseek(fptr,to_pos,SEEK_SET);
```

```
  if(status ==0){
    char string[n];
    while(!feof(fptr) && counter<n)
    {
      char ch = fgetc(fptr);
      printf("%c",ch);
      string[counter] = ch;
      counter += 1;
    }
    fclose(fptr);
    printf("\nThe reversed String is:\n");
    for(int i=counter-1;i>=0;i--){
      printf("%c",string[i]);
    }

  }
  else
    goto check;

}
```

4. Write the syntax of the following file I/O functions and Explain every option in each function with suitable example :

a. fopen()

b. fclose()

c. fread()

d. fwrite()

fopen(): The fopen() function is used to open a file and associates an I/O stream with it. This function takes two arguments. The first argument is a pointer to a string containing the name of the file to be opened while the second argument is the mode in which the file is to be opened.

SYNTAX: `FILE *fopen(const char *path, const char *mode);`

fclose(): the fclose() function closes a stream pointed to by stream. The fclose function flushes any unwritten data in the stream's buffer.

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

fread(): The fread() function generally used for binary files to read the binary data from the given file stream.
SYNTAX: `size_t fread(void *ptr, size_t size, size_t nmemb, FILE *stream)`

fwrite(): fwrite function writes a block of data to the stream. It will write an array of count elements to the current position in the stream. For each element, it will write size bytes.
SYNTAX: `size_t fwrite(const void *ptr, size_t size, size_t nmemb, FILE *stream);`

5- Write a program in C to create and store information in a text file.
Example : Input a sentence for the file : This is the content of the file test.txt.
Expected Output : The file test.txt created successfully...!!
Source Code:

```
#include <stdio.h>
void main() {
   FILE *fp;
   fp = fopen("sample.txt", "w");
   char ch[50]="This is the content of the file test.txt";
   fwrite(&ch, sizeof(ch),1,fp);
   printf("The file test.txt created successfully...!!");
   fclose(fp);
}
```

6- Write a C program to open a file named INVENTORY and store it in the following.

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

```
    int n,i;
    printf("Number of items : ");
    scanf("%d", &n);
    FILE *fp = fopen("INVENTORY.txt","w");
    fprintf(fp, "Data item\t\tNumber\t\tPrice\t\tQuantity\n");
    fprintf(fp, "\n");
    char item[100], number[100];
    int price, quantity;
    for (i=0; i<n; i++){
        scanf("%s %s %d %d", item, &number, &price, &quantity);
        fprintf(fp, "%s\t\t%s\t\t%d\t\t%d\n", item, number, price,
quantity);
    }
    fclose(fp);
}
```

7- Write a C program to read a given file, convert first letter of each word into uppercase and copy the contents of converted file into a new file

Source Code:

```
#include <string.h>
#include <ctype.h>

#define filename "file.txt"
#define newfilename "updated.txt"

int main(){
  FILE *fptr,*fptr1;
  fptr = fopen(filename,"r");
  fptr1 = fopen(newfilename,"w");

  int count = 0;

  while(!feof(fptr)){
    char ch = fgetc(fptr);
    if(ch != ' ' && islower(ch)){
      if(count == 0){
```

```
        fputc((char)(ch-32),fptr1);
      }
      else
        fputc(ch,fptr1);
      count += 1;
    }
    else{
      if(isupper(ch)){
        fputc(ch,fptr1);
        count += 1;
    }
      else{
        fputc(ch,fptr1);
        count = 0;
  }
}
  }
}
```

8- Write a C program to read the name and marks of n number of students from the user and store them in a file. If the file previously exists, then add the information of n students to the end of existing content.

Source Code:

```
#include <stdio.h>
int main() {
    char name[50];
    int marks,i,n;
    printf("Enter number of students: ");
    scanf("%d",&n);
    FILE *fptr;
    fptr=(fopen("C:\\student.txt","a"));
    if(fptr==NULL) {
        printf("Error!");
        exit(1);
    }
    for (i=0;i<n;++i) {
        printf("For student%d\nEnter name: ",i+1);
        scanf("%s",name);
```

```
        printf("Enter marks: ");
        scanf("%d",&marks);
        fprintf(fptr,"\nName: %s \nMarks=%d \n",name,marks);
    }
    fclose(fptr);
    return 0;
}
```

9- Write a C program to print the following from a given file:

a) Number of characters

b) Number of spaces

c) Number of tabs

d) Number of newlines

Source Code:

```
#include<stdio.h>
int main(){
    FILE *fptr1;
    char file1[] ="file1.txt";
    int ch, character = 0, line = 0, space = 0, tab = 0;
    fptr1 = fopen(file1,"r");
    while ((ch = fgetc(fptr1)) != EOF)
        {
            character++;
            if (ch == ' ')
                space++;
            if (ch == '\n')
                line++;
            if (ch == '\t')
                tab++;
        }
        fclose(fptr1);

        printf("\nNumber of characters = %d", character);
        printf("\nNumber of spaces = %d", space);
        printf("\nNumber of tabs = %d", tab);
```

```
        printf("\nNumber of lines = %d", line);
    }
```

10- Create a structure named employee containing name, age and basic pay. Write a C program to create 5 employee records and write to a file. Thenread the records from file and display it

Use the concept of Structures to make a data type for storing employee details.
Print those details on the file.

12- Write a program in C to read an existing file.

Example : Input the file name to be opened : test.txt
Expected Output :
The content of the file test.txt is : This is the content of the file test.txt.

Source Code:

```
#include <stdio.h>
int main ()
{
    FILE * fp;
    char str;
    fp = fopen ("file1.txt","r");
    str = fgetc(fp);
    while (str != EOF)
    {
      printf ("%c", str);
      str = fgetc(fp);
    }
    fclose (fp);
    return 0;
}
```

13- Write a program in C to write multiple lines in a text file.

Test Data :

Input : The number of lines to be written : 4

test line 1

test line 2

test line 3

test line 4

Expected Output :

The content of the file test.txt is :

test line 1

test line 2

test line 3

test line 4

Source Code:

```c
#include <stdio.h>
int main ()
{
    FILE * fp;
    int i,n;
    char str[20];
    printf("Enter the number of lines to be written: ");
    scanf("%d", &n);
    fp = fopen ("file1.txt","w");
    for(i = 0; i < n+1;i++)
    {
        char str[20];
    }
    fclose (fp);
    return 0;
}
```

14- Write a program in C to Find the Number of Lines in a Text File.

Source Code:

```c
#include <stdio.h>
#define MAX_FILE_NAME 100
```

```c
int main()
{

    FILE *fp;

    int count = 0;

    char filename[MAX_FILE_NAME];

    char c;    printf("Enter file name: ");
    scanf("%s", filename);
    fp = fopen(filename, "r");
    if (fp == NULL) {
        printf("Could not open file %s", filename);
        return 0;
    }
    // Extract characters from file and store in character c
    for (c = getc(fp); c != EOF; c = getc(fp))
        if (c == '\n')
            count = count + 1;

    fclose(fp);

    printf("The file %s has %d lines\n ", filename, count);
    return 0;
}
```

OR

```c
#include <stdio.h>
#define FILENAME "file.txt"
int main()
{
  FILE *fp;
  char ch;
  int c=0;
  fp=fopen(FILENAME,"r");
  if(fp==NULL)
  {
```

```
    printf("File does not exist",);
    return -1;
  }
  while((ch=fgetc(fp))!=EOF)
  {
    if(ch=='\n')
      c++;
  }
  fclose(fp);
  printf("Total number of lines are: %d\n",c);
  return 0;
}
```

15- Write a program in C to count a number of words and characters in a file.

Source Code:

```
#include <stdio.h>
#define FILENAME "file.txt"
int main()
{
  FILE *fp;
  char ch;
  int wrd=1,charctr=1;
  fp=fopen(FILENAME,"r");
  if(fp==NULL)
  {
    printf("File does not exist");
    return -1;
  }
  else
        {
        ch=fgetc(fp);
        while(ch!=EOF)
            {
                printf("%c",ch);
                if(ch==' '||ch=='\n')
                    {
                        wrd++;
```

```
                    }
                    else
                    {
                        charctr++;
                    }
                ch=fgetc(fp);
            }
        printf("\n The number of words in the  file are :
%d\n",wrd-2);
        printf(" The number of characters in the  file  are :
%d\n",charctr-1);
        }
  fclose(fp);
    return 0;
}
```

16- Write a program in C to find the content of the file and number of lines in a Text File.

Test Data :

Input :

The filename to be opened : test.txt

Expected Output:

The content of the file test.txt are :

test line 1

test line 2

test line 3

test line 4

The lines in the file are : 4

Source Code:

17- Write a program in C to delete a specific line from a file

Source Code:

```c
#include<stdio.h>
int main(){
    FILE *fptr1, *fptr2;
    char file1[] ="file1.txt";
    char file2[] ="file2.txt";
    char curr;
    int del, linenum = 0;
    printf("Please enter the line number you want to delete : ");
    scanf("%d", &del);
    fptr1 = fopen(file1,"r");
    fptr2 = fopen(file2, "w");
    curr = getc(fptr1);
    if(curr!=EOF)
    {linenum =1;}
    while(1){
        if(del != linenum)
        {putc(curr, fptr2);}
        curr = getc(fptr1);
        if(curr =='\n')
        {linenum++;}
        if(curr == EOF)
        {break;}
    }
    fclose(fptr1);
    fclose(fptr2);
}
```

18- Write a program in C to replace a specific line with another text in a file

Source Code:

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

#define MAX 256

  int main()
  {
        FILE *fptr1, *fptr2;
```

```
        int lno, linectr = 0;
        char str[MAX],fname[MAX];
        char newln[MAX], temp[] = "temp.txt";

    printf("\n\n Replace a specific line in a text file with a new
text :\n");

printf("-------------------------------------------------------------
\n");
    printf(" Input the file name to be opened : ");
        fgets(fname, MAX, stdin);
        fname[strlen(fname) - 1] = '\0';
        fptr1 = fopen(fname, "r");
        if (!fptr1)
        {
                printf("Unable to open the input file!!\n");
                return 0;
        }
        fptr2 = fopen(temp, "w");
        if (!fptr2)
        {
                printf("Unable to open a temporary file to
write!!\n");
                fclose(fptr1);
                return 0;
        }
        printf(" Input the content of the new line : ");
        fgets(newln, MAX, stdin);
        printf(" Input the line no you want to replace : ");
        scanf("%d", &lno);
        lno++;

        while (!feof(fptr1))
        {
            strcpy(str, "\0");
            fgets(str, MAX, fptr1);
            if (!feof(fptr1))
            {
                linectr++;
```

```
                if (linectr != lno)
                    {
                        fprintf(fptr2, "%s", str);
                    }
                    else
                    {
                        fprintf(fptr2, "%s", newln);
                    }
                }
        }
        fclose(fptr1);
        fclose(fptr2);
        remove(fname);
        rename(temp, fname);
        printf(" Replacement did successfully..!! \n");
        return 0;
  }
```

19- Write a program in C to copy a file in another name

Source Code:

```
#include<stdio.h>
int main(){
    FILE *fptr1, *fptr2;
    char file1[] ="file1.txt";
    char file2[] ="file2.txt";
    char ch;
    fptr1 = fopen(file1,"r");
    fptr2 = fopen(file2, "w");
    while ((ch = getc(fptr1)) != EOF)
        fputc(ch, fptr2);
    printf("File copied successfully.\n");
    fclose(fptr1);
    fclose(fptr2);
}
```

20- Write a program in C to merge two files and write it in a new file

Source Code:

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

int main() {
   FILE *fp1 = fopen("file1.txt", "r");
   FILE *fp2 = fopen("file2.txt", "r");
   FILE *fp3 = fopen("file3.txt", "w");
   char c;
   if (fp1 == NULL || fp2 == NULL || fp3 == NULL) {
         puts("Could not open files");
         exit(0);
   }

   while ((c = fgetc(fp1)) != EOF)
      fputc(c, fp3);
   while ((c = fgetc(fp2)) != EOF)
      fputc(c, fp3);

   printf("Merged file1.txt and file2.txt into file3.txt");
   fclose(fp1);
   fclose(fp2);
   fclose(fp3);
   return 0;
}
```

PART C

1- C program to read the name and marks of a number of students and store them in a file?

ANSWER-

```
#include <stdio.h>
int main() {
	char name[50];
	int marks,i,n;
	printf("Enter number of students: ");
	scanf("%d",&n);
	FILE *fptr;
	fptr=(fopen("C:\\student.txt","w"));
	if(fptr==NULL) {
		printf("Error!");
		exit(1);
	}
	for (i=0;i<n;++i) {
		printf("For student%d\nEnter name: ",i+1);
		scanf("%s",name);
		printf("Enter marks: ");
		scanf("%d",&marks);
		fprintf(fptr,"\nName: %s \nMarks=%d \n",name,marks);
	}
	fclose(fptr);
	return 0;
}
```

2,3,4 Left Out

5- Write a C Program to Reverse the Contents of a File and Print it.

ANSWER-

```c
#include<stdio.h>
#include<conio.h>
int main()
{
    FILE *fp;
    char ch, fname[30], newch[500];
    int i=0, j, COUNT=0;
    printf("Enter the filename with extension: ");
    gets(fname);
    fp = fopen(fname, "r");
    if(!fp)
    {
        printf("Error in opening the file...\nExiting...");
        getch();
        return 0;
    }
    printf("\nThe original content is:\n\n");
    ch = getc(fp);
    while(ch != EOF)
    {
        COUNT++;
        putchar(ch);
        newch[i] = ch;
        i++;
        ch = getc(fp);
    }
    printf("\n\n\n");
    printf("The content in reverse order is:\n\n");
    for(j=(COUNT-1); j>=0; j--)
    {
        ch = newch[j];
        printf("%c", ch);
    }
    printf("\n");
    getch();
    return 0;
}
```

6-Write a C Program to Count No of Lines, Blank Lines, and Comments in a given file.

ANSWER-

```
#include <stdio.h>

int main()
{
    int line_count = 0, n_o_c_l = 0, n_o_n_b_l = 0, n_o_b_l = 0,
n_e_c = 0;
    FILE *fp1;
    char ch;
    char * arrr = "C:\\Users\\acer\\Documents\\file4.txt";
    fp1 = fopen(arrr, "r");

    while ((ch = fgetc(fp1))!= EOF)
    {
        if (ch  ==  '\n')
        {
            line_count++;
        }
        if (ch  ==  '\n')
        {
            if ((ch = fgetc(fp1))  ==  '\n')
            {
                fseek(fp1, -1, 1);
                n_o_b_l++;
            }
        }
        if (ch  ==  ';')
        {
            if ((ch = fgetc(fp1))  ==  '\n')
            {
                fseek(fp1, -1, 1);
                n_e_c++;
            }
        }
    }
```

```
    fseek(fp1, 0, 0);
    while ((ch = fgetc(fp1))!= EOF)
    {
        if (ch  ==  '/')
        {
            if ((ch = fgetc(fp1))  ==  '/')
            {
                n_o_c_l++;
            }
        }
    }
    printf("Total no of lines: %d\n", line_count);
    printf("Total no of comment line: %d\n", n_o_c_l);
    printf("Total no of blank lines: %d\n", n_o_b_l);
    printf("Total no of non blank lines: %d\n", line_count-n_o_b_l);
    printf("Total no of lines end with semicolon: %d\n", n_e_c);

    return 0;
}
```

7-Predict the output of the following code

ANSWER-

```
#include<stdio.h>
int main()
{
int f1, f2;
FILE *fp;
fp = fopen("datafile.txt", "w");
f1 = EOF;
f2 = feof(fp);
if(f1 == f2)
{
printf("EOF and feof(), both returns
the same value");
}
else
{
```

```
printf("EOF and feof() both
returns different values");
}
return 0;
}
```

ANSWER- ERROR COMPILING

8- Write a C program to read the name and marks of n number of students from the user and store them in a file. If the file previously exists, then add the information of n students to the end of existing content

```
#include <stdio.h>
int main() {
  char name[50];
  int marks,i,n;
  printf("Enter number of students: ");
  scanf("%d",&n);
  FILE *fptr;
  fptr=(fopen("file1.txt","a"));
  for (i=0;i<n;++i) {
    printf("Enter student name: ",i+1);
    scanf("%s",name);
    printf("Enter marks: ");
    scanf("%d",&marks);
    fprintf(fptr,"\nName: %s \nMarks=%d \n",name,marks);
  }
  fclose(fptr);
  return 0;
}
```

9- Write a C program to copy a number of bytes from a specific offset to another file.

ANSWER-

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

```c
#include <string.h>

int main(int argc,char *argv[])
{
    FILE *fp1;
    FILE *fp2;

    int count       = 0;
    int location    = 0;
    int totBytes    = 0;

    unsigned char data[1024];

    if( argc < 5 )
    {
        printf("Insufficient Arguments!!!\n");
        printf("Please use \"program-name source-file-name
target-file-name offset N\" format.\n");
        return -1;
    }

    fp1 = fopen(argv[1],"r");
    if( fp1 == NULL )
    {
        printf("\n%s File can not be opened : \n",argv[1]);
        return -1;
    }

    fseek(fp1,0,SEEK_END);

    count    = ftell(fp1);
    location = atoi(argv[3]);       // offset of source file to copy
    totBytes = atoi(argv[4]);       // number of bytes to copy

    if( count < (location  + totBytes) )
    {
        printf("\nGiven number of bytes can not be copy, due to file
size.\n");
        return -1;
```

```
    }

    fp2 = fopen(argv[2],"w");
    if( fp2 == NULL )
    {
        printf("\n%s File can not be opened\n",argv[2]);
        return -1;
    }

    fseek(fp1,location,SEEK_SET);

    fread(data,totBytes,1,fp1);

    fwrite(data,totBytes,1,fp2);

    data[totBytes]=0;

    printf("\nCopied content is : \"%s\"\n",data);

    fclose(fp1);
    fclose(fp2);

    return 0;
}
```

10- Write a program in C to read the file and store the lines into an array

ANSWER-

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

#define LSIZE 128
#define RSIZE 10

int main(void)
{
    char line[RSIZE][LSIZE];
```

```
      char fname[20];
    FILE *fptr = NULL;
    int i = 0;
    int tot = 0;
    printf("\n\n Read the file and store the lines into an array
:\n");

printf("-------------------------------------------------------\n");
      printf(" Input the filename to be opened : ");
      scanf("%s",fname);

    fptr = fopen(fname, "r");
    while(fgets(line[i], LSIZE, fptr))
      {
         line[i][strlen(line[i]) - 1] = '\0';
         i++;
    }
    tot = i;
      printf("\n The content of the file %s  are : \n",fname);
    for(i = 0; i < tot; ++i)
    {
        printf(" %s\n", line[i]);
    }
    printf("\n");
    return 0;
}
```