

Name	Adwait S Purao
UID no.	2021300101
Experiment No.	10

AIM:	Implement various operations on files to solve a given problem.
Program 1	
PROBLEM STATEMENT :	A publishing company holds in file detail of all the books they publish. However, in the future, they wish to maintain two distinct files (i) paperbacks (ii) hardbacks. Write a program that reads a file containing details of both paperback and hardback books and creates two files as specified above. Assume that the first character in each input record indicates if the book is paperback(p) or hardback(h) or both(b).
ALGORITHM:	<ol style="list-style-type: none"> 1. START 2. Define structure book_t containing character variable type, character arrays name, and author 3. Define int function print having character array filename[] as a parameter. 4. Initialize three file pointers fp,fp1, and fp2. 5. Use fp to open filename in reading mode 6. If fp is equal to NULL print file not found return 0 7. Initialize l to 0 8. Define book_t variable b 9. Use fp1 to open Paperbacks.txt in write mode and fp2 to open Handbacks.txt in write mode 10. While(file scan of fp for b.type, b.name, and b.author is not equal to EOF) If(file scan is equal to 'p') file write of fp1 printing b.name and b.author else If(file scan is equal to 'h') file write of fp2 printing b.name and b.author Else If(file scan is equal to 'b') file write of fp1 printing b.name and b.author file write of fp2 printing b.name and b.author 11. Close fp1 12. Close fp2 13. Close fp 14. Return 0 15. Define int main() 16. Declare character array filename[20] 17. Input filename 18. Concatenate filename with ".txt" 19. l = print(filename) 20. Return 0

21. STOP

PROGRAM:

```
#include <stdio.h>
#include <string.h>
typedef struct book
{
    char type;
    char name[50];
    char author[20];
} book_t;
int print(char filename[])
{
    FILE *fp, *fp1, *fp2;
    fp = fopen(filename, "r");
    if (fp == NULL)
    {
        printf("File Not Found");
        return 0;
    }
}
```

```
    }  
    book_t b;  
    fp1 = fopen("Paperbacks.txt", "w");  
    fp2 = fopen("Hardbacks.txt", "w");  
    while (fscanf(fp, " %c,  
    %[^,],%[^\\n]\\n", &b.type, b.name,  
    b.author) != EOF)  
    {  
        if (b.type == 'b')  
        {  
            fprintf(fp1, "%s, %s\\n",  
b.name, b.author);  
            fprintf(fp2, "%s, %s\\n",  
b.name, b.author);  
        }  
        else if (b.type == 'p')  
            fprintf(fp1, "%s, %s\\n",  
b.name, b.author);  
        else if (b.type == 'h')  
            fprintf(fp2, "%s, %s\\n",  
b.name, b.author);  
    }  
    fclose(fp1);
```

```
        fclose(fp2);
        fclose(fp);
        return 0;
    }
    int main()
    {
        int i;
        char filename[20];
        printf("Enter the name of the file
where the records are kept: ");
        scanf("%s", filename);
        strcat(filename, ".txt");
        i = print(filename);
        return 0;
    }
```

RESULT:

```
Enter the name of the file where the records are kept: Books
PS C:\Users\aspur\C PROGRAMS\mydirectory> █
```

```

structures3.c  structures3.c  Events.txt  Hardbacks.txt  Paperbacks.txt
directory > Hardbacks.txt
1  Raja Shivchhatrapati, Babasaheb Purandare
2  Chhava, Shivaji Sawant
3  Majhi Janmathep, Veer Savarkar
4  My inventions, Nikola Tesla
5  Life's Amazing Secrets, Gaur Gopal Das

structures2.c  Paperbacks.txt  Books.txt  nPr.c  recursion_oddevensum.c  p
mydirectory > Paperbacks.txt
1  Chhava, Shivaji Sawant
2  My inventions, Nikola Tesla
3  The world as I see It, Albert Einstien
4  Life's Amazing Secrets, Gaur Gopal Das
5  The Monk who sold his ferrari, Robin Sharma
6

```

Program 2

PROBLEM STATEMENT :

Set up a file containing vehicle records which hold registration number and owner information (name and address). Write a program which, given a vehicle's registration number, will rapidly retrieve and print the owner information.

ALGORITHM:

1. START
2. Define structure vehicle_v containing character arrays reg[7], name[30] and addr[30] as variables
3. Define int function print with character array filename[] as a parameter.
4. Define FILE pointer fp
5. Use fp to open filename in reading mode
6. If fp is equal to NULL print File not Found return 0
7. Close fp

8. Declare vehicle_t variable v, integer variables n and flag =1, and character array rg[7]
9. Input no of searches n
10. Loop from l = 0 to 6
a. Use fp to open filename in reading mode
b. Input registration number rg
c. While (file scan of fp for v.reg, v.name and v.addr) is not equal to EOF If (strcmp(v.reg,rg) is equal to 0) print v.reg, v.name, and v.addr flag = 0 break d. If (flag) print not found e.
Close fp
11. Return 0
12. Define int main()
13. Input character array filename
14. Concatenate ".txt" to filename
15. Int l = print(filename)
16. Return 0
17. STOP

**PROGRA
M:**

```
#include<stdio.h>
#include<string.h>
typedef struct vehicle
{
    char reg[7];
    char name[30];
    char addr[30];
}vehicle_t;
int print(char filename[])
```

```
{
    FILE *fp;
    fp=fopen(filename,"r");
    if (fp == NULL)
    {
        printf("FILE NOT FOUND!\n");
        return 0;
    }
    fclose(fp);
    vehicle_t v;
    int n,flag=1;
    char rg[7];
    printf("Enter the number of records you
want to search: ");
    scanf("%d",&n);
    for(int i=0;i<n;i++)
    {
        fp = fopen(filename,"r");
        printf("\nEnter the registration
number: ");
        scanf(" %s",rg);

        while(fscanf(fp,"%[^,],%[^,],%[^\\n]\\n",v
```

```
.reg,v.name,v.addr)!=EOF)
    if(strcmp(v.reg,rg)==0)
    {
        printf("%s %s
%s\n",v.reg,v.name,v.addr);
        flag=0;
        break;
    }
    if (flag)
        printf("Not Found\n");
        flag = 1;
        fclose(fp);
    }
    return 0;
}
int main()
{
    int i;
    char filename[20];
    printf("Enter the name of the file: ");
    scanf(" %s",filename);
    strcat(filename, ".txt");
    i = print(filename);
```



```
return 0;  
}
```

RESULT:

```
PS C:\Users\aspur\C PROGRAMS> cd "c:\Users\aspur\C PROGRAMS\mydire  
Enter the name of the file: Drivers  
Enter the number of records you want to search: 3  
  
Enter the registration number: YI1234  
Not Found  
  
Enter the registration number: ER0934  
ER0934 Elon Musk California  
  
Enter the registration number: GH7890  
GH7890 Virat Kohli Delhi  
PS C:\Users\aspur\C PROGRAMS\mydirectory> █
```

```

.c  structures3.c  files.c  Drivers.txt  files2.c  Hardbacks
directory > Drivers.txt
1  AD1235,Adwait Purao,Kurla
2  AB9887,Swanil Joshi,Mulund
3  GS3214,Sachin Tendulkar,Bandra
4  GH7890,Virat Kohli,Delhi
5  ER0934,Elon Musk,California

```

Program 3

PROBLEM STATEMENT:

A file EVENTS consists of records of events that have occurred during one calendar year. Each record describes one event and starts with a date field giving the number of the day in the year during which the event occurred. The file is ordered in ascending order by this day number. A file SELECT consists of single-field records, the field being a day no. The file too is ordered in ascending sequence by this field. Write a program that reads these 2 files and produces a report which shows, for each date in SELECT, the no. of events which occurred on that date. For example Day Events 13 12 45 6 352 4

ALGORITHM:

1. START
2. Define void function check with two character arrays filename1 and filename2 as parameters.
3. Define FILE pointers fp1 and fp2
4. Use f1 to open filename1 in reading mode and f2 to open filename2 in reading mode
5. While(file scan of fp2 for day1 is not equal to EOF)
 - a. While (file scanf of fp1 for day2 and events is not equal to EOF)
 - if(day1 is equal to day2) count++
 - else print day1 and count count = 1 break
6. Print day1 and count
7. Define int main()
8. Scanf filename1 and filename2
9. Concatenate ".txt" to filename1 and filename2 both
10. Call function print(filename1, filename2)
11. Return 0
12. STOP

PROGRAM:	<pre>#include<stdio.h> #include<string.h> void check(char filename1[], char filename2[]) { int day1, day2, count=0; char events[20]; FILE *fp1, *fp2; fp1=fopen(filename1,"r"); fp2=fopen(filename2,"r"); while(fscanf(fp2,"%d\n",&day1)!=EOF) { while(fscanf(fp1,"%d, %[^\\n]\\n",&day2,events)!=EOF) { if(day2==day1) count++; else { printf("%d %d\n",day1,count); count=1; break; } } } printf("%d %d\n",day1,count); }</pre>

```
}  
int main()  
{  
    char filename1[20], filename2[20];  
    printf("Enter the filename containing the  
events: ");  
    scanf("%s",filename1);  
    strcat(filename1, ".txt");  
    printf("Enter the filename containing the  
dates: ");  
    scanf("%s",filename2);  
    strcat(filename2, ".txt");  
    check(filename1,filename2);  
    return 0;  
}
```

RESULT:

```
PS C:\Users\aspur\C PROGRAMS> cd "c:\Users\aspur\C PROGRAMS\mydirectory\" ; if ($?) { gcc files3.c -o files3 } ; if ($?) { .\files3 }  
Enter the filename containing the events: Events  
Enter the filename containing the dates: Dates  
2 2  
48 2  
89 2  
70 3  
PS C:\Users\aspur\C PROGRAMS\mydirectory> █
```

```
structures3.c  Events.txt  Dates.txt
directory > Events.txt
1 2, Sphoorthi
2 2, Oculus
3 48, Tug of War
4 48, Aelan-Ae-Jung
5 89, Face Cup
6 89, Freshers Party
7 70, Start-A-Thon
8 70, SPIT MUN
9 70, Techrace

directory > Dates.txt
1 2
2 48
3 89
4 70
```

CONCLUSION:

In the above experiment we learned about the basic operations we can perform on files .We learned about fopen , fclose , fscanf, fprintf. We learned about EOF.