(a) Write a program to read a file and display its contents along with line numbers before each line.

(b) Write a program to append the contents of one file at the end of another.

(c) Suppose a file contains student's records with each record containing name and age of a student. Write a program to read these records and display them in sorted order by name.

(d) Write a program to copy contents of one file to another. While doing so replace all lowercase characters to their equivalent uppercase characters.

(e) Write a program that merges lines alternately from two files and writes the results to new file. If one file has less number of lines than the other, the remaining lines from the larger file should be simply copied into the target file.

(f) Write a program to encrypt/decrypt a file using:

(1) Offset cipher: In this cipher each character from the source file is offset with a fixed value and then written to the target file.

For example, if character read from the source file is 'A', then write a character represented by 'A' + 128 to the target file.

(2) Substitution cipher: In this cipher each for character read from the source file a corresponding predetermined character is written to the target file.

For example, if character 'A' is read from the source file, then a '!' would be written to the target file. Similarly, every 'B' would be substituted by '5' and so on.

(g) In the file 'CUSTOMER.DAT' there are 100 records with the following structure:

```
struct  customer
{
   int  accno ;  char  name[ 30 ] ;  float  balance ;
} ;
```

In another file 'TRANSACTIONS.DAT' there are several records with the following structure:

```
struct  trans
{
   int  accno ;  char  trans_type ;  float  amount ;
} ;
```

The element **trans_type** contains D/W indicating deposit or withdrawal of amount. Write a program to update 'CUSTOMER.DAT' file, i.e., if the **trans_type** is 'D' then update the **balance** of 'CUSTOMER.DAT' by adding **amount** to balance for the corresponding **accno**. Similarly, if **trans_type** is 'W' then subtract the **amount** from **balance.** However, while subtracting the amount ensure that the amount should not get overdrawn, i.e., at least 100 Rs. should remain in the account.

(h) There are 100 records present in a file with the following structure:

```
struct  date
{
   int  d, m, y ;
} ;

struct  employee
{
   int  empcode[ 6 ] ;  char  empname[ 20 ] ;
   struct  date join_date ;  float  salary ;
};
```

Write a program to read these records, arrange them in ascending order by **join_date** and write them to a target file.

(i) A hospital keeps a file of blood donors in which each record has the format:

Name: 20 Columns
Address: 40 Columns
Age: 2 Columns
Blood Type: 1 Column (Type 1, 2, 3 or 4)

Write a program to read the file and print a list of all blood donors whose age is below 25 and whose blood type is 2.

(j) Given a list of names of students in a class, write a program to store the names in a file on disk. Make a provision to display the **n**th name in the list, where **n** is read from the keyboard.

(k) Assume that a Master file contains two fields, roll number and name of the student. At the end of the year, a set of students join the class and another set leaves. A Transaction file contains the roll numbers and an appropriate code to add or delete a student.

Write a program to create another file that contains the updated list of names and roll numbers. Assume that the Master file and the Transaction file are arranged in ascending order by roll numbers. The updated file should also be in ascending order by roll numbers.

(l) Given a text file, write a program to create another text file deleting the words "a", "the", "an" and replacing each one of them with a blank space.