1.	Write a program to display the contents of a given file.
2.	Receives records from keyboard and writes them to a file.
3.	Write a program to copy one file to another file.
4.	Rewrite the above program with command line argument. The source and destination file name are taken from command line.
5.	Count chars, spaces, tabs and new lines in a file. At the end of the file program should display appropriate message.
6.	Count number of words in a line. Words are separated by whitespace characters.
7.	Write a program that reads student data (roll no, name, age) from keyboard for 5 students and write them into file. The program then reads the data from the same file and displays on screen.
8.	Write a program to append one file at the end of other.
9.	Write a program to append a file to itself.
10.	Receives records from keyboard and writes them to a file in binary mode.
11.	Reads records from binary file and display them on screen.
12.	Write a program to demonstrate fseek() function.
13.	Write a program illustrating the use of error handling function.
14.	Write a program to accept a file of verifying length lines and changes it to a formatted file with 60 characters in each line.
15.	Write a program that calculates average number of character per line in a file.
16.	Write a program to delete last line of any file.
17.	Write a program to delete blank lines in a file.
18.	Write a program to insert a blank line after the seventh line in a file.
19.	Write a program to that copies one text file to another and insert blank lines between paragraphs in the new file. Paragraphs are identified by a newline character.
20.	Write a program to copy only lines beginning with user specified character.
21.	Write a program to delete sixth line in a file. Do not change sixth line as a blank line. Delete it completely.
22.	Write a program to insert blank line after each line in a file.
23.	Write a program to duplicate the fourth line in a file.
24.	Write a program to copy a file, insert two space characters at the beginning of each line.
25.	Write a program that inserts 21st character of each line in a file to a new line. All extracted character are to on the same file. If a line in input file has fewer than 21 characters, write the last character. If line is blank then copy nothing. At the end of file write a new line to the new file and close it.
26.	Write a program that reads the first word of each line of input and prints them out at the end.

27.	Write a program to read a file and display contents with its line numbers.
28.	Write a program to change all uppercase character in a file to lowercase, and vice-versa.
29.	Write a program which writes whole ASCII set in the file "ASC_SET". After that program reads and prints only digits and uppercase alphabets from the file.
30.	Write a program which stores 10 integers in a file and then reads them and stores all the positive number in one file, while all negative numbers in another file.
31.	Write a program that reads the odd number between 300 to 500 to a text file.
32.	Write a program that writes the multiple of number between lowlimit and highlimit to a text file. Assume that lowlimit < highlimit and number is not equal to zero.
33.	Write a program to read a score from a file, count the score over 90, copy the score over 90 to a new file and print the number of score over 90 to the monitor.
34.	Write a program to find the size of a text file without traversing it character by character.
35.	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.
36.	Write a program to carry out the following: - To read a text file "TRIAL.TXT" consisting of a maximum of 50 lines of text, each line with a maximum of 80 characters. - Count and display the number of words contained in the file. - Display the total number of four letter words in the text file. Assume that the end of a word may be a space, comma or a full-stop followed by one or more spaces or a newline character.
37.	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.
38.	Write a program to carry out the following: (a) Read a text file 'INPUT.TXT' (b) Print each word in reverse order Example, Input: INDIA IS MY COUNTRY Output: AIDNI SI YM YRTNUOC Assume that each word length is maximum of 10 characters and each word is separated by newline/blank characters.
39.	Write a program to read a list of words, sort the words in alphabetical order and display them one word per line. Also give the total number of words in the list. Output format should be: Total Number of words in the list is _____ Alphabetical listing of words is:

	------ ------ Assume the end of the list is indicated by ZZZZZZ and there are maximum in 25 words in the Text file.
40.	Write a program to encrypt/decrypt a file using: (1) An offset cipher: In an offset 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 convert this into a new character by offsetting 'A' by a fixed value, say 128, and then writing the new character to the target file. (2) A substitution cipher: In this each character read from the source file is substituted by a corresponding predetermined character and this character is written to the target file. For example, if character 'A' is read from the source file, and if we have decided that every 'A' is to be substituted by '!', then a '!' would be written to the target file in place of every 'A' Similarly, every 'B' would be substituted by '5' and so on.
41.	Write a text analyser program that reads the text file. The program prints a menu that gives the user the options of counting lines, words, characters or sentences or all of the above provide a separate function for each option. At the end of analysis write an appropriate report.
42.	Write a menu driven text utility program. The program will have following capabilities: 1. Copy a user named file to a new file. 2. Change the file format to a double space 3. Remove all blank lines 4. Display the content of file as a series of 60 character lines with no words split between lines.
43.	Write a program to store an array of real numbers in file and read them.
44.	Write a program which enters N student records in a file in binary form.
45.	Write a program to print all the records of file created in above program, append a record and update the record.
46.	Write a program to find smallest integer from the N integer stored in file "INT"
47.	Read and Write records to a file using structure.
48.	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.
49.	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 of join_date and write them in to a target file.

50. 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 (n is data to be read) and to display all names starting with S.

51. Create an inventory file using the data shown below:

Part no	Price	Quantity on hand	Reorder point	Minimum order
0123	1.23	23	20	20
0234	2.34	34	50	25
3456	34.56	56	50	10
4567	45.67	7	10	5
5678	6.78	75	75	25

Write a program to read the inventory file and create an inventory report. The report will contain the part number, price, quantity on hand, reorder point, minimum order and order amount. The order amount is calculated when the quantity on hand falls below the reorder point. It is calculated as the sum of reorder point and the minimum order less the quantity on hand. Provide a report heading such as “inventory report” captions for each column and an “end of report” message at the end of report. Print the part number with leading zeroes.

52. Create an employee file as shown in below table:

Employee no	Department	Pay rate	Exempt	Hours worked
0101	41	8.11	Y	49
0722	32	7.22	N	40
1273	23	5.43	Y	39

	<table><tr><td>2584</td><td>14</td><td>6.74</td><td>N</td><td>45</td></tr></table> Write a program that read the employee file and create a payroll register. The register will contain the following data. 1. Employee number 2. Department 3. Pay rate 4. Exempt 5. Hours worked 6. Base pay 7. Over time (over time pay is only for non exempt employees. An employee is exempt if Y appears in the exempt column. Over time is paid at time and one half for all hours worked over 40. 8. Total pay
53.	Write a program to record information about stock received at the store. The user enter the information about each product received the item name, the price per user, the price per unit, the quantity and date received and the supplier. Each product is represented by a structure which is stored as a record in a file called "inventory". The structure also contains a member named "count". Which is used to record the number of units the number of unit of the product in the store at any time. This member is initialized to "quantity received" and would be decremented by 1 each time a unit of the product purchased.
54.	Write a program that uses the inventory file produced in above program. The user present the program with a shopping list of items to be purchased. The program inputs the stored inventory file and then it decrements the count for each item on the shopping list and charges the customer appropriately. The program must check for the presence of the item and for the sufficient quantities in stock. Before terminating the program, it must update the inventory file to reflect the depletion in stock. It might also print a list of items that need to be restocked.
55.	You are given a data file EMPLOYEE.DAT with the following record structure: struct employee { int empno ; char name[30] ; int basic, grade ; } ; Every employee has a unique empno and there are supposed to be no gaps between employee numbers. Records are entered into the data file in ascending order of employee number, empno. It is intended to check whether there are missing employee numbers. Write a program segment to read the data file records sequentially and display the list of missing employee numbers.
56.	In a small firm employee numbers are given in serial numerical order that is 1, 2, 3, etc.

	Create a file of employee data with following information: - Employee number, name, sex, gross salary. - If more employees join, append their data to the file. - If an employee with serial number 25 (say) leaves, delete - The record by making gross salary 0. -If some employee's gross salary increases, retrieve the record and update the salary. Write a program to implement the above operations.
57.	Write a payroll program which reads the employee information from a file and prints their pay slip. The pay slips are stored in pay.lst file.
58.	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 ; } ;space The parameter 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 make sure that the amount should not get overdrawn, i.e. at least 100 Rs. Should remain in the account.
59.	Assume that a Master file contains two fields, Roll no. 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.