1.	PURPOSE	PROGRAM NAME
2.	Printing anjul	Anjul.c
3.	Equivalence or partial order matrix	Array2.c
4.	Entering an array of words and altering a string at some position of the array	array_of_words.c
5.	Average mark of a class	Average.c
6.	Bubble sort	Bubble.c
7.	Game of chomp	Chomp.c
8.	Printing ref, sym and trans closures of a matrix	Closure.c
9.	Finding x^n after entering two polynomials	Coeff.c
10.	Printing days, months and years after entering days	Days.c
11.	*Finding determinant of any order matrix	Determinent.c
12.	Noticing functioning of escape sequences	escape_sequence.c
13.	Factorial from iteration	Factorial.c
14.	Factorial from recursion	Factre.c
15.	Fibonacci from iteration	Fibonacci.c
16.	Matchstick game	Game.c
17.	Printing grade after entering the percentage	Grade.c
18.	Tower of hanoi with recursion	Hanoi.c
19.	Printing inverse of a 3*3 matrix	Inverse.c
20.	Race b/w hare and turtle	Kachua.c
21.	Parallel downloads time required(simple)	Lab8_1.c
22.	Parallel downloads time required with speed sharing	Lab8_2.c
23.	Printing top 3 and bottom 3 after entring marks for each student in the class	Lab8_3.c
24.	Finding transpose, square and inverse of a 2 or 3 order matrix	Lab9_1.c
25.	Sorting of strings	Lab9_2.c
26.	Checking the validity of a date being entered	Lab9_3.c
27.	Finding m&c in y=mx+c by entering several values of x and y	Lab9_4.c
28.	Use of pointers to change the value of a integer	Lab10_2.c
29.	Swapping of two float numbers using a function	Lab10_3.c
30.	Printing the mirror image of a 2D array	Lab10_9.c
31.	mc_cluskey method	mc_cluskey.c
32.	Use of memmove	Memmove.c
33.	Multiple function call using pointers	multiple_function_call.c, ptrfunc.c
34.	My version of strcopy	Mystrcopy.c
35.	Printing n^m without using maths library	nm.c

36.	Type casting example from float to double	Parsing.c
37.	Pascal triangle	Pascal.c
38.	Pascal triangle using recursion	Pascal_rec.c
39.	Pattern of number entered by user forming inverted triangle	Pattern.c
40.	Game: predition of your day	Prediction.c
41.	Printing the array with the use of pointers	Ptr1.c
42.	Copying two strings into two empty strings using ponters	Ptr2.c
43.	Race between two players	Race.c
44.	Counting which word appeared how many times in a paragraph	Shakespeare.c
45.	Sorting using pointers	Sort.c
46.	Use of atof, strtod, and strtol	Str1.c
47.	Use of strstr	Str2.c
48.	Sorting of a name and d.o.b structure wrt name	String.c
49.	Use of strtok	Strtok.c
50.	Print the three arrays in a structure	Structure.c
51.	Printing all possible subsets in a A*A relation	Subsets.c
52.	Sum of two numbers	Sum2.c
53.	Swapping using 3 variables	Swap.c
54.	Swapping using 2 variables	Swap2.c
55.	Checking number of students with same grades by entering grades of all students(EOF introduced)	Switch.c
56.	Print the topological order of a partial order	T_Order.c
57.	Printing the triangle of stars	Triangle.c
58.	Use of ++ operator	use_of_++.c
59.	Printing the xnacci series	Xnacci.c
60.	Printing the text with proper alignment	Margin.c
61.	Print the entered amount of money in words for a cheque	Word.c
62.	Printing all the possibilities of cards from the deck of 52 cards. Use of **typedef**, **structres passing to functions** and defining **bit widths** is enumerated.	Bitwise.c
63.	Use of enumerations	Enum.c
64.	File handling first program, opening a file for writing	Fh1.c
65.	How to read from a sequential file	Fh2.c
66.	Creating a random access file and storing a blank structure into it with 100 spaces for storing data in that file	Fh3.c
67.	Writing data to the random access file	Fh4.c
68.	Reading data from a random access file	Fh5.c
69.	Find the accounts with positive balance from a file named "practice.odt" where the information about some account numbers,	Credit.c

	names and balance was stored in fh2.c program	
70.	Maintain the record of transactions in a bank	Bank.c
71.	Maintain the record of Names and age of 100 people with different serial numbers in nameage.dat file.	Nameage.c
72.	Linked list creation and storing characters in the linked list.	Nodes.c
73.	Stack creation and storing integer values in a stack. LIFO	Stack.c
74.	Creation and working on queues.	Queue.c
75.	To input a tree and print all its transversals i.e. Inorder, Preorder and Postorder transversals.	Tree.c
76.	Enter two linked lists and the apply certain functions like delete and merge on those lists entered.	Linkmerge.c
77.	Create a link list of numbers, reverse that list and the store the reversed list in other list and print the reversed list.	Linkrev.c
78.	Print the postfix expression of the infix expression	Postfix.c
79.	Evaluate a postfix expression	Postevaluate.c
80.	Put the entered sentence in a tree with words from the string put in the tree and then printing all its possible transversals.	Tokentree.c
81.	Determine height of a tree.	Height.c
82.	Print a tree	Treeprint.c
83.	Define my own header file which contains a function area	area_header.c
84.	Use my own header file "area_header.c" in the current file to find area.	use_area_header.c
85.	Using fuctions of library <stdarg> which include va_list, va_start, va_arg and va_end.	Variable.c
86.	Use of <stdlib> functions, exit() and atexit()	Exit.c
87.	Use of functions from <signal.h> library	Signal.c
88.	Use of calloc function of <stdlib.h> library	Alloc.c
89.	Merge sort of an array	Merge.c
90.	Breadth first search	Bfs.c
91.	Find a^b using exponential by squaring. Complexiety is log (base 2)n	Exponent.c