1. Write a program to print the sum of two numbers entered by user by defining your own function.

2. Define a function that returns the product of two numbers entered by user.

3. Write a program to print the circumference and area of a circle of radius entered by user by defining your own function.

4. Define two functions to print the maximum and the minimum number respectively among three numbers entered by user.

5. Define a program to find out whether a given number is even or odd.

6. A person is elligible to vote if his/her age is greater than or equal to 18. Define a function to find out if he/she is elligible to vote.

7. Define a function to find out if number is prime or not.

8. Write a program which will ask the user to enter his/her marks (out of 100). Define a function that will display grades according to the marks entered as below:

Marks	Grade
91-100	AA
81-90	AB
71-80	BB
61-70	BC
51-60	CD
41-50	DD
<=40	Fail

9. Write a program to print the factorial of a number by defining a function named 'Factorial'.

Factorial of any number n is represented by n! and is equal to 1*2*3*....*(n-1)*n. E.g.-

4! = 1*2*3*4 = 24

3! = 3*2*1 = 6

2! = 2*1 = 2

Also,

1! = 1

0! = 0

10. Print the multiplication table of 15 using recursion.

11. Define a function to print the prime factors of a number.

12. Using recursion, define a function to know nth term of a Fibonacci series.

Nth term of Fibonacci series is

F(n) = F(n-1)+F(n-2)

F(0) = 0

F(1) = 1

F(2) = F(1)+F(0) = 1+0 = 1

F(3) = F(2)+F(1) = 1+1 = 2

F(4) = F(3)+F(2) = 2+1 = 3

13. Define a function named 'perfect' that determines if parameter number is a perfect number. Use this function in a program that determines and prints all the perfect numbers between 1 and 1000.

[An integer number is said to be "perfect number" if its factors, including 1(but not the number itself), sum to the number. E.g., 6 is a perfect number because 6=1+2+3].

14. Define a function to calculate power of a number raised to other i.e. ab using recursion where the numbers 'a' and 'b' are to be entered by the user

15. Write a program that takes as input your gross salary and your total saving and uses another function named taxCalculator() to calculate your tax. The taxCalculator() function takes as parameters the gross salary as well as the total savings amount. The tax is calculated as follows:

(a) The savings is deducted from the gross income to calculate the taxable income. Maximum deduction of savings can be Rs. 100,000, even though the amount can be more than this.

(b) For up to 100,000 as taxable income the tax is 0 (Slab 0); beyond 100,000 to 200,000 tax is 10% of the difference above 100,000 (Slab 1); beyond 200,000 up to 500,000 the net tax is the tax calculated from Slab 0 and Slab 1 and then 20% of the taxable income exceeding 200,000 (Slab 2); if its more than 500,000, then the tax is tax from Slab 0, Slab 1, Slab 2 and 30% of the amount exceeding 500,000.

16. Write a function that takes your date of birth in YYYY, MM and DD format (separated by spaces) as input as well as the current date, in same format, and calculates your age in years, months and days. You must check for leap years also. Write a separate function to check for leap year.

17. Write a program to print the address of a variable whose value is input from user.

18. Write a program to print the address of the pointer to a variable whose value is input from user.

19. Write a program to print the value of the address of the pointer to a variable whose value is input from user.

20. Write a program to print a number which is entered from keyboard using pointer.

21. Write a function which will take pointer and display the number on screen. Take number from user and print it on screen using that function.

22. Write a program to find out the greatest and the smallest among three numbers using pointers.

23. Write a program to find the factorial of a number using pointers.

24. Write a program to reverse the digits a number using pointers.

25. Take 10 integer inputs from user and store them in an array and print them on screen.

26. Take 10 integer inputs from user and store them in an array. Again ask user to give a number. Now, tell user whether that number is present in array or not.

27. Take 20 integer inputs from user and print the following:

    number of positive numbers

    number of negative numbers

    number of odd numbers

    number of even numbers

    number of 0.

28. Take 10 integer inputs from user and store them in an array. Now, copy all the elements in another array but in reverse order.

29. Write a program to find the sum and product of all elements of an array.

30. Initialize and print all elements of a 2D array.

31. Find the largest and smallest elements of an array.

32. Write a program to check if elements of an array are same or not it read from front or back. E.g.-

    2 3 15 15 3 2

33. Take an array of 10 elements. Split it into middle and store the elements in two dfferent arrays. E.g.-

INITIAL array :

58 24 13 15 63 9 8 81 1 78

After spliting :

58 24 13 15 63

9 8 81 1 78

34. Consider an integer array, the number of elements in which is determined by the user. The elements are also taken as input from the user. Write a program to find those pair of elements that has the maximum and minimum difference among all element pairs.

35. If the input array is [10, 12, 20, 30, 25, 40, 32, 31, 35, 50, 60], your program should be able to find that the subarray lies between the indexes 3 and 8.

36. Write a program to print sum, average of all numbers, smallest and largest element of an array.

37. Take an array of length n where all the numbers are nonnegative and unique. Find the element in the array possessing the highest value. Split the element into two parts where first part contains the next highest value in the array and second part hold the required additive entity to get the highest value. Print the array where the highest value get splitted into those two parts.

Sample input: 4 8 6 3 2

Sample output: 4 6 2 6 3 2

38. Write a program to shift every element of an array to circularly right. E.g.-

INPUT : 1 2 3 4 5

OUTPUT : 5 1 2 3 4

39. Initialize a 2D array of 3*3 matrix. E.g.-

1 2 3

4 5 6

7 8 9

Check if the matrix is symmetric or not.

40. Input any number. Find the sum of the digits of the number using a recursive function.

41. Pass a 2D array to function and access all its elements.

42. Write a program to add and multiply two 3x3 matrices.

43. Write a program to reverse a string with and without using any predefined function.

44. Write a program to find the length of a string without using predefined functions.

45. Check the occurrence of the letter 'e' and the word 'is' in the sentence "This is umbrella" without using predefined functions.

46. Write a program to find the number of vowels, consonents, digits and white space characters in a string

47. Write a program to concatenate two strings "Hello" and "World".

48. Write a program to find out the largest and smallest word in the string "This is an umbrella".

49. Write a program to check if a given string is a Palindrome.

A palindrome reads same from front and back e.g.- aba, ccaacc, mom, etc.

50. Write down the names of 10 of your friends in an array and then sort those in alphabetically ascending order.

51. Write a program to delete all the consonents from the string "Hello, have a good day".

52. Write a program to delete the word "the" in the sentence "This is the lion in the cage".

53. Write a program to check if the two strings entered by user are anagrams or not. Two words are said to be anagrams if the letters of one word can be rearranged to form the other word. For example, jaxa and ajax are anagrams of each other.

54. Input a string which contains some palindrome substrings. Find out the position of palindrome substrings if exist and replace it by *. (For example if input string is “bob has a radar plane” then it should convert in “*** has a ***** plane”.

55. Write a program to replace a given substring in a sentence with another string. For example, in the sentence, ” A batman with bat” if we replace ”bat” with ”snow”, the new sentence should be printed as ”A snowman with snow”.

56. Write a program to reverse individual words in a string, where each word may be delimited by a dot, comma, space or tab, like www.google.com should become www.elgoog.moc.

57. Write a macro to calculate area and perimeter of a rectangle.

58. Write a macro to compare two numbers.

59. Write a macro to find average of two numbers.

60. Write a macro to find absolute value of number.

61. Write a macro to calculate simple interest from principal, rate of interest and time.

Simple interest = (principal*rate of interest*time)/100.

62. Write a program to store and print the roll no., name , age and marks of a student using structures.

63. Write a program to store the roll no. (starting from 1), name and age of 5 students and then print the details of the student with roll no. 2.

64. Write a program to store and print the roll no., name, age, address and marks of 15 students using structure.

65. Write a program to add two distances in inch-feet using structure. The values of the distances is to be taken from the user.

66. Enter the marks of 5 students in Chemistry, Mathematics and Physics (each out of 100) using a structure named Marks having elements roll no., name, chem_marks, maths_marks and phy_marks and then display the percentage of each student.

67. Write a program to add, subtract and multiply two complex numbers using structures to function.

68. Write a structure to store the roll no., name, age (between 11 to 14) and address of students (more than 10). Store the information of the students.

1 - Write a function to print the names of all the students having age 14.

2 - Write another function to print the names of all the students having even roll no.

3 - Write another function to display the details of the student whose roll no is given (i.e. roll no. entered by the user).

69. Write a structure to store the name, account number and balance of customers (more than 10) and store their information.

1 - Write a function to print the names of all the customers having balance less than $200.

2 - Write a function to add $100 in the balance of all the customers having more than $1000 in their balance and then print the incremented value of their balance.

70. Write a program to compare two dates entered by user. Make a structure named Date to store the elements day, month and year to store the dates. If the dates are equal, display "Dates are equal" otherwise display "Dates are not equal".

71. Write a structure to store the names, salary and hours of work per day of 10 employees in a company. Write a program to increase the salary depending on the number of hours of work per day as follows and then print the name of all the employees along with their final salaries.

Hours of work per day	8	10	>=12
Increase in salary	$50	$100	$150

72. Let us work on the menu of a library. Create a structure containing book information like accession number, name of author, book title and flag to know whether book is issued or not.

Create a menu in which the following can be done.

1 - Display book information

2 - Add a new book

3 - Display all the books in the library of a particular author

4 - Display the number of books of a particular title

5 - Display the total number of books in the library

6 - Issue a book

(If we issue a book, then its number gets decreased by 1 and if we add a book, its number gets increased by 1)

73.Create a structure named Date having day, month and year as its elements. Store the current date in the structure. Now add 45 days to the current date and display the final date.

74. Write a program by creating an 'Employee' class having the following functions and print the final salary.

1 - 'getInfo()' which takes the salary, number of hours of work per day of employee as parameters

2 - 'AddSal()' which adds $10 to the salary of the employee if it is less than $500.

3 - 'AddWork()' which adds $5 to the salary of the employee if the number of hours of work per day is more than 6 hours.

75. Create a class called 'Matrix' containing constructor that initializes the number of rows and the number of columns of a new Matrix object. The Matrix class has the following information:

1 - number of rows of matrix

2 - number of columns of matrix

3 - elements of matrix (You can use 2D vector)

The Matrix class has functions for each of the following:

1 - get the number of rows

2 - get the number of columns

3 - set the elements of the matrix at a given position (i,j)

4 - adding two matrices.

5 - multiplying the two matrices

You can assume that the dimensions are correct for the multiplication and addition.

76. Create a class with a function that prints "This is parent class" and its subclass with another function that prints "This is child class". Now, create an object for each class and call

1 - function of the parent class by the object of the parent class

2 - function of the child class by the object of the child class

3 - function of the parent class by the object of the child class

77. Create a class named 'Member' having the following members:

Data members

1 - Name

2 - Age

3 - Phone number

4 - Address

It also has a function named 'printSalary' which prints the salary of the members.

Two classes 'Employee' and 'Manager' inherits the 'Member' class. The 'Employee' and 'Manager' classes have data members 'specialization' and 'department' respectively. Now, assign name, age, phone number, address and salary to an employee and a manager by making an object of both of these classes and print the same.

78. Create a class named 'Rectangle' with two data members 'length' and 'breadth' and two functions to print the area and perimeter of the rectangle respectively. Its constructor having parameters for length and breadth is used to initialize the length and breadth of the rectangle. Let class 'Square' inherit the 'Rectangle' class with its constructor having a parameter for its side (suppose s) calling the constructor of its parent class. Print the area and perimeter of a rectangle and a square.

79. Now repeat the above example to print the area of 10 squares.

Hint-Use array of objects

80. Create a class named 'Shape' with a function to print "This is a shape". Then create two other classes named 'Rectangle' and 'Circle' inheriting the Shape class, both having a function to print "This is rectangular shape" and "This is circular shape" respectively. Create a subclass 'Square' of 'Rectangle' having a function to print "Square is a rectangle". Now call the function of the 'Shape' and the 'Rectangle' class by the object of the 'Square' class.

81. Write a program to print the names of students by creating a Student class. If no name is passed while creating an object of the Student class, then the name should be "Unknown", otherwise the name should be equal to the String value passed while creating the object of the Student class.

82. Create a class named 'Rectangle' with two data members- length and breadth and a function to calculate the area which is 'length*breadth'. The class has three constructors which are :

1 - having no parameter - values of both length and breadth are assigned zero.

2 - having two numbers as parameters - the two numbers are assigned as length and breadth respectively.

3 - having one number as parameter - both length and breadth are assigned that number.

Now, create objects of the 'Rectangle' class having none, one and two parameters and print their areas.

83. Suppose you have a Piggie Bank with an initial amount of $50 and you have to add some more amount to it. Create a class 'AddAmount' with a data member named 'amount' with an initial value of $50. Now make two constructors of this class as follows:

1 - without any parameter - no amount will be added to the Piggie Bank

2 - having a parameter which is the amount that will be added to the Piggie Bank

Create an object of the 'AddAmount' class and display the final amount in the Piggie Bank.

84. Create a class named 'Programming'. While creating an object of the class, if nothing is passed to it, then the message "I love programming languages" should be printed. If some String is passed to it, then in place of "programming languages" the name of that String variable should be printed.

For example, while creating the object if we pass "cpp", then "I love cpp" should be printed.

85. Create a class named 'PrintNumber' to print various numbers of different datatypes by creating different functions with the same name 'printn' having a parameter for each datatype.

86. Create a class to print an integer and a character using two functions having the same name but different sequence of the integer and the character parameters.

For example, if the parameters of the first function are of the form (int n, char c), then that of the second function will be of the form (char c, int n).

87. Create a class to print the area of a square and a rectangle. The class has two functions with the same name but different number of parameters. The function for printing the area of rectangle has two parameters which are its length and breadth respectively while the other function for printing the area of square has one parameter which is the side of the square.

88. Create a class 'Student' with three data members which are name, age and address. The constructor of the class assigns default values to name as "unknown", age as '0' and address as "not available". It has two functions with the same name 'setInfo'. First function has two parameters for name and age and assigns the same whereas the second function takes has three parameters which are assigned to name, age and address respectively. Print the name, age and address of 10 students.

Hint - Use array of objects

89. Create a class 'Degree' having a function 'getDegree' that prints "I got a degree". It has two subclasses namely 'Undergraduate' and 'Postgraduate' each having a function with the same name that prints "I am an Undergraduate" and "I am a Postgraduate" respectively. Call the function by creating an object of each of the three classes.

90. A boy has his money deposited $1000, $1500 and $2000 in banks-Bank A, Bank B and Bank C respectively. We have to print the money deposited by him in a particular bank.

Create a class 'Bank' with a function 'getBalance' which returns 0. Make its three subclasses named 'BankA', 'BankB' and 'BankC' with a function with the same name 'getBalance' which returns the amount deposited in that particular bank. Call the function 'getBalance' by the object of each of the three banks.

91. A class has an integer data member 'i' and a function named 'printNum' to print the value of 'i'. Its subclass also has an integer data member 'j' and a function named 'printNum' to print the value of 'j'. Make an object of the subclass and use it to assign a value to 'i' and to 'j'. Now call the function 'printNum' by this object.

92. All the banks operating in India are controlled by RBI. RBI has set a well defined guideline (e.g. minimum interest rate, minimum balance allowed, maximum withdrawal limit etc) which all banks must follow. For example, suppose RBI has set minimum interest rate applicable to a saving bank account to be 4% annually; however, banks are free to use 4% interest rate or to set any rates above it.

Write a program to implement bank functionality in the above scenario. Note: Create few classes namely Customer, Account, RBI (Base Class) and few derived classes (SBI, ICICI, PNB etc). Assume and implement required member variables and functions in each class.

93. Write a program to print the name, salary and date of joining of 10 employees in a company. Use array of objects.

94. Write a program to print the roll number and average marks of 8 students in three subjects (each out of 100). The marks are entered by the user and the roll numbers are automatically assigned.

95. Write a program to calculate the average height of all the students of a class. The number of students and their heights are entered by the user.

96. Lets create a bank account. Create a class named 'BankAccount' with the following data members

1 - Name of depositor

2 - Address of depositor

3 - Type of account

4 - Balance in account

5 - Number of transactions

Class 'BankAccount' has a function for each of the following

1 - Generate a unique account number for each depositor

For the first depositor, account number will be BA1000, for the second depositor it will be BA1001 and so on

2 - Display information and balance of depositor

3 - Deposit more amount in the balance of any depositor

4 - Withdraw some amount from the balance deposited

5 - Change the address of depositor

After creating the class, do the following operations

1 - Enter the information (name, address, type of account, balance) of the depositors. Number of depositors are to be entered by the user.

2 - Print the information of any depositor.

3 - Add some amount to the account of any depositor and then display the final information of that depositor

4 - Remove some amount from the account of any depositor and then display the final information of that depositor

5 - Change the address of any depositor and then display the final information of that depositor

6 - Randomly repeat these processes for some other bank accounts and after that print the total number of transactions.

97. Write a program to create a directory that contains the following information.

(a) Name of a person

(b) Address

(c) Telephone Number (if available with STD code)

(d) Mobile Number (if available)

(e) Head of the family

98. Create two classes named Mammals and MarineAnimals. Create another class named BlueWhale which inherits both the above classes. Now, create a function in each of these classes which prints "I am mammal", "I am a marine animal" and "I belong to both the categories: Mammals as well as Marine Animals" respectively. Now, create an object for each of the above class and try calling

1 - function of Mammals by the object of Mammal

2 - function of MarineAnimal by the object of MarineAnimal

3 - function of BlueWhale by the object of BlueWhale

4 - function of each of its parent by the object of BlueWhale

99. Make a class named Fruit with a data member to calculate the number of fruits in a basket. Create two other class named Apples and Mangoes to calculate the number of apples and mangoes in the basket. Print the number of fruits of each type and the total number of fruits in the basket.

100. We want to calculate the total marks of each student of a class in Physics,Chemistry and Mathematics and the average marks of the class. The number of students in the class are entered by the user. Create a class named Marks with data members for roll number, name and marks. Create three other classes inheriting the Marks class, namely Physics, Chemistry and Mathematics, which are used to define marks in individual subject of each student. Roll number of each student will be generated automatically.

101. We want to store the information of different vehicles. Create a class named Vehicle with two data member named mileage and price. Create its two subclasses

*Car with data members to store ownership cost, warranty (by years), seating capacity and fuel type (diesel or petrol).

*Bike with data members to store the number of cylinders, number of gears, cooling type(air, liquid or oil), wheel type(alloys or spokes) and fuel tank size(in inches)

Make another two subclasses Audi and Ford of Car, each having a data member to store the model type. Next, make two subclasses Bajaj and TVS, each having a data member to store the make-type.

Now, store and print the information of an Audi and a Ford car (i.e. model type, ownership cost, warranty, seating capacity, fuel type, mileage and price.) Do the same for a Bajaj and a TVS bike.

102. Create a class named Shape with a function that prints "This is a shape". Create another class named Polygon inheriting the Shape class with the same function that prints "Polygon is a shape". Create two other classes named Rectangle and Triangle having the same function which prints "Rectangle is a polygon" and "Triangle is a polygon" respectively. Again, make another class named Square having the same function which prints "Square is a rectangle".

Now, try calling the function by the object of each of these classes.

103. All the banks operating in India are controlled by RBI. RBI has set a well defined guideline (e.g. minimum interest rate, minimum balance allowed, maximum withdrawal limit etc) which all banks must follow. For example, suppose RBI has set minimum interest rate applicable to a saving bank account to be 4% annually; however, banks are free to use 4% interest rate or to set any rates above it.Write a program to implement bank functionality in the above scenario. Note: Create few classes namely Customer, Account, RBI (Base Class) and few derived classes (SBI, ICICI, PNB etc). Assume and implement required member variables and functions in each class.