Practice Question Bank (PCS-251)

STRINGS	
1.	Develop a C program to copy a string to another without using the inbuilt strcpy() function.
2.	Develop a C program to Concatenate Two Strings and display the concatenated string to the output screen without using predefined string function.
3.	Develop a C program to compare two strings using a function and return 1 if the strings are equal and 0 otherwise without using strcmp( ) function. Display the appropriate message to the output screen.
4.	Develop a C program to find the number of occurrences of each alphabet in a string accepted from the user and display the same to the output screen. Assume that the string contains only alphabets.
5.	Develop a program to calculate length of a string with and without inbuilt function.
6.	Develop a program to count the number of words in a string
7.	Develop a program to accept a sentence and count the characters without space.
8.	Write a C program to remove both leading and trailing white space characters in a string.
9.	Develop a program to capitalize each alternate word in a string.
10.	Write a C program to remove extra whitespaces or blanks from a string.
11.	Develop a program to reverse a given string without using any predefined function.
12.	Develop a program to find the number of occurrences of each alphabet in a given string and display the same to the output screen. Assume that the string contain only alphabets.
13.	Develop a program to extract a portion of a string and print the extracted string. Assume that 'm' characters are extracted, starting with the 'n'th character.
14.	Develop a program to check whether a given name is present in the array or not

15.	Develop a program to read names in a 2-D character array and sort them in ascending order.
16.	Develop a program to check whether a given string is Anagram or not
17.	Write a user-defined function to remove all repeated characters in a string.
18.	Design a C program that accepts a sentence from the user and extracts the vowels and stores by capitalizing each vowel into another string. Display the extracted and capitalized vowel string in the calling program.
19.	Write a C program that accepts a main string along with a starting position and ending position of a substring to be extracted from the user in the main program. Print the extracted string in the calling program.
20.	Write a C program to accept a string from the user and replace all the occurrences of punctuation marks with a # character. Display the modified string to the console. Implement using a UDF.
21.	Write a C program to accept a string from the user and display the frequency of punctuation marks present in it.
22.	Write a C program to accept a string from the user. Implement a UDF to display the frequency of vowels and spaces present in it to the console. Ignore the case.
23.	Write a user defined function that returns a string by deleting ‘m’ characters from a position ‘n’ of a string to the calling program. Accept the string and the number of characters and position of the character to be deleted from the string in the main program.
24.	Write a program in C to read a string from the keyboard and sort it using bubble sort.
25.	Write a program in C to find the number of times a given word 'the' appears in the given string.
26.	Write a program in C to find the largest and smallest words in a string.
27.	Write a program in C to split strings by space into words.
28.	Write a C program to find the length of the longest substring of a given string without repeating characters.
29.	Write a program in C to count the total number of alphabets, digits and special characters in a string.

Pointers	
1.	Develop a program to initialize a variable 'a' and create a pointer ‘ptr’ pointing to that variable. Perform the following tasks: i. Print the address of a ii. Print the address of ptr iii. Print the value in a iv. Print the value in ptr v. Print the value pointed by ptr
2.	Write a program in C to demonstrate the use of the & (address of) and *(value at address) operators.
3.	Demonstrate a program to access variables through pointers and show pointer arithmetic on int and char data types. The operations such as: a) Increment/Decrement of a Pointer. b) Addition/Subtraction of integer to a pointer
4.	Develop a program to find the sum and product of two pointer variables ‘Ptr1’ and 'ptr2' pointing to value 'a' and value 'b'.
5.	Write a program in C to find the maximum number between two numbers using a pointer.
6.	Develop a program to check the size of integer, character and a float pointer. print The following: i. Print the size of original data ii. Print the size of data pointed by pointers iii. Print the size of pointer variables
7.	Develop a program to swap two numbers using pointers. Display the swapped numbers in the calling function using call by value and call by reference.
8.	Develop a program using pointers to read an array of integers and print its elements.
9.	Develop a program to find second largest element in an array using pointers.
10.	Develop a program using pointers to read an array of integers and print its elements in the reverse order.
11.	Develop a c program to sort n elements of an array using pointers(use method of insertion sort). Display the sorted array on the output screen using below functions: Sort(): to sort the array Display(): to display the unsorted and sorted array

	Main(): calling function
12.	Write a C program to accept & store N real numbers into an array. Find their mean and standard deviation using pointers and display the same to the output screen. Sample: Input: N=9 Array[ ] = {22, -3, 11, -2, 5, 7,12,13, 18} Output: Mean μ : 9.22 Std. Deviation: 7.91 Hint: Use Std. Deviation formula: $\sigma = \sqrt{\frac{\sum(x_i - \mu)^2}{N}}$ N = the size of the population xi = each value from the population μ = the population mean
13.	Write a C program to accept N numbers from the user into an array then multiply all integers by 2 and then divide by 3. Implement using a pointer to perform these operations and then subsequently display the modified array to the screen.
14.	Write a C Program to Swap Numbers in Cyclic Order Using Call by Reference. Sample: Input: Enter a, b and c respectively: 1 2 3 Output: Value before swapping: a = 1 b = 2 c = 3 Value after swapping: a = 3 b = 1 c = 2
15.	Write a C program using functions to find the frequency Count of array elements. The called function should display the count on the output screen. Use Pointers for the implementation.
16.	Accept the set of N observations from the user. Write a user defined function to find the median and range of the set of observations. Display the median and range in the calling program. Decide upon the appropriate data types for storage and calling mechanism.
17.	Develop a program to find the sum and product of two integers(n1,n2) using a UDF where the input n1, n2 to be passed as call by value along with two pointers and calculate sum (in the first pointer) and product(in the second pointer) by call by reference in the same UDF. Hint: void compute(int,int,int*,int*);
18.	Design a UDF that returns 1 if the string is a palindrome otherwise zero. Accept a string from user in the calling program. Display an appropriate message in the calling program. Implement a C program for the same. Make use of a pointer to determine the string is palindrome or not. Ignore the case.
19.	Develop a C program to reverse and display a string to the screen using a pointer.

20.	Write a C program to accept N integers from the user into an array. If any integer is positive then subtract 2 from it, if integer is negative add 1 to it and if zero then do nothing. Implement using a pointer to perform these operations and finally display the modified array to the screen.
21.	Accept a string from the user. Using a pointer display only the first alphabets of each of the words in uppercase present in it to the screen separated by a single space. Implement a C program for the same.
22.	Develop a C program to find the concatenation of two strings without using in-built function strcat() and return the length of the concatenated string. Make use of UDF and pointers.
23.	WAP in C to demonstrate the returning of a pointer from a function.
24.	WAP in C to accept an array from the user and an integer k. Implement a UDF to check whether k is present in the array. If k is found in array, then return a pointer to the location where it is found else return 0. Display the appropriate message in the calling function.
25.	WAP in C to accept a string from the user. Implement a user defined function that will return a pointer to the character present in the string which has largest ASCII value. Display the resultant character value in the calling program. Assume the string contains only printable characters.
	Structures
26.	Develop a program to Store Information of Students Using Structure.
27.	Develop a program to initialize a structure student with members: roll number, age, gender, marks (assume marks as float data type). Calculate the size of structure.
28.	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.
29.	Develop a program in C to read a structure in the main program of an Employee that contains Name, EmpCode and Salary as the members. Write a function display the details of the employee in the following format.

	ABC Corporation **Name : Surjit Singh** **EmpCode : E111** **DOB : 14th Feb 1995** **Nett. Salary : Rs. 95,000/-**
30.	Write a structure to store the name, account number and balance of customers (more than 10) and store their information. i. Write a function to print the names of all the customers having balance less than $200. ii. 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.
31.	Develop a C program to read and display Name, Roll Number, Date of Birth and Date of Admission details of a student from the keyboard where the date of birth and date of admission consists of further three members such as day, month and year as a separate structure. Implement using a C structure.
32.	Write a structure to store the roll no., name, age (between 12 to 15) and address of students (more than 10). Store the information of the students. i. Write a function to print the names of all the students having age 15. ii. Write another function to print the names of all the students having even roll no. iii. Write another function to display the details of the student whose roll no is given (i.e. roll no. entered by the user).
33.	Design a structure that stores the performance of a test conducted for a class of N students. The members of the structure are Roll No, Name of the student, Sem No, Name of the Section (A1,A2,….T1,T2) and Percentage Score. Implement a C program to store and display the Roll No, Name and Percentage scores to the screen.
34.	Develop a program to read attributes of an item from the user such as ItemCode, ItemName, Quantity and Rate. Implement a C program using a structure to find the total cost of the inventory of storing N items in the stock.
35.	Develop a program to find total and average sales of 'N' employees (empcode, emp name, emp sale) using array of structures.

36.	Develop a program to read a set of name, roll number, date of birth and date of admission of 'n' students in the college where date of birth and date of admission consists of three members such as day,month, and year as a separate structure.