

Computer Science (083) Practical File for Class XII

Index for C++ Programming

Name: _____

Section: _____

Roll No.: _____

| SN. | Program Description | Date | Sign. | | | | | | | | | | | | | | |
|----------------------|--|----------------------|---------------|-------------|------------------|-----------|-------------|-----------|------------|-----------|------------|-----------|----------|-----------|-------|--|--|
| 1. | Write a program to input two numbers m and n and display first m multiples of n. | | | | | | | | | | | | | | | | |
| 2. | Write a program to input day number of a week and display the corresponding day name. | | | | | | | | | | | | | | | | |
| 3. | Write a menu driven program to calculate the TSA and volume of a cube, cuboid, or cylinder depending upon user's choice. | | | | | | | | | | | | | | | | |
| 4. | Write a program to read a string and print out the following: 1) No. of capital alphabets, 2) No. of small alphabets, 3) No. of non-alphabets | | | | | | | | | | | | | | | | |
| 5. | Write a program to read a string and print it after replacing each of its capital alphabets by the corresponding small alphabet and each small alphabet by its corresponding capital alphabet. | | | | | | | | | | | | | | | | |
| 6. | Write a program to input 10 elements in an array and then display these elements. | | | | | | | | | | | | | | | | |
| 7. | Write a program to input 10 elements in an array and then display these elements in reverse order. | | | | | | | | | | | | | | | | |
| 8. | Write a program to input elements in a 2D array and then display this array in matrix form. | | | | | | | | | | | | | | | | |
| 9. | Write a program to input elements in a 2D array and then display the sum of main diagonal elements of this array. | | | | | | | | | | | | | | | | |
| 10. | Write a function to check whether a given string is palindrome or not. | | | | | | | | | | | | | | | | |
| 11. | <p>A class STUDENT has 3 data members: Name, Roll Number, Marks of 5 subjects, Stream and member functions to input and display data. It also has a function member to assign stream on the basis of the table given below:</p> <table style="margin-left: 40px;"> <thead> <tr> <th style="text-align: left;"><u>Average Marks</u></th> <th style="text-align: left;"><u>Stream</u></th> </tr> </thead> <tbody> <tr> <td>96% or more</td> <td>Computer Science</td> </tr> <tr> <td>91% - 95%</td> <td>Electronics</td> </tr> <tr> <td>86% - 90%</td> <td>Mechanical</td> </tr> <tr> <td>81% - 85%</td> <td>Electrical</td> </tr> <tr> <td>75% - 80%</td> <td>Chemical</td> </tr> <tr> <td>71% - 75%</td> <td>Civil</td> </tr> </tbody> </table> <p>Declare a structure STUDENT and define the member functions. Write a program to define a structure STUDENT and input the marks of n (≤ 20) students and for each student allot the stream. (Don't use any array).</p> | <u>Average Marks</u> | <u>Stream</u> | 96% or more | Computer Science | 91% - 95% | Electronics | 86% - 90% | Mechanical | 81% - 85% | Electrical | 75% - 80% | Chemical | 71% - 75% | Civil | | |
| <u>Average Marks</u> | <u>Stream</u> | | | | | | | | | | | | | | | | |
| 96% or more | Computer Science | | | | | | | | | | | | | | | | |
| 91% - 95% | Electronics | | | | | | | | | | | | | | | | |
| 86% - 90% | Mechanical | | | | | | | | | | | | | | | | |
| 81% - 85% | Electrical | | | | | | | | | | | | | | | | |
| 75% - 80% | Chemical | | | | | | | | | | | | | | | | |
| 71% - 75% | Civil | | | | | | | | | | | | | | | | |
| 12. | <p>Define a class student with the following specifications:</p> <p>Private members of the class: Admission Number - An Integer Name - string of 20 characters Class - Integer Roll Number - Integer</p> <p>Public members of the class: getdata() - To input the data showdata() - To display the data</p> <p>Write a program to define an array of 10 objects of this class, input the data in this array and then display this list.</p> | | | | | | | | | | | | | | | | |

| | | | |
|-----|--|--|--|
| 13. | Define a POINT class for two-dimensional points (x, y). Include a default constructor, a copy constructor, a negate() function to transform the point into its negative, a norm() function to return the point's distance from the origin (0,0), and a print() function besides the functions to input and display the coordinates of the point. Use this class in a menu driven program to perform various operations on a point. | | |
| 14. | Write a program to find the specific element from an array of 10 elements using binary search technique. | | |
| 15. | Create a class student with data members name, class, section, roll No. and function members getdata(), printdata(), and promote(). From this class derive a class 'Sr_std' with additional data member stream. Also include another function member change_stream(). Use these classes in a program. | | |
| 16. | Write a program to input the name of a text file from the user and display: <ul style="list-style-type: none"> a) The number of blanks present in the file. b) The number of lines present in the file. c) The number of capital alphabets present in the file. d) The number of small alphabets present in the file. e) The number of lines starting with a capital alphabet. f) The number of words present in the file. g) The number of digits present in the file. h) The number of words ending with a vowel | | |
| 17. | Write a program to input the name of a text file from the user. Then input a string and search for the string in the file. The program should an appropriate message if the file with the given name is not present. | | |
| 18. | Write a program to input a text file name, read the contents of the file and create a new file named COPY.TXT, which shall contain only those words from the original file which don't start with an uppercase vowel (i.e., with 'A', 'E', 'I', 'O', 'U'). For example, if the original file contains The First Step To Getting The Things You Want Out Of Life is This: Decide What You Want. - Ben Stein Then the text file COPY.TXT shall contain The First Step To Getting The Things You Want Life is This: Decide What You Want. - Ben Stein | | |
| 19. | Write an interactive C++ program to open a text file and then display the following: <ul style="list-style-type: none"> 1) Frequency table of all the alphabetic characters present in the file. 2) Number of numeric characters present in the file. | | |
| 20. | Create a text file (w/o using any C++ program) containing the names of students and their marks in the following format: <p style="text-align: center;">Ajay 350 Vijay 340</p> where name and marks are separated by either a space or a tab and end of line is a record separator. Write a program to read this file and display the records in two columns name and marks. Within the name column, the students' names are to be left justified and marks are to be right justified in the marks column. | | |

| | | | |
|-----|---|--|--|
| 21. | <p>Declare a structure <i>telerec</i> in C++, containing name (20 characters) and telephone number. Write a program to maintain a file of telephone records. The program should allow the following functions on the file:</p> <ol style="list-style-type: none"> 1) To append records in the file. 2) Display the name for a given telephone number. If the telephone number does not exist then display error message "record not found". 3) Display the telephone number(s) for a given name. If the name does not exist then display error message "record not found". | | |
| 22. | <p>A blood bank maintains a data file that contains the following information for every donor: Name, Date of Birth, Telephone number, Blood group. Write a program in C++ to do the following:</p> <ol style="list-style-type: none"> 1) Given a blood group, display name, date of birth and phone number of all the persons of the given blood group. 2) Append records in the file. 3) Input a telephone number and modify the corresponding record. | | |
| 23. | <p>Create two payroll files COMP1.DAT and COMP2.DAT. Each of the files should have the following fields:</p> <p style="margin-left: 40px;">EmpNo : Integer Name : A string of 20 characters Salary : A floating point number.</p> <p>Both the files should be created in the increasing order of the EmpNo. Your program should then merge the two files and obtain a third file NEWCOMP.DAT. The program should also display the data from all the three files.</p> <p>Do not use arrays for merging and sorting of the files. You can assume that the EmpNo are unique.</p> | | |
| 24. | <p>Write a menu driven program in C++ to perform the following functions on a binary file "BOOK.DAT" containing objects of the following class:</p> <pre>class Book { int BookNo; char Book_name[20]; public: // function to enter book details void enterdetails(); //function to display Book details void showdetails(); //function to return Book_no int Rbook_no() {return Book_no;} //function to return Book_name int Rbook_name() {return Book_name;} };</pre> <ol style="list-style-type: none"> 1. Append Records 2. Modify a record for a given book no. 3. Delete a record with a given book no. 4. Search for a record with a given Book name 5. Display a sorted list of records (sort on Book No.) 6. Display a sorted list of records (Sort on Book Name) | | |
| 25. | <p>Write a program that reads a list of numbers from the user and places them in an array of type float. Once the numbers are stored in the array, the program should find their average and print it along with the list of numbers. Use pointer notation wherever possible.</p> | | |
| 26. | <p>Write a program to create a linked list in which each node contains the roll number and marks for a student of a class. Then display a list of all those students whose marks are above 50.</p> | | |

| | | | |
|-----|--|--|--|
| 27. | <p>Write a program to do the following: Declare an integer variable and initialize it to any negative value (say -56). Declare five pointer variables: pointer to int, pointer to char, pointer to float, pointer to long, and pointer to unsigned int. Store the address of the integer variable in the pointer to int. Then store this pointer's value in each of the other four pointer variables. Then display these values by using dereference operator with each pointer variable. Increment each pointer variable and then again display these values by using dereference operator with each pointer variable. (Discuss the output in the class room with your teacher.)</p> | | |
| 28. | <p>Write a menu driven program which allows the user to perform the following operations on a one-dimensional array:</p> <p style="padding-left: 40px;">Insertion, deletion, searching, sorting (bubble, selection, insertion), display.</p> <p>The program should automatically perform binary search if the array is sorted and linear search otherwise.</p> | | |
| 29. | <p>Write a program to input integer data in two arrays. Sort one of the arrays in ascending order and the other in descending order. Then merge them into a third array so that the data in the third array is in ascending order. The program should then display the data from all the three arrays.</p> | | |
| 30. | <p>Write a function in C++ which accepts an integer array and its size as arguments/parameters and exchanges the values of first half side elements with the second half side elements of the array.</p> <p>Example: If an array of eight elements has initial contents as 2,4,1,6,7,9,23,10 The function should rearrange the array as 7,9,23,10,2,4,1,6</p> | | |
| 31. | <p>Write a function in C++ to find and display the sum of each row and each column of a two-dimensional array of type float. Use the array and its size as parameters with float as its return type.</p> | | |
| 32. | <p>Write a function in C++ which accepts an integer array and its size as arguments/parameters and assigns the elements into a two-dimensional array of integers in the following format: If the array is 1, 2, 3, 4, 5, 6 If the array is 1, 2, 3 The resultant 2-D array is: The resultant 2-D array is:</p> <pre> 1 2 3 4 5 6 1 2 3 1 2 3 4 5 0 1 2 0 1 2 3 4 0 0 1 0 0 1 2 3 0 0 0 1 2 0 0 0 0 1 0 0 0 0 0 </pre> | | |
| 33. | <p>Write a program to input two matrices, find their sum, difference, or product depending upon user's choice, and then display the resultant matrix along with the original matrices.</p> | | |
| 34. | <p>Write a menu driven program which allows the user to perform the following operations on a stack (Array implementation):</p> <ol style="list-style-type: none"> 1) Push 2) Pop 3) Display | | |
| 35. | <p>Write a menu driven program which allows the user to perform the following operations on a queue (Array implementation):</p> <ol style="list-style-type: none"> 1) Insert 2) Delete 3) Display | | |

Index for SQL

Name: _____

Section: _____

Roll No.: _____

Consider the tables given below and answer the questions that follow:

Table: **Employee**

| No | Name | Salary | Zone | Age | Grade | Dept |
|----|---------|--------|--------|-----|-------|------|
| 1 | Mukul | 30000 | West | 28 | A | 10 |
| 2 | Kritika | 35000 | Centre | 30 | A | 10 |
| 3 | Naveen | 32000 | West | 40 | | 20 |
| 4 | Uday | 38000 | North | 38 | C | 30 |
| 5 | Nupur | 32000 | East | 26 | | 20 |
| 6 | Moksh | 37000 | South | 28 | B | 10 |
| 7 | Shelly | 36000 | North | 26 | A | 30 |

Table: **Department**

| Dept | DName | MinSal | MaxSal | HOD |
|------|---------|--------|--------|-----|
| 10 | Sales | 25000 | 32000 | 1 |
| 20 | Finance | 30000 | 50000 | 5 |
| 30 | Admin | 25000 | 40000 | 7 |

Write SQL commands to:

Create Table

1. Create the table Employee.
2. Create the table Department.

Insert data in a table

3. Insert data in the table Employee
4. Insert data in the table Department.

Simple Select

5. Display the details of all the employees.
6. Display the Salary, Zone, and Grade of all the employees.
7. Display the records of all the employees along with their annual salaries. The Salary column of the table contains monthly salaries of the employees.
8. Display the records of all the employees along with their annual salaries. The Salary column of the table contains monthly salaries of the employees. The new column should be given the name "Annual Salary".

Conditional Select using Where Clause

9. Display the details of all the employees who are below 30 years of age.
10. Display the names of all the employees working in North zone.
11. Display the salaries of all the employees of department 10.

Using NULL

12. Display the details of all the employees whose Grade is NULL.
13. Display the details of all the employees whose Grade is not NULL.

Using DISTINCT Clause

14. Display the names of various zones from the table Employee. A zone name should appear only once.
15. Display the various department numbers from the table Employee. A department number should be displayed only once.

Using Logical Operators (NOT, AND, OR)

16. Display the details of all the employees of department 10 who are above 30 years of age.
17. Display the details of all the employees who are getting a salary of more than 35000 in the department 30.
18. Display the names and salaries of all the employees who are not working in department 20.

19. Display the names and salaries of all the employees who are working neither in West zone nor in Centre zone.
20. Display the names of all the employees who are working in department 20 or 30.
21. Display the details of all the employees whose salary is between 32000 and 38000.
22. Display the details of all the employees whose grade is between 'A' and 'C'.
23. Display the details of all the employees aged above 30 in West zone.

Using IN Operator

24. Display the names of all the employees who are working in department 20 or 30. (Using IN operator)
25. Display the names and salaries of all the employees who are working neither in West zone nor in Centre zone. (Using IN operator)

Using BETWEEN Operator

26. Display the details of all the employees whose salary is between 32000 and 38000. (Using BETWEEN operator)
27. Display the details of all the employees whose grade is between 'A' and 'C'. (Using BETWEEN operator)

Using LIKE Operator

28. Display the name, salary, and age of all the employees whose names start with 'M'.
29. Display the name, salary, and age of all the employees whose names end with 'a'.
30. Display the name, salary, and age of all the employees whose names contain 'a'
31. Display the name, salary, and age of all the employees whose names do not contain 'a'
32. Display the details of all the employees whose names contain 'a' as the second character.

Using Aggregate functions

33. Display the sum and average of the salaries of all the employees.
34. Display the highest and the lowest salaries being paid in department 10.
35. Display the number of employees working in department 10.

Using ORDER BY clause

36. Display the details of all the employees in the ascending order of their salaries.
37. Display the details of all the employees in the descending order of their names.
38. Display the details of all the employees in the ascending order of their grades and within grades in the descending order of their salaries.

Using GROUP BY clause

39. Display the total number of employees in each department.
40. Display the highest salary, lowest salary, and average salary of each zone.
41. Display the average age of employees in each department only for those departments in which average age is more than 30.

Using UPDATE, DELETE, ALTER TABLE

42. Put the grade B for all those whose grade is NULL.
43. Increase the salary of all the employees above 30 years of age by 10%.
44. Delete the records of all the employees whose grade is C and salary is below 30000.
45. Delete the records of all the employees of department 10 who are above 40 years of age.
46. Add another column HireDate of type Date in the Employee table.

JOIN of two tables

47. Display the details of all the employees who work in Sales department.
48. Display the Name and Department Name of all the employees.
49. Display the names of all the employees whose salary is out of the specified range for the corresponding department.
50. Display the name of the department and the name of the corresponding HOD for all the departments.