

C Lab Activities

- **Lab 1:** Overview, Introduction and Demonstrations
- **Lab 2:** Formatted and Unformatted Input/output in C
- **Lab 3:** Branching in Control Structure
- **Lab 4:** Looping in Control Structure
- **Lab 5:** Array in C
- **Lab 6:** String in C
- **Lab 7:** Pointers in C
- **Lab 8:** User-defined functions in C
- **Lab 9:** Structure in C
- **Lab 10:** File handling in C
- **Group project on C:** Maximum 4 students in a group at the end of the course

Note

- While writing lab reports, students should write the necessary theory, code and sample input/output of each program covered in the lab.*
- In every lab class, students must bring the Pre-Report {Cover page to Flowchart} and also the Final report of an earlier lab.*

Sample

Pre-Report/Initials {Cover page to Flowchart} and **Final Report** {Initials+ Source code + Output + Discussion & Conclusion}

- **Cover page** “available on photocopy pasal or you can make your own cover page”
- **Title:** The title of that particular lab, for example, the title for lab 2 will be “*Formatted and Unformatted Input/output in C*”.
- **Theory:** all the theoretical details related to that particular lab
- **Algorithm**
- **Flowchart**
- **Source code:** code or program in C programming language.
- **Output**
- **Discussion and Conclusion**

IDE/Compiler

CodeBlocks/ Turbo C

INPUT-OUTPUT

- Hint:**

Potential Energy (PE) = mgh , where $g=9.8 \text{ m/s}^2$

- [illegible]

CONTROL STRUCTURE

- Determine the net payable amount on a sale using **if-else** statements. The net payable amount consists of the sale price plus sales tax. The sales tax is decided as
 - 6% of the sale price for national items
 - 13% of the sale price for foreign items
- An electric power distribution company charges its domestic consumers as follows:

Consumption Units	Rate of Charge
0 – 50	Rs. 0.50 per unit
51 – 100	Rs. 100 plus Rs. 0.65 per unit excess of 100
101 – 200	Rs. 230 plus Rs. 0.80 per unit excess of 200
201 and above	Rs. 390 plus Rs. 1.00 per unit excess of 300

Now, WAP reads the customer number and power consumed and displays the amount to be paid by the consumer using the concept of an **else-if ladder**.

3. Determine the net payable amount (sale price + tax) using the concept of the **switch statement**. The sales tax is decided as:
 - 10% of the sale price for grocery items.
 - 15% of the sale price for cosmetics items.
 - 20% of the sale price for clothing items.
 - 24% of the sale price for liquor items

Also, apply a 2% discount on the total price if the total price is more than Rs. 6000

4. WAP to find the sum of all odd natural numbers that are divisible by P and not by Q in a given range using the concept of:
 - © do loop
 - © while loop
 - © for loop.
5. WAP to display the following pattern:

```
*
 * *
* * *
* * * *
* * * * *
```

6. WAP to calculate the sum and average of N numbers.
 - **Case 1:** If a negative number is entered, the loop terminates.
 - **Case 2:** Negative numbers are skipped from the calculation.

ARRAY

1. WAP in C to find the *sum* of principal and secondary diagonal elements of a 2D array and check whether the *sum* is an Armstrong number or not.
2. Write a program to read a 4*4 square matrix, find the minimum value of that matrix and assign it to variable *min*, and replace all the elements of the principal diagonal with the *min* if the element is odd otherwise replace all the elements of secondary elements with *min* and display the updated matrix.
3. Write a program to find the largest and smallest element of an array, display the numbers in ascending as well as descending order using a single function and display the result in the calling function.
4. WAP in C to concatenate two strings entered by the user without using the string handling function.
5. WAP in C to check whether a string entered by the user is palindrome or not without using the string handling function.

USER-DEFINED FUNCTION

1. WAP to find the sum of all odd natural numbers that are divisible by P and not by Q in a given range using the concept of:
 - © function with return type and arguments
 - © function without return type and with arguments

- © function without return type and no arguments
- © function with return type and no arguments
- 2. WAP in C to calculate the value of a^b using a recursive function but without using the **pow()** function.
- 3. WAP to check the n^{th} term of the Fibonacci series is even or odd using the concept of recursive function.

STRUCTURES

1. Write a program that reads name, roll, program and marks obtained in five subjects by students until the user enters 'e' and displays the student detail and total marks obtained by each student.
2. Write a program to read structure "college" having name, estDate and location whereas estDate is another structure having day, month and year as members. Display the records of 10 colleges.
3. Write a program to read the heights of two students and display the difference between their heights. Use feet and inches as members of a structure to define height.
4. Create an array of structures named Employee with name and salary as structure members and the array of structures is passed to a function which sorts in ascending order based on salary and display the sorted array from main.

POINTER

1. Write a program to calculate the sum and average of integer numbers between M and N using a pointer.
2. WAP to input and display the details of n number of students entered by the user by passing structure by reference to the function.
3. Write a program to calculate the sum and average of integer numbers between M and N using the concept of array and pointer.

FILE MANAGEMENT

1. WAP in C to copy the contents of one file into another
2. WAP in C to write strings entered by the user to the file until the user presses N or n and displays the contents of the file.
3. WAP in C to illustrate the concept of formatted file input. Your program should read/input the name, roll, marks and address of students within the file and display it in the console.
4. WAP in C to read name, roll, marks and address of n students from a file student.txt using fread()