Activity 1

Write a C program that prompts the user to enter a string. Check if the string is a palindrome (reads the same forwards and backwards). Print "Palindrome" if it is a palindrome, otherwise print "Not a palindrome".

Activity 2

Write a C program that prompts the user to enter a string. Count the number of vowels (a, e, i, o, u) in the string and print the result on the console.

Activity 3

Write a C program that prompts the user to enter a string. Count the number of occurrences of a specific character in the string and print the result on the console.

Activity 4

You have been tasked with developing an Employee Management System for a company. The system should allow the user to perform various operations, such as adding employees, displaying employee details, and calculating average salaries. To implement this system, you need to use structures in C programming.

Example Output:

Employee Management System

Menu:

1. Add an employee
2. Display employee details
3. Calculate average salary
4. Exit

Enter your choice: 1

Enter employee ID: 101
Enter employee name: John Smith
Enter employee salary: 5000

Employee added successfully!

Menu:

1. Add an employee
2. Display employee details
3. Calculate average salary
4. Exit

Enter your choice: 1

Enter employee ID: 102
Enter employee name: Jane Doe
Enter employee salary: 6000

Employee added successfully!

Menu:

1. Add an employee
2. Display employee details
3. Calculate average salary
4. Exit

Enter your choice: 2

Employee Details:
ID: 101
Name: John Smith
Salary: 5000

ID: 102
Name: Jane Doe
Salary: 6000

Menu:

1. Add an employee
2. Display employee details
3. Calculate average salary
4. Exit

Enter your choice: 3

Average salary: 5500

Menu:

1. Add an employee
2. Display employee details
3. Calculate average salary
4. Exit

Enter your choice: 4

Exiting the program...

Activity 5

You have been assigned the task of developing a Student Database Management system for a school. The system should allow the user to perform operations such as adding students, searching for a student, and displaying student details. To implement this system, you need to use structures in C programming.

Student Database Management

Menu:

1. Add a student
2. Search for a student
3. Display all students
4. Exit

Enter your choice: 1

Enter student ID: 101
Enter student name: John Smith
Enter student age: 15
Enter student grade: 10th

Enter marks for 5 subjects:
Subject 1: 90
Subject 2: 85
Subject 3: 92
Subject 4: 88
Subject 5: 91

Student added successfully!

Menu:

1. Add a student
2. Search for a student
3. Display all students
4. Exit

Enter your choice: 1

Enter student ID: 102
Enter student name: Jane Doe
Enter student age: 16
Enter student grade: 11th

Enter marks for 5 subjects:
Subject 1: 88

Subject 2: 92
Subject 3: 85
Subject 4: 90
Subject 5: 89

Student added successfully!

Menu:

1. Add a student
2. Search for a student
3. Display all students
4. Exit

Enter your choice: 2

Enter student ID to search: 101

Student Details:
ID: 101
Name: John Smith
Age: 15
Grade: 10th
Marks: 90 85 92 88 91

Menu:

1. Add a student
2. Search for a student
3. Display all students
4. Exit

Enter your choice: 3

Student Details:
ID: 101
Name: John Smith
Age: 15
Grade: 10th
Marks: 90 85 92 88 91

ID: 102
Name: Jane Doe
Age: 16
Grade: 11th

Marks: 88 92 85 90 89

Menu:

1. Add a student
2. Search for a student
3. Display all students
4. Exit

Enter your choice: 4

Exiting the program…