1. **Write a program to get input of two 3x3 matrices and find out the sum and product of the matrices and display the result of sum and product.**
2. **PROGRAM**

#include <stdio.h>

int main() {

    int matrix1[3][3], matrix2[3][3], sum[3][3], product[3][3];

    printf("Enter the elements of first 3x3 matrix: \n");

    for (int i = 0; i < 3; i++) {

        for (int j = 0; j < 3; j++)

            scanf("%d", &matrix1[i][j]);

    }

    printf("Enter the elements of second 3x3 matrix: \n");

    for (int i = 0; i < 3; i++) {

        for (int j = 0; j < 3; j++)

            scanf("%d", &matrix2[i][j]);

    }

    // Sum of two matrices

    printf("Sum of two matrices: \n");

    for (int i = 0; i < 3; i++) {

        for (int j = 0; j < 3; j++) {

            sum[i][j] = matrix1[i][j] + matrix2[i][j];

            printf("%d ", sum[i][j]);

        }

        printf("\n");

    }

    // Product of two matrices

    printf("Product of two matrices: \n");

    for (int i = 0; i < 3; i++) {

        for (int j = 0; j < 3; j++) {

            product[i][j] = 0;

            for (int k = 0; k < 3; k++)

                product[i][j] += matrix1[i][k] \* matrix2[k][j];

            printf("%d ", product[i][j]);

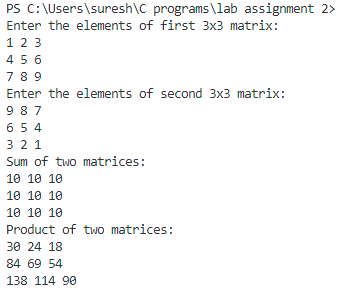
        }

        printf("\n");

    }

}

1. **OUTPUT**

****

1. **Write a program to sort an array in ascending order.**
2. **PROGRAM**

#include<stdio.h>

int main() {

    int n, temp;

    printf("Enter the number of elements in the array: ");

    scanf("%d", &n);

    int arr[n];

    printf("Enter the elements of the array: ");

    for(int i = 0; i < n; i++) {

        scanf("%d", &arr[i]);

    }

    for(int i = 0; i < n; i++) {

        for(int j = i+1; j < n; j++) {

            if(arr[i] > arr[j]) {

                temp = arr[i];

                arr[i] = arr[j];

                arr[j] = temp;

            }

        }

    }

    printf("The array in ascending order is: ");

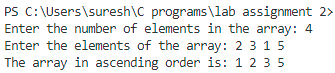
    for(int i = 0; i < n; i++) {

        printf("%d ", arr[i]);

    }

}

1. **OUTPUT**

****

1. **Write a program to find smallest element in an array.**
2. **PROGRAM**

#include<stdio.h>

int main() {

    int n;

    printf("Enter the number of elements in the array: ");

    scanf("%d", &n);

    int arr[n];

    printf("Enter the elements of the array: ");

    for(int i = 0; i < n; i++) {

        scanf("%d", &arr[i]);

    }

    int smallest = arr[0];

    for(int i = 1; i < n; i++) {

        if(arr[i] < smallest) {

            smallest = arr[i];

        }

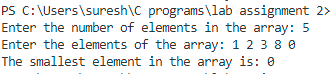
    }

    printf("The smallest element in the array is: %d", smallest);

    return 0;

}

1. **OUTPUT**

****

1. **Write a program to get a string as input and print the length of string, reverse of the string.**
   1. **Using String Library Function**
   2. **Using your own function**
2. **PROGRAM**

#include <stdio.h>

#include <string.h>

void reverse(char \*string, int length)

{

    for (int i = 0; i < length; i++)

    {

        char temp = string[i];

        string[i] = string[length - i - 1];

        string[length - i - 1] = temp;

    }

    printf("The reverse of the string is: %s\n", string);

}

void find\_length(char \*string)

{

    int length = 0;

    while (string[length] != '\0')

    {

        length++;

    }

    printf("The length of the string is: %d\n", length);

}

int main()

{

    char string[40];

    printf("Enter a string: ");

    fgets(string, 40, stdin);

    // length using library function

    int length = strlen(string);

    printf("The length of the string using library function is: %d\n", length);

    // length using custom function

    find\_length(string);

    // reverse of string using library function

    printf("The reverse of the string using library function is: %s\n", strrev(string));

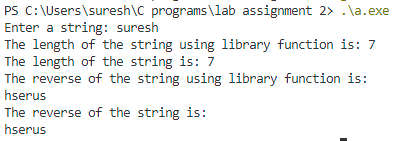
    // reverse of string using custom function

    reverse(string, length);

    return 0;

}

1. **OUTPUT**

****

1. **Write a program that takes input of two numbers and any one operator in (+,-,\*, /, %) as input and pass those numbers and an operator to the function. The function should calculate the result of two numbers as indicated by operator and return the result. Display the result of computation in your program.**
   1. **PROGRAM**

#include <stdio.h>

int calculate(float a, float b, char op)

{

    switch (op)

    {

    case '+':

        return a + b;

    case '-':

        return a - b;

    case '\*':

        return a \* b;

    case '/':

        if (b == 0)

        {

            printf("Error: Division by zero\n");

            return 0;

        }

        return a / b;

    default:

        printf("Error: Invalid operator\n");

        return 0;

    }

}

int main()

{

    float num1, num2;

    char op;

    printf("Enter the first number: ");

    scanf("%f", &num1);

    printf("Enter the operator: ");

    scanf(" %c", &op);

    printf("Enter the second number: ");

    scanf("%f", &num2);

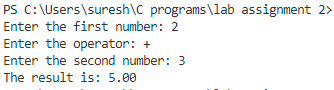
    float result = calculate(num1, num2, op);

    printf("The result is: %.2f\n", result);

    return 0;

}

* 1. **OUTPUT**

****

1. **Write a program to find factorial of given integer using recursion.**
   1. **PROGRAM**

#include<stdio.h>

int factorial(int n)

{

    if (n == 0)

    {

        return 1;

    }

    return n \* factorial(n - 1);

}

int main() {

    int n;

    printf("Enter a number: ");

    scanf("%d", &n);

    printf("The factorial of %d is: %d\n", n, factorial(n));

    return 0;

}

* 1. **OUTPUT**

****

1. **Write a program to find first 10 numbers in Fibonacci series using recursion.**
   1. **PROGRAM**

#include <stdio.h>

int fibonacci(int n)

{

    if (n == 0)

    {

        return 0;

    }

    if (n == 1)

    {

        return 1;

    }

    return fibonacci(n - 1) + fibonacci(n - 2);

}

int main()

{

    for (int i = 0; i < 10; i++)

    {

        printf("%d ", fibonacci(i));

    }

    return 0;

}

* 1. **OUTPUT**

****

1. **Write a program to find umber a times a vowel ‘e’ appears in a given string.**
   1. **PROGRAM**

#include <stdio.h>

int main()

{

    char str[100];

    printf("Enter a string: ");

    fgets(str, 100, stdin);

    int count = 0;

    for (int i = 0; str[i] != '\0'; i++)

    {

        if (str[i] == 'e')

        {

            count++;

        }

    }

    printf("The number of 'e' in the string is: %d\n", count);

    return 0;

}

* 1. **OUTPUT**

****

1. **Write a program defining an array with dynamic memory allocation of integers and compute the sum of the array using function of your own.**
   1. **PROGRAM**

#include <stdio.h>

#include <stdlib.h>

void sum(int \*arr, int n)

{

    int sum = 0;

    for (int i = 0; i < n; i++)

    {

        sum += arr[i];

    }

    printf("The sum of the elements is: %d\n", sum);

}

int main()

{

    int n;

    printf("Enter the number of elements: ");

    scanf("%d", &n);

    int \*arr = (int \*)malloc(n \* sizeof(int));

    printf("Enter the elements: ");

    for (int i = 0; i < n; i++)

    {

        scanf("%d", &arr[i]);

    }

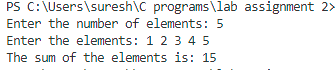
    sum(arr, n);

    free(arr);

    return 0;

}

* 1. **OUTPUT**

****

1. **Write a program to swap two numbers defining a function swap(int, int ) using:**
2. **Pass by value**
   * 1. **PROGRAM**

#include<stdio.h>

void swap(int a, int b)

{

    int temp;

    temp = a;

    a = b;

    b = temp;

    printf("After swapping inside a swap function a = %d and b = %d\n", a, b);

}

int main()

{

    int a, b, temp;

    printf("Enter the value of a and b: ");

    scanf("%d %d", &a, &b);

    printf("Before swapping a = %d and b = %d\n", a, b);

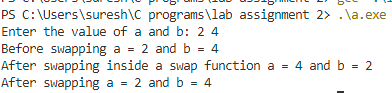
    swap(a, b);

    printf("After swapping a = %d and b = %d\n", a, b);

    return 0;

}

* + 1. **OUTPUT**

****

**b) Pass by reference**

* 1. **PROGRAM**

#include<stdio.h>

void swap(int \*a, int \*b)

{

    int temp;

    temp = \*a;

    \*a = \*b;

    \*b = temp;

    printf("After swapping inside a swap function a = %d and b = %d\n", \*a, \*b);

}

int main()

{

    int a, b, temp;

    printf("Enter the value of a and b: ");

    scanf("%d %d", &a, &b);

    printf("Before swapping a = %d and b = %d\n", a, b);

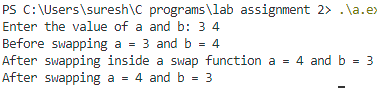
    swap(&a, &b);

    printf("After swapping a = %d and b = %d\n", a, b);

    return 0;

}

* 1. **OUTPUT**

****

1. **Write a program defining a structure to store the data for a student with fields ( rollno, f\_name, l\_name, address, mobileno) , input the data for a student and display the record in appropriate format.**
   1. **PROGRAM**

#include<stdio.h>

struct student

{

    int rollno;

    char f\_name[20];

    char l\_name[20];

    char address[50];

    char mobile\_no[11];

};

void display(struct student s)

{

    printf("\nStudent details:\n");

    printf("Roll no: %d\n", s.rollno);

    printf("First name: %s\n", s.f\_name);

    printf("Last name: %s\n", s.l\_name);

    printf("Address: %s\n", s.address);

    printf("Mobile no: %s\n", s.mobile\_no);

}

int main()

{

    struct student s1;

    printf("Enter roll no: ");

    scanf("%d", &s1.rollno);

    printf("Enter first name: ");

    scanf("%s", &s1.f\_name);

    printf("Enter last name: ");

    scanf("%s", &s1.l\_name);

    printf("Enter address: ");

    scanf("%s", &s1.address);

    printf("Enter mobile no: ");

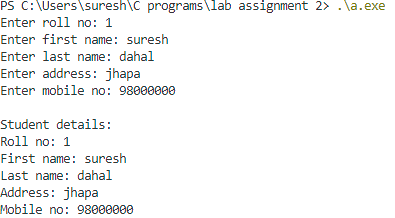
    scanf("%s", &s1.mobile\_no);

    display(s1);

    return 0;

}

* 1. **OUTPUT**

****

1. **Write a program to pass array to a function using pointer and return array from a function using pointer.**
   1. **PROGRAM**

#include <stdio.h>

int \*fun(int arr[])

{

    for (int i = 0; i < 5; i++)

    {

        arr[i] = arr[i] \* arr[i];

    }

    return arr;

}

int main()

{

    int arr[5] = {1, 2, 3, 4, 5};

    int \*square\_arr = fun(arr);

    printf("Square of array: ");

    for (int i = 0; i < 5; i++)

    {

        printf("%d ", \*(square\_arr + i));

    }

    return 0;

}

* 1. **OUTPUT**

****

1. **Write a program that illustrates the similarity between array and pointer.**
   1. **PROGRAM**

#include<stdio.h>

int main()

{

    int arr[5] = {1, 2, 3, 4, 5};

    int \*ptr = arr;

    // int \*ptr = &arr[0]; // Both are same

    // array name is a pointer to the first element of the array

    // arr[i] = \*(arr + i) = \*(ptr + i) = ptr[i]

    printf("Value of arr[0]: %d\n", \*ptr);

    printf("Value of arr[1]: %d\n", \*(ptr + 1));

    printf("Value of arr[2]: %d\n", ptr[2]);

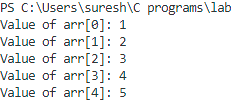
    printf("Value of arr[3]: %d\n", \*(ptr + 3));

    printf("Value of arr[4]: %d\n", \*(ptr + 4));

    return 0;

}

* 1. **OUTPUT**

****

1. **Write a program using array of structure to store the data for 10 employees with fields (Essn, f\_name, l\_name, address, mobileno), input the data for n employees and display the record in appropriate format.**
   1. **PROGRAM**

#include<stdio.h>

struct Employee {

    int essn;

    char fname[20];

    char lname[20];

    char address[50];

    char mobile[11];

};

int main() {

    struct Employee emp[10];

    printf("Enter the details of 10 employees:\n");

    for (int i = 0; i < 10; i++) {

        printf("\nEnter the essn of employee %d: ", i + 1);

        scanf("%d", &emp[i].essn);

        printf("Enter the first name of employee %d: ", i + 1);

        scanf("%s", emp[i].fname);

        printf("Enter the last name of employee %d: ", i + 1);

        scanf("%s", emp[i].lname);

        printf("Enter the address of employee %d: ", i + 1);

        scanf("%s", emp[i].address);

        printf("Enter the mobile number of employee %d: ", i + 1);

        scanf("%s", emp[i].mobile);

    }

    printf("\nDetails of 10 employees:\n");

    printf("\nESSN\tFirst name\tLast name\tAddress\tMobile\n");

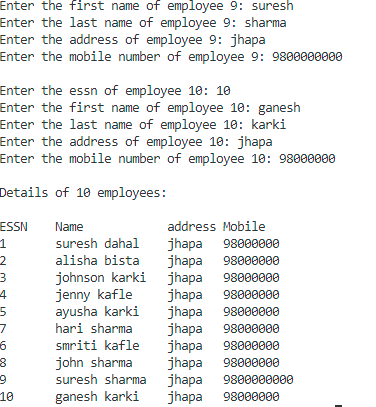
    for (int i = 0; i < 10; i++) {

        printf("%d\t%s\t%s\t%s\t%s\n", emp[i].essn, emp[i].fname, emp[i].lname, emp[i].address, emp[i].mobile);

    }

}

* 1. **OUTPUT**

****

1. **Write a program defining a union to store the data for a student with fields (rollno, f\_name, l\_name, address, mobileno). Illustrate the property of Union by accessing only one data member at a time.**
   1. **PROGRAM**

#include <stdio.h>

union Student

{

    int roll\_number;

    char fname[20];

    char lname[20];

    char address[50];

    char mobile[11];

};

int main()

{

    union Student std;

    printf("Enter the details of 10 students:\n");

    printf("\nEnter the roll\_number of student: ");

    scanf("%d", &std.roll\_number);

    printf("Roll number before entering name %d\n", std.roll\_number);

    printf("Enter the first name of student:");

    scanf("%s", std.fname);

    printf("First name before entering last name %s\n", std.fname);

    printf("Enter the last name of student:");

    scanf("%s", std.lname);

    printf("Last name before entering address %s\n", std.lname);

    printf("Enter the address of student:");

    scanf("%s", std.address);

    printf("Address before entering mobile number %s\n", std.address);

    printf("Enter the mobile number of student:");

    scanf("%s", std.mobile);

    printf("\nDetails of 2 students:\n");

    printf("\nroll\_number %d\n", std.roll\_number);

    printf("First name %s\n", std.fname);

    printf("Last name %s\n", std.lname);

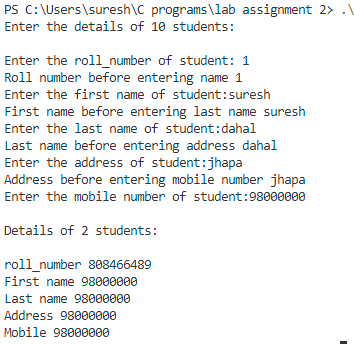
    printf("Address %s\n", std.address);

    printf("Mobile %s\n", std.mobile);

    return 0;

}

* 1. **OUTPUT**

****

1. **Write a program to prompt user to input filename and read the content of file and display in screen.**
   1. **PROGRAM**

#include<stdio.h>

int main() {

    FILE \*fptr;

    char file\_name[100];

    printf("Enter the file name: ");

    scanf("%s", file\_name);

    fptr = fopen(file\_name, "r");

    if (fptr == NULL) {

        printf("Error! File not found.");

        return 1;

    }

    printf("Content from the file: \n");

    printf("----------------------------\n");

    char ch;

    while ((ch = fgetc(fptr)) != EOF) {

        printf("%c", ch);

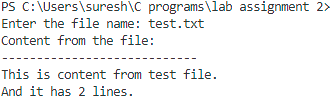
    }

    fclose(fptr);

    return 0;

}

* 1. **OUTPUT**

****

1. **Write a program to read from a text file and count the number of lines and characters in that file.**
   1. **PROGRAM**

#include <stdio.h>

int main()

{

    FILE \*fptr;

    char filename[100], ch;

    int total\_character = 0, total\_lines = 0;

    printf("Enter the filename to open: ");

    scanf("%s", filename);

    fptr = fopen(filename, "r");

    if (fptr == NULL)

    {

        printf("Error! File not found.");

        return 1;

    }

    while ((ch = fgetc(fptr)) != EOF)

    {

        total\_character++;

        if (ch == '\n' || ch == '\0')

            total\_lines++;

    }

    printf("Total characters: %d\n", total\_character);

    printf("Total lines: %d\n", total\_lines);

    fclose(fptr);

    return 0;

}

* 1. **OUTPUT**

****

1. **Write a program to read a text file and copy all contents in the new file.**
   1. **PROGRAM**

#include<stdio.h>

int main()

{

    FILE \*fptr1, \*fptr2;

    char filename[100], c;

    printf("Enter the filename to open for reading: ");

    scanf("%s", filename);

    fptr1 = fopen(filename, "r");

    if (fptr1 == NULL)

    {

        printf("Cannot open file %s \n", filename);

        return 0;

    }

    printf("Enter the filename to open for writing: ");

    scanf("%s", filename);

    fptr2 = fopen(filename, "w");

    if (fptr2 == NULL)

    {

        printf("Cannot open file %s \n", filename);

        return 0;

    }

    c = fgetc(fptr1);

    while (c != EOF)

    {

        fputc(c, fptr2);

        c = fgetc(fptr1);

    }

    printf("Contents copied to %s", filename);

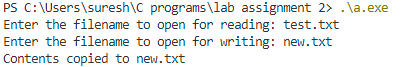
    fclose(fptr1);

    fclose(fptr2);

    return 0;

}

* 1. **OUTPUT**

****