**LABORATORY MANUAL**

|  |  |
| --- | --- |
| **Lab Course name** | **PROGRAMMING FOR PROBLEM SOLVING LAB** |
| **Lab Course code** | **PCS 101** |
| **Semester** | **II** |
| **Academic year** | **2024-2025** |
| **Student Name** |  |
| **Program name** | **B.Tech CSE** |
| **School** | **UIT** |

**INDEX**

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **S.N** | **Name of the experiment** | **Page.no** | **Date** | **Signature** |
| 1. | Write a Program to display the “Hello” word. |  |  |  |
| 2. | Write a program to calculate the area of Rectangle using user’s input. |  |  |  |
| 3. | Write a program to check weather a given number is even or odd. |  |  |  |
| 4. | Write a program to find the largest of three numbers using nested if else. |  |  |  |
| 5. | Write a program to find whether the given number is Armstrong or not. |  |  |  |
| 6. | Write a program to construct a Fibonacci series up to n terms. |  |  |  |
| 7 | Write a program to check weather a given number is prime or not. |  |  |  |
| 8 | Write a program to print the different pattern. |  |  |  |
| 9 | Write a program to sum of the elements of array. |  |  |  |
| 10. | Write a program to demonstrate the string functions. |  |  |  |
| 11. | Write a program for addition of two matrix. |  |  |  |
| 12. | Write a program to find the transpose of a given matrix & check whether it is symmetric or not |  |  |  |
| 13. | Write a program to calculate the factorial for given number using recursive function. |  |  |  |
| 14. | Write a program printing the elements of structure array using pointers |  |  |  |
| 15. | Write a program to open text file |  |  |  |

|  |  |  |
| --- | --- | --- |
| **EXPERIMENT NO.: 1** | | |
| **Title of Experiment:** Write a Program to display the “Hello” word. | | |
| Program: B.Tech CSE | Course: PROGRAMMING FOR PROBLEM SOLVING LAB  Course Code: PCS 101(L) | Semester: II |
| Page no: ¼ | | |

#include <stdio.h>

int main() {

printf("Hello\n"); // Prints "Hello" to the console

return 0; // Indicates that the program executed successfully

}

gcc hello.c -o hello

./hello

|  |  |  |
| --- | --- | --- |
| **EXPERIMENT NO.: 2** | | |
| **Title of Experiment:** Write a program to calculate the area of Rectangle using user’s input. | | |
| Program: B.Tech CSE | Course: PROGRAMMING FOR PROBLEM SOLVING LAB  Course Code: PCS 101(L) | Semester: II |
| Page no: ¼ | | |

#include <stdio.h>

int main() {

float length, width, area;

// Prompt user for input

printf("Enter the length of the rectangle: ");

scanf("%f", &length); // Read the length from user

printf("Enter the width of the rectangle: ");

scanf("%f", &width); // Read the width from user

// Calculate area

area = length \* width;

// Display the result

printf("The area of the rectangle is: %.2f\n", area);

return 0;

}

|  |  |  |
| --- | --- | --- |
| **EXPERIMENT NO.: 3** | | |
| **Title of Experiment:** Write a program to check weather a given number is even or odd. | | |
| Program: B.Tech CSE | Course: PROGRAMMING FOR PROBLEM SOLVING LAB  Course Code: PCS 101(L) | Semester: II |
| Page no: ¼ | | |

#include <stdio.h>

int main() {

int number;

// Prompt user for input

printf("Enter an integer: ");

scanf("%d", &number);

// Check if the number is even or odd

if (number % 2 == 0) {

printf("%d is an even number.\n", number);

} else {

printf("%d is an odd number.\n", number);

}

return 0;

}

|  |  |  |
| --- | --- | --- |
| **EXPERIMENT NO.: 4** | | |
| **Title of Experiment:** Write a program to find the largest of three numbers using nested if else. | | |
| Program: B.Tech CSE | Course: PROGRAMMING FOR PROBLEM SOLVING LAB  Course Code: PCS 101(L) | Semester: II |
| Page no: ¼ | | |

#include <stdio.h>

int main() {

int num1, num2, num3;

// Prompt user for input

printf("Enter three numbers:\n");

scanf("%d %d %d", &num1, &num2, &num3);

// Nested if-else to find the largest number

if (num1 >= num2) {

if (num1 >= num3) {

printf("The largest number is: %d\n", num1);

} else {

printf("The largest number is: %d\n", num3);

}

} else {

if (num2 >= num3) {

printf("The largest number is: %d\n", num2);

} else {

printf("The largest number is: %d\n", num3);

}

}

return 0;

}

|  |  |  |
| --- | --- | --- |
| **EXPERIMENT NO.: 5** | | |
| **Title of Experiment:** Write a program to find whether the given number is Armstrong or not. | | |
| Program: B.Tech CSE | Course: Introduction to Python Programming Lab  Course Code: PCS 151(L) | Semester: I |
| Page no: ¼ | | |

#include <stdio.h>

#include <math.h>

int main() {

int num, originalNum, remainder, result = 0, n = 0;

// Prompt user for input

printf("Enter an integer: ");

scanf("%d", &num);

originalNum = num;

// Find the number of digits in the number

while (originalNum != 0) {

originalNum /= 10;

n++;

}

originalNum = num;

// Calculate the sum of the digits raised to the power n

while (originalNum != 0) {

remainder = originalNum % 10;

result += pow(remainder, n);

originalNum /= 10;

}

// Check if the number is an Armstrong number

if (result == num) {

printf("%d is an Armstrong number.\n", num);

} else {

printf("%d is not an Armstrong number.\n", num);

}

return 0;

}

|  |  |  |
| --- | --- | --- |
| **EXPERIMENT NO.: 6** | | |
| **Title of Experiment:** Write a program to construct a Fibonacci series up to n terms. | | |
| Program: B.Tech CSE | Course: Introduction to Python Programming Lab  Course Code: PCS 151(L) | Semester: I |
| Page no: ¼ | | |

**#include <stdio.h>**

**int main() {**

**int n, first = 0, second = 1, next, i;**

**// Prompt user for the number of terms**

**printf("Enter the number of terms: ");**

**scanf("%d", &n);**

**printf("Fibonacci Series up to %d terms:\n", n);**

**// Handle the special cases where n <= 0**

**if (n <= 0) {**

**printf("Please enter a positive integer.\n");**

**} else {**

**for (i = 1; i <= n; i++) {**

**if (i == 1) {**

**printf("%d ", first);**

**continue;**

**}**

**if (i == 2) {**

**printf("%d ", second);**

**continue;**

**}**

**next = first + second; // Calculate the next term**

**first = second; // Update the first term**

**second = next; // Update the second term**

**printf("%d ", next);**

**}**

**}**

**printf("\n");**

**return 0;**

**}**

|  |  |  |
| --- | --- | --- |
| **EXPERIMENT NO.: 7** | | |
| **Title of Experiment:** Write a program to check weather a given number is prime or not. | | |
| Program: B.Tech CSE | Course: Introduction to Python Programming Lab  Course Code: PCS 151(L) | Semester: I |
| Page no: ¼ | | |

#include <stdio.h>

int main() {

int num, i, isPrime = 1;

// Prompt user for input

printf("Enter a positive integer: ");

scanf("%d", &num);

// Handle special cases

if (num <= 1) {

printf("%d is not a prime number.\n", num);

return 0;

}

// Check for factors other than 1 and itself

for (i = 2; i <= num / 2; i++) {

if (num % i == 0) {

isPrime = 0; // Number is not prime

break;

}

}

// Output the result

if (isPrime) {

printf("%d is a prime number.\n", num);

} else {

printf("%d is not a prime number.\n", num);

}

return 0;

}

|  |  |  |
| --- | --- | --- |
| **EXPERIMENT NO.: 8** | | |
| **Title of Experiment:** Write a program to print the different pattern. | | |
| Program: B.Tech CSE | Course: Introduction to Python Programming Lab  Course Code: PCS 151(L) | Semester: I |
| Page no: ¼ | | |

#include <stdio.h>

int main() {

int n, i, j;

// Prompt user for input

printf("Enter the number of rows for the pyramid: ");

scanf("%d", &n);

printf("Pyramid Pattern:\n");

// Outer loop for rows

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

// Print spaces for alignment

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

printf(" ");

}

// Print stars

for (j = 1; j <= (2 \* i - 1); j++) {

printf("\*");

}

printf("\n");

}

return 0;

}

\*

\*\*\*

\*\*\*\*\*

\*\*\*\*\*\*\*

\*\*\*\*\*\*\*\*\*

#include <stdio.h>

int main() {

int n, i, j;

// Prompt user for input

printf("Enter the number of rows for the triangle: ");

scanf("%d", &n);

printf("Right-Angled Triangle Pattern:\n");

// Outer loop for rows

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

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

printf("\*");

}

printf("\n");

}

return 0;

}

\*

\*\*

\*\*\*

\*\*\*\*

\*\*\*\*\*

#include <stdio.h>

int main() {

int n, i, j;

// Prompt user for input

printf("Enter the number of rows for the inverted pyramid: ");

scanf("%d", &n);

printf("Inverted Pyramid Pattern:\n");

// Outer loop for rows

for (i = n; i >= 1; i--) {

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

printf(" ");

}

for (j = 1; j <= (2 \* i - 1); j++) {

printf("\*");

}

printf("\n");

}

return 0;

}

\*\*\*\*\*\*\*\*\*

\*\*\*\*\*\*\*

\*\*\*\*\*

\*\*\*

\*

#include <stdio.h>

int main() {

int n, i, j;

// Prompt user for input

printf("Enter the number of rows for the diamond: ");

scanf("%d", &n);

printf("Diamond Shape Pattern:\n");

// Upper half of the diamond

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

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

printf(" ");

}

for (j = 1; j <= (2 \* i - 1); j++) {

printf("\*");

}

printf("\n");

}

// Lower half of the diamond

for (i = n - 1; i >= 1; i--) {

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

printf(" ");

}

for (j = 1; j <= (2 \* i - 1); j++) {

printf("\*");

}

printf("\n");

}

return 0;

}

\*

\*\*\*

\*\*\*\*\*

\*\*\*\*\*\*\*

\*\*\*\*\*\*\*\*\*

\*\*\*\*\*\*\*

\*\*\*\*\*

\*\*\*

\*

#include <stdio.h>

int main() {

int n, i, j, coeff;

// Prompt user for input

printf("Enter the number of rows for Pascal's Triangle: ");

scanf("%d", &n);

printf("Pascal's Triangle:\n");

for (i = 0; i < n; i++) {

// Print leading spaces

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

printf(" ");

}

// Print coefficients

coeff = 1;

for (j = 0; j <= i; j++) {

printf("%d ", coeff);

coeff = coeff \* (i - j) / (j + 1);

}

printf("\n");

}

return 0;

}

1

1 1

1 2 1

1 3 3 1

1 4 6 4 1

|  |  |  |
| --- | --- | --- |
| **EXPERIMENT NO.: 9** | | |
| **Title of Experiment:** Write a program to sum of the elements of array. | | |
| Program: B.Tech CSE | Course: Introduction to Python Programming Lab  Course Code: PCS 151(L) | Semester: I |
| Page no: ¼ | | |

#include <stdio.h>

int main() {

int n, sum = 0;

// Prompt user for the number of elements in the array

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

scanf("%d", &n);

int arr[n]; // Declare an array of size 'n'

// Input elements in the array

printf("Enter %d elements:\n", n);

for (int i = 0; i < n; i++) {

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

}

// Calculate the sum of array elements

for (int i = 0; i < n; i++) {

sum += arr[i];

}

// Output the sum

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

return 0;

}

|  |  |  |
| --- | --- | --- |
| **EXPERIMENT NO.: 10** | | |
| **Title of Experiment:** Write a program to demonstrate the string functions. | | |
| Program: B.Tech CSE | Course: Introduction to Python Programming Lab  Course Code: PCS 151(L) | Semester: I |
| Page no: ¼ | | |

#include <stdio.h>

#include <string.h>

int main() {

char str1[100], str2[100], str3[100];

int length;

// Input two strings

printf("Enter the first string: ");

gets(str1); // Input first string

printf("Enter the second string: ");

gets(str2); // Input second string

// 1. strlen() - Calculate the length of the string

length = strlen(str1);

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

// 2. strcpy() - Copy contents of one string to another

strcpy(str3, str1);

printf("Copy of the first string is: %s\n", str3);

// 3. strcat() - Concatenate two strings

strcat(str1, str2);

printf("Concatenated string: %s\n", str1);

// 4. strcmp() - Compare two strings

if (strcmp(str1, str2) == 0)

printf("The two strings are equal.\n");

else

printf("The two strings are not equal.\n");

// 5. strchr() - Find the first occurrence of a character in a string

char ch = 'a';

char \*ptr = strchr(str1, ch);

if (ptr != NULL)

printf("First occurrence of '%c' in the string is at position: %ld\n", ch, ptr - str1 + 1);

else

printf("Character '%c' not found in the string.\n", ch);

return 0;

}

|  |  |  |
| --- | --- | --- |
| **EXPERIMENT NO.: 11** | | |
| **Title of Experiment:** Write a program for addition of two matrix. | | |
| Program: B.Tech CSE | Course: Introduction to Python Programming Lab  Course Code: PCS 151(L) | Semester: I |
| Page no: ¼ | | |

#include <stdio.h>

int main() {

int m, n, i, j;

// Prompt user for matrix dimensions

printf("Enter the number of rows and columns of the matrices: ");

scanf("%d %d", &m, &n);

// Declare two matrices and a result matrix

int matrix1[m][n], matrix2[m][n], sum[m][n];

// Input elements for the first matrix

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

for (i = 0; i < m; i++) {

for (j = 0; j < n; j++) {

printf("Element [%d][%d]: ", i + 1, j + 1);

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

}

}

// Input elements for the second matrix

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

for (i = 0; i < m; i++) {

for (j = 0; j < n; j++) {

printf("Element [%d][%d]: ", i + 1, j + 1);

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

}

}

// Perform matrix addition

for (i = 0; i < m; i++) {

for (j = 0; j < n; j++) {

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

}

}

// Display the result

printf("\nSum of the two matrices:\n");

for (i = 0; i < m; i++) {

for (j = 0; j < n; j++) {

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

}

printf("\n");

}

return 0;

}

|  |  |  |
| --- | --- | --- |
| **EXPERIMENT NO.: 12** | | |
| **Title of Experiment:** Write a program to find the transpose of a given matrix & check whether it is symmetric or not | | |
| Program: B.Tech CSE | Course: Introduction to Python Programming Lab  Course Code: PCS 151(L) | Semester: I |
| Page no: ¼ | | |

#include <stdio.h>

int main() {

int m, n, i, j;

int symmetric = 1; // Flag to check symmetry

// Input matrix dimensions

printf("Enter the number of rows and columns of the matrix: ");

scanf("%d %d", &m, &n);

// Declare the matrix and its transpose

int matrix[m][n], transpose[n][m];

// Input matrix elements

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

for (i = 0; i < m; i++) {

for (j = 0; j < n; j++) {

printf("Element [%d][%d]: ", i + 1, j + 1);

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

}

}

// Calculate the transpose of the matrix

for (i = 0; i < m; i++) {

for (j = 0; j < n; j++) {

transpose[j][i] = matrix[i][j];

}

}

// Display the original matrix

printf("\nOriginal matrix:\n");

for (i = 0; i < m; i++) {

for (j = 0; j < n; j++) {

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

}

printf("\n");

}

// Display the transpose of the matrix

printf("\nTranspose of the matrix:\n");

for (i = 0; i < n; i++) {

for (j = 0; j < m; j++) {

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

}

printf("\n");

}

// Check if the matrix is square and symmetric

if (m == n) {

for (i = 0; i < m; i++) {

for (j = 0; j < n; j++) {

if (matrix[i][j] != transpose[i][j]) {

symmetric = 0;

break;

}

}

}

if (symmetric) {

printf("\nThe matrix is symmetric.\n");

} else {

printf("\nThe matrix is not symmetric.\n");

}

} else {

printf("\nThe matrix is not square, so it cannot be symmetric.\n");

}

return 0;

}

|  |  |  |
| --- | --- | --- |
| **EXPERIMENT NO.: 13** | | |
| **Title of Experiment:** Write a program to calculate the factorial for given number using recursive function. | | |
| Program: B.Tech CSE | Course: Introduction to Python Programming Lab  Course Code: PCS 151(L) | Semester: I |
| Page no: ¼ | | |

#include <stdio.h>

// Function to calculate factorial recursively

int factorial(int n) {

// Base case: factorial of 0 or 1 is 1

if (n == 0 || n == 1) {

return 1;

}

// Recursive case: factorial of n is n \* factorial of (n-1)

else {

return n \* factorial(n - 1);

}

}

int main() {

int num;

// Input: prompt user for the number

printf("Enter a number to calculate its factorial: ");

scanf("%d", &num);

// Check for negative input

if (num < 0) {

printf("Factorial is not defined for negative numbers.\n");

} else {

// Call the recursive function and display the result

printf("Factorial of %d is %d\n", num, factorial(num));

}

return 0;

}

|  |  |  |
| --- | --- | --- |
| **EXPERIMENT NO.: 14** | | |
| **Title of Experiment:** Write a program printing the elements of structure array using pointers. | | |
| Program: B.Tech CSE | Course: Introduction to Python Programming Lab  Course Code: PCS 151(L) | Semester: I |
| Page no: ¼ | | |

#include <stdio.h>

// Define a structure to store information

struct Student {

int roll\_no;

char name[50];

float marks;

};

int main() {

int n;

// Input: number of students

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

scanf("%d", &n);

// Declare an array of structures

struct Student students[n];

// Input student details using structure array

for (int i = 0; i < n; i++) {

printf("\nEnter details for student %d:\n", i + 1);

printf("Roll Number: ");

scanf("%d", &students[i].roll\_no);

printf("Name: ");

getchar(); // To consume newline character left by previous scanf

fgets(students[i].name, sizeof(students[i].name), stdin); // Read string with spaces

printf("Marks: ");

scanf("%f", &students[i].marks);

}

// Pointer to the structure array

struct Student \*ptr = students;

// Print student details using pointer

printf("\nStudent Details:\n");

for (int i = 0; i < n; i++) {

printf("\nStudent %d:\n", i + 1);

printf("Roll Number: %d\n", (ptr + i)->roll\_no);

printf("Name: %s", (ptr + i)->name); // fgets already includes the newline

printf("Marks: %.2f\n", (ptr + i)->marks);

}

return 0;

}

|  |  |  |
| --- | --- | --- |
| **EXPERIMENT NO.: 15** | | |
| **Title of Experiment:** Program to illustrate the concept of method overriding in python | | |
| Program: B.Tech CSE | Course: Introduction to Python Programming Lab  Course Code: PCS 151(L) | Semester: I |
| Page no: ¼ | | |

#include <stdio.h>

int main() {

FILE \*file;

char data[100];

// Open the file in append mode

file = fopen("example.txt", "a");

// Check if the file was opened successfully

if (file == NULL) {

printf("Error opening the file.\n");

return 1;

}

// Input data to be appended

printf("Enter text to append to the file: ");

fgets(data, sizeof(data), stdin); // Read input including spaces

// Write data to the file

fprintf(file, "%s", data);

// Close the file

fclose(file);

printf("Data successfully appended to the file.\n");

return 0;

}