PRATHYUSHA ENGINEERING COLLEGE

DEPARTMENT OF COMPUTER SCIENCE AND ENGINEERING

LAB MANUAL

for

CS3271 – PROGRAMMING IN C LABORATORY

(Regulation 2021, II Semester)

ACADEMIC YEAR: 2021 – 2022

(Even Semester)

PREPARED BY

E.SHIMONA,

Assistant Professor / CSE

PRATHYUSHA ENGINEERING COLLEGE

VISION

To emerge as a premier technical, engineering and management institution in the country by imparting quality education and thus facilitate our students to blossom in to dynamic professional so that they play a vital role for the progress of the nation and for a peaceful co-existence of our fellow human being.

MISSION

Prathyusha Engineering College will strive to emerge as a premier Institution in the country by
- To provide state of art infrastructure facilities
- Imparting quality education and training through qualified, experienced and committed members of the faculty
- Empowering the youth by providing professional leadership
- Developing centers of excellence in frontiers areas of Engineering, Technology and Management
- Networking with Industry, Corporate and Research Organizations
- Promoting Institute-Industry partnership for the peace and prosperity of the nation

DEPARTMENT OF COMPUTER SCIENCE AND ENGINEERING

VISION

Our Vision is to build a strong teaching & research environment in the field of computer science and engineering for developing a team of young dynamic computer science engineers, researchers, future entrepreneurs who are adaptive to respond to the challenges of 21st century. Our commitment lies in producing disciplined human individuals, capable of contributing solutions to solve problems faced by our society.

MISSION
- To provide a quality undergraduate and graduate education in both the theoretical and applied foundations of computer science and engineering.
- To train the students to effectively apply this education to solve real-world problems, thus amplifying their potential for lifelong high-quality careers and gives them a competitive advantage in the ever-changing and challenging global work environment of the 21st century.
- To initiate collaborative real-world industrial projects with industries and academic institutions to inculcate facilities in the arena of Research & Development
- To prepare them with an understanding of their professional and ethical responsibilities

PROGRAMME EDUCATIONAL OBJECTIVES

PEO-1: To train the graduates to be excellent in computing profession by updating technical skill-sets and applying new ideas as the technology evolves.

PEO-2: To enable the graduates to excel in professional career and /or higher education by acquiring knowledge in mathematical, computing and engineering principles.

PEO-3: To enable the graduates, to be competent to grasp, analyze, design, and create new products and solutions for the real time problems that are technically advanced economically feasible and socially acceptable

PEO- 4: To enable the graduates to pursue a productive career as a member of multi-disciplinary and cross-functional teams, with an appreciation for the value of ethic and cultural diversity and an ability to relate engineering issues to broader social context.

PROGRAMME OUTCOMES AND PROGRAMME SPECIFIC OUTCOMES

1. An ability to apply knowledge of computing, mathematics, science and engineering fundamentals appropriate to the discipline.
2. An ability to analyze a problem, and identify and formulate the computing requirements appropriate to its solution.
3. An ability to design, implement, and evaluate a computer-based system, process, component, or program to meet desired needs with appropriate consideration for public health and safety, cultural, societal and environmental considerations.
4. An ability to design and conduct experiments, as well as to analyze and interpret data.
5. An ability to use current techniques, skills, and modern tools necessary for computing practice.
6. An ability to analyze the local and global impact of computing on individuals, organizations, and society.
7. Knowledge of contemporary issues.
8. An understanding of professional, ethical, legal, security and social issues and responsibilities.
9. An ability to function effectively individually and on teams, including diverse and multidisciplinary, to accomplish a common goal.
10. An ability to communicate effectively with a range of audiences.
11. Recognition of the need for and an ability to engage in continuing professional development.
12. An understanding of engineering and management principles and apply these to one's own work, as a member and leader in a team, to manage projects.

PROGRAMME SPECIFIC OBJECTIVES (PSO's)

A graduate of the Computer Science and Engineering Program will able,

PSO1: To Analyze, Design and Develop computer programs / Applications in the areas related to Web-Technologies, Networking, Algorithms, Cloud Computing, Data analytics, Computer Vision, Cyber-Security and Intelligent Systems for efficient design of Computer-based and Mobile-based systems of varying complexity.

PSO2: To use modern software tools (like NS2, MATLAB, OpenCV, etc.) for designing, simulating, analyzing and generating experimental results for real-time problems and case studies

PSO3: To Apply Software Engineering practices and strategies for developing Projects related to emerging technologies.

LIST OF EXPERIMENTS

1. **I/O statements, operators, expressions**

EXPNO: 1A   **SOLVING QUADRATIC EQUATION**

Aim:

To write a C program to find the root of a quadratic equation.

Algorithm:

1. Start
2. Enter three coefficients a,b,c.
3. Calculate d=b*b-4*a a.
4. Check if d=0 the root are equal, and calculate
   root1=root2=-b/(2.0*a)
5. If d>0 then root are real and distinct. Calculate root1=-b/(2.0*a)
         root1= (-b+sqrt (d))/(2*a)
         root1= (-b-sqrt (d))/(2*a)
6. If d>0 then root are imaginary root contains real and imaginary parts.
   Calculate realp =-b(2*a)
         imgp = sqrt(d)/(2*a) and
         root1= real p+imgp
         root1= real p-imgp
7. Display root1, root2.
8. Stop.

Sample Output:

Enter three coefficient of quadratic equation

1

2

1

Roots are equal

Roots 1= root2=-1.000000

Result:

Thus the program is executed successfully.

EXPNO: 1B   **FINDING BIGGEST AMOUNG TWO NUMBERS**

USING TERNARY OPERATOR

Aim:

To write a c program to find biggest among two numbers using ternary operator.

Algorithm:

1. Start.
2. Read two numbers A and B.
3. Check C= (A>B)? A:B. That is if A is bigger than B then C=A otherwise C=B.
4. Print bigger value as C.
5. Stop.

Sample Output:

Enter the value of A: 34

Enter the value of B: 78

The Biggest value is = 78

Result:

Thus the program is executed successfully.

EXPNO: 1C   CONVERTING CENTIGRADE TO FAHRENTHEIT

Aim:

To write a C program to convert centigrade to Fahrenheit.

Algorithm:

1. Start.
2. Read centigrade value C.
3. Calculate F=C*(9/5)+32
4. Print Fahrenheit value F.
5. Stop.

Sample Output:

Conversion of centigrade to Fahrenheit

165

Centigrade value: 165.000000

Formula used

F=C*(9/5)+32

Fahrenheit value: 329.000000

Result:

Thus the program is executed Successfully.

2. Decision-making constructs: if-else, goto, switch-case, break-continue

EXPNO: 2A   **CHECKING LEAP YEAR OR NOT**

Aim:

To write a C program to check leap year or not.

Algorithm:

1. Start.
2. Read year.
3. Check whether year is divisible by 4 and 400 and not divisible by 100.
4. If the condition is true then print year is leap year. Otherwise print year is not leap year.
5. Stop.

Sample Output:

Enter the Value of N: 2000

2000 is a leap year.

Enter the value of N: 1700

1700 is not leap year.

Result:

Thus the program is executed successfully.

EXPNO: 2B   **BIGGEST AMONG TWO NUMBERS USING GOTO**

Aim:

To write a c program to find biggest among two numbers using goto.

Algorithm:

1. Start.
2. Read two numbers a, b.
3. If a>b then go to greater. Otherwise print B is greater than A.
4. In greater section print A is greater than B.
5. Stop.

Sample Output:

Enter two numbers

45

12

A is greater than B

Enter two numbers

45

97

B is greater than A

Result:

Thus the program is executed successfully.

EXPNO: 2C   **SIMULATING SIMPLE CALCULATOR**

Aim:

To write a C program to simulate simple calculator.

Algorithm:

1. Start.
2. Display the menu
3. Read the choice
4. If the choice is 1, then Read two number, Add two numbers, and print the result.
5. If the choice is 2, then Read two number, subtract a-b and print the result.
6. If the choice is 3, then Read two number, multiply two numbers and print result.
7. If the choice is 4, then Read two number, divide a/b and print the result.
8. If the choice is 5, then exit.

Sample Output:

Type your expression (num1 op num2)

65-98

65.00-98.0-33.00

Type your expression (num1 op num2)

65+76

65.00+76.0141.00

Type your expression (num1 op num2)

7*7

7*7=49

Type your expression (num1 op num2)

6/6

6/0=0

Result:

Thus the program is executed Successfully.

EXPNO: 2D   **PRINTING NUMBER DIGITS**

Aim:

   To write a C program to print number digits.

Algorithm:

1. Start.
2. Read n
3. If n is1 then print "ONE"'
    If n is 2 then print "TWO",
    Similarly print till number 9 and number 0.
4. Stop.

Sample Output:

   Enter the value of N: 1

   ONE

   Enter the value of N: 2

   TWO

   Enter the value of N: 3

   THREE

   Enter the value of N: 4

   FOUR

   Enter the value of N: 5

   FIVE

   Enter the value of N: 6

   SIX

   Enter the value of N: 7

   SEVEN

   Enter the value of N: 8

   EIGHT

   Enter the value of N: 9

   NINE

   Enter the value of N: 0

   ZERO

Result:

   Thus the program is executed Successfully.

EXPNO: 2E   **ILLUSTRATION OF CONTINUE STATEMENT**

Aim:

    To write a C program to illustrate continue statement.

Algorithm:

1. Start.
2. Assign j=10
3. For (i=0; i<=j; i++)
   If i==j then continue
   Otherwise print i value
4. Stop.

Sample Output:

    Hello 0

    Hello 1

    Hello 2

    Hello 3

    Hello 4

    Hello 5

    Hello 6

    Hello 7

    Hello 8

    Hello 9

    Hello 10

Result:

    Thus the program is executed successfully.

3. Loops: for, while, do-while

ILLUSTRATION OF FOR EVALUATE $1+3+2+3+3+3\ldots N^3$

Aim:

To write a C program to evaluate the series.

Algorithm:

1. Start
2. Read n, assign sum=0
3. For i=1 to n
   Calculate sum=sum+1/(i*i*i)
4. Print sum
5. Stop.

Sample Output:

Enter n 5

Sun =225

Result:

Thus the program is executed successfully.

EXPNO: 3B   **ILLUSTRATION OF FOR GENERATING FIBONACCI SERIES**

Aim:

To write a C program to generate a Fibonacci series.

Algorithm:

1. Start.
2. Read num
3. Assign f1=0, f2=1.
4. Set loop for 1 and for all i<num value and calculate f3=f1+f2.
5. Display f1
6. Assign f1=f2 and f2=f3
7. Stop.

Sample Output:

Enter how many number 7

0 1 1 2 3 5 8

Result:

Thus the program is executed Successfully.

EXPNO: 3C   **ILLUSTRATION OF WHILE**

CHECKING PALINDROME NUMBER OR NOT

Aim:

To write a C program to check whether given numbers is palindrome number or not.

Algorithm:

1. Start.
2. Read n.
3. Assign rev=0 and temp=h
4. Calculate
   r=n%10
   rev=rev*10+r
   n=n/10
5. Repeat step 4 till n>0
6. Check whether the value of temp and revere equal.
7. If both are equal then print number is palindrome.
8. If both are equal then print number is palindrome.
9. Stop.

Sample Output:

Enter the number: 123

The reversed of number 123 is = 321

It is not palindrome

Enter the number: 151

The reversed of number 151 is = 151

It is not palindrome

Result:

Thus the program is executed Successfully.

EXPNO: 3D   **ILLUSTRATION OF DO- WHILE : PRINTING NUMBER**

Aim:

To Write a C program to print numbers using do – while loop.

Algorithm:

1. Start.
2. Assign digit=0
3. Using do – while loop
   Print the digit and increment i. If digit<=9
4. Stop.

Sample Output:

0 1 2 3 4 5 6 7 8 9

Result:

Thus the program is executed Successfully.

4. Arrays: 1D and 2D, Multi-dimensional arrays, traversal

EXPNO: 4A   **COMPUTING MEAN VALLUE OF N NUMBERS**

Aim:

To Write a C program computing mean value of N numbers

Algorithm:

1. Start.
2. Read number of elements n, and n number in any array.
3. Assign sum=0.
4. Read one element at a time from the array and add it to the sum.
5. Calculate mean=sum/n
6. Print mean
7. Stop.

Sample Output:

Enter number of elements

6

Enter the element

8

3

9

4

5

6

Result:

Thus the program is executed Successfully.

EXPNO: 4B   **ADDITION OF TWO MATRICES**

Aim:

To write a C program to add matrices.

Algorithm:

1. Start.
2. Read the element of matrix A.
3. Read the element of matrix B.
4. Set a loop up to the row.
5. Set a inner loop up to the column.
6. Add the element of A and B column wise and store the result in sum matrix.
7. After execution of two loop, print the values in sum matrix.
8. Stop.

Sample Output:

Enter the elements of matrix A

101

111

010

Enter the elements of matrix B

011

100

111

Sum of two matrix

112

211

121

Result:

Thus the program is executed successfully.

EXPNO: 4C   **MULTIPLICATIONS OF TWO MATRICES**

Aim:

To write a C program to multiply two matrices.

Algorithm:

1. Start.
2. Enter the row and column of the matrix A.
3. Enter the row and column of the matrix B.
4. Enter the element of the matrix A.
5. Enter the element of the matrix B.
6. Print the element of the matrix A in matrix from.
7. Print the element of the matrix B in matrix from.
8. Set a loop up to row.
9. Set an inner loop up to column.
10. Set another inner loop up to column.
11. Multiply the A and B matrix and store and the element in the C matrix.
12. Print the resultant matrix.
13. Stop.

Sample Output:

Enter the row and column value of A matrix 3 3

Enter the row and column value of B matrix 3 3

Enter the element of matrix A1

2

3

4

5

6

7

8

9

Enter the element of matrix B9

8

7

6

5

4

3

2

1

The resultant matrix is

30	24	18
84	69	54
138	114	90

Result:

Thus the program is executed successfully.

EXPNO: 4D   **LINEAR SEARCH**

Aim:

To write a C program to perform linear search.

Algorithm:

1. Start.
2. Read n and n number in array A.
3. Read the key element to be searched
4. Assign sound=0
5. Access the element one at a time and compare it with the key.
6. If both are equal then print key found and increment sound variable.
7. After checking all the element check if found=0 then print key not found.
8. Stop.

Sample Output:

Enter the value of n6

Enter 6 number 1

9

2

8

3

7

Enter key element to be searched3

Key found

Result:

Thus the program is executed successfully.

5. Strings: operations

EXPNO: 5   **SORTING NAMES IN ASCENDING ORDER**

Aim:

    To write a C program to sort the names in alphabetical order using string function.

Algorithm:

1. Start.
2. Enter number of names.
3. Enter the name.
4. Set two loops and compare every two strings.
5. Repeat step 5 until all the strings are compared.
6. Print he names in sorted order.
7. Stop.

Sample Output:

    Enter number of names

    5

    Enter the names

    varsha

    denisha

    jovita

    suthan

    Effie

    The names to sorted order

    Effie

    denisha

    jovita

    suthan

    varsha

Result:

    Thus the program is executed successfully.

6. Functions: call, return, passing parameters by (value, reference), passing arrays to function

EXPNO:6A   **FINDING MAX OF THREE NUMBER USING FUNCTION**

Aim:

To write a C program to find maximum of three number using maximum of three numbers using function.

Algorithm:

1. Start.
2. Read a, b, c.
3. Call max unction with the parameters a, b, c.
4. Print maximum value.
5. Stop.

Max Function:

1. Start.
2. If a>b and a>c then assign max=9.
3. Otherwise if b>c then assign max=b. Otherwise assign max=c
4. Return max.
5. Stop.

Sample Output:

Enter three integer values:

56

28

79

Maximum is:79

Result:

Thus the program is executed successfully.

EXPNO: 6B   **SWAPPING TWO NUBERS USING PASS BY VALUE**

Aim:

To write a C program swapping two numbers using pass by Value.

Algorithm:

1. Start.
2. Call swap function and pass a,b s arguments.
3. Calculate
   X=x+y
   Y=x-y
   X=x-y
4. Display the swapped numbers
5. Stop.

Sample Output:

Before swap values are 10 20

In swap function values are 20 10

After swap values are 10 20

Result:

Thus the program is executed successfully.

EXPNO: 6C   **SWAPPING TWO NUMBERS USING PASS BY REFERENCE**

Aim:

To write a C program swapping two numbers using pass by address/ reference.

Algorithm:

1. Start.
2. Call swap function and pass address of a,b as arguments
3. Calcullate
   *x=*x+*y
   *y=*x-*y
   *x=*x-*y

4. Display x, y values

5. Stop.

Sample Output:

Before swap values are 10 20

In swap function value are 20 10

After swap value are 20 10

Result:

Thus the program is executed successfully.

EXPNO: 6D   **SORTING NUMBERS IN ASCENDING ORDER USING FUNCTION**

Aim:

To Write a C program to sort numbers in ascending order using function.

Algorithm:

1. Start.
2. Read total number of items n
3. Read n numbers in array a.
4. Call function sort with the parameters a,n.
5. Print the sorted array.
6. Stop.

Sort function:

1. Start.
2. Compare elements to sort in ascending order.
3. If the first element is greater than second element, then interchange the elements.
4. Repeat step 2 and 3 to get finally sorted element.
5. Stop.

Sample output:

Enter the numbers to sort: 15 90 25 84 28 10

Sorted Numbers: 10 15 25 28 84 90

Result:

Thus the program is executed successfully.

7. Recursion

EXPNO: 7   **FACTORIAL OF GIVEN NUMBER USING RECURSIVE FUNCTION**

Aim:

   To write a C program to find the factorial of a number using recursion.

Algorithm:

1. Start.
2. Enter the number.
3. Call the factorial recursive function by passing the number.
4. Print the result.
5. Stop.

Recursive Function:

1. Start the function.
2. Assign fact=1
3. If num equals 1 them factorial is 1
4. Calculate fact= x* fact(x-1)
5. Return fact value.

Sample Output:

   Enter the number:

   5

   Factorial of 5=120

Result:

   Thus the program is executed successfully.

8. Pointers: Pointers to functions, Arrays, Strings, Pointers to Pointers, Array of Pointers

EXPNO: 8A **POINTERS TO FUNCTIONS**

CALCULATING AREA OF A TRIANGLE

Aim:

To write a C program to calculate area of a triangle using pointers and functions.

Algorithm:

1. Start.
2. Read base, height
3. Print area
4. Stop.

Sample Output:

Enter the base of the triangle: 10

Enter the height of the triangle: 5

Area of the triangle with base 10.0 and height 5.0=25.00

Result:

Thus the program is executed successfully.

EXPNO: 8B   **POINTERS & ARRAYS FINDING MEAN OF N NUMBERS USING ARRAYS**

Aim:

1. Start.
2. Assign mean=0.0
3. Read numbers of element
4. Read n elements is array
5. Add all the elements in the array and divide the sum by total number of elements to find the mean
6. Print sum and mean
7. Stop

Sample Output:

Enter the number of elements

Enter the number: 1

Enter the number:2

Enter the number:3

Enter the number:4

Enter the number:5

The numbers you entered are;

1 2 3 4 5

The sum is:15

The mean is : 3.00

Result:

Thus the program is executed successfully.

9. Structures: Nested Structures, Pointers to Structures, Arrays of Structures and Unions

EXPNO: 9A   **NESTED STRUCTURES:**

PRINTING EMPLOYEE PERSONAL DETAILS

Aim :

To write C program to calculate the gross salary using structure within Structure

Algorithm:

1. Start
2. Struct employee

   e_code[4]=char

   e_name[20]=char

   e_bp=float

   struct

   e_da=float

   e_hra=float

   allo

   e_pf=float

   end struct

3. Read the employee code,name basic pay, da, hra, pf
4. Print the employee code, name basic pay, hra, pf.
5. Calculate gross
   Salary=emp1.e_bp+emp1.allo.e_da+emp1. allo.e_har+emp1.e_pf.
6. print the gross salary.
7. stop

Sample Output:

Enter the code : 1990

Enter the name : Ramesh

Enter the basic pay : 15000

Enter the dearness allowance:1500

Enter the house rent allowance:1000

Enter the provident fund:800

Code   : 1990

Name   : Ramesh

Basic pay   : 15000.00

Dearnedd allowance   : 1500.00

House rent allowance : 1000.00

Provident fund   : 800.00

Net pay   :18300.00

Result:

Thus the program is executed successfully.

EXPNO: 9B **POINTERS TO STRUCTURS: PRINTING**

STUDENT DETAILS

Aim:

To write a c program to print student details using pointers to structures.

Algorithm:

1. Start.
2. Declare student structure
3. Read student roll number, student name, branch, marks.
4. Print student roll number, student name, branch, marks.
5. Stop.

Sample Output:

Enter Roll no: 1000

Enter Name: Ebisha

Enter Branch: CSE

Enter Marks: 90

Roll Number: 1000

Name: Ebisha

Branch: CSE

Marks: 90

Result:

Thus the program is executed successfully.

EXPNO: 9C   **ARRAY OF STRUCTURES: CALCULATING**

STUDENT MARK DETAILS

Aim:

To Write a C program to calculate the total marks using array of structures.

Algorithm:

1. Start.
2. Declare the structure with members.
3. Initialize the marks of the students.
4. Calculate the subject total by adding student [i].sub 1+ student[i]. sub2+ student[i].sub 3.
5. Print the total marks of the students
6. Stop.

Sample Output:

TOTAL MARKS

Student [1]:193

Student [2]:197

Student [3]:164

Result:

Thus the program is executed successfully.

10. Files: reading and writing, File pointers, file operations, random access, processor directives

EXPNO: 10A   **READING AND WRITING FILE**

Aim:

To write a C program to read and write a file.

Algorithm:

1. Start.
2. Create a file pointer.
3. Read the file name to be opened.
4. Open the with write mode.
5. Write the data.
6. Open the file with read mode
7. Print the data
8. Stop.

Sample Output:

Enter the file name: c.txt

Data in File: Welcome to C programming.

Result:

Thus the program is executed successfully.

EXPNO: 10B   **PROGRAM TO COUNT THE NUMER OF CHARACTERS AND NUMBER OF LINES IN A FILE USING FILE POINTERS**

Aim:

  To write a C Program to count number of characters and number of lines in a

File using file pointer.

Algorithm:

1. Start
2. Create file name
3. Enter the file name
4. Open the file with read mode
5. Till the end of file reached read one character at time

(a) if it is newline character ' /n', then increment non_ of_lines count value by one

(b) if it is a character then increment no_of_characters count value by one.

6. Print on_ of_ lines _and on_ of_ characters values
7. Stop

Sample Output:

Enter the filename: Letter. Txt

In the file Letter. Txt, there is 1 line and 18 characters

Result:

Thus the program is executed successfully.

EXPNO: 10C   **PROGRAM TO COPY ONE FILE TO ANOTHER**

Aim:

To write a C program to one file to another

Algorithm:

1. Start
2. Create file pointers
3. Read two file names
4. Open first file with read mode
5. Open second file with write mode
6. Read first file character by character and write the characters in the second file till end of file reached
7. Close both pointers
8. Stop

Sample Output:

Enter the name of the first filename:

a.txt

Enter the name of the second filename:

b.txt

FILE COPIED

Result:

Thus the program is executed successfully.

EXPNO: 10D   **PROGRAM TO RANDOMLY READ THE NTH**

RECORD OF A FILE

Aim:

To read nth record in file using random access method.

Algorithm:

1. Start
2. Create employee structure with variables
3. Read the file name with 'rb' mode
4. Read the record number to read
5. From the pointed by fp read a record of the specified record starting from the beginning of the file
6. Print the record values
7. Stop

Sample Output:

Enter the rec_ no you want to read: 06

Employee CODE: 06

Name: Tanya

HRA, DA and TA: 20000 10000 3000

Result:

Thus the program is executed successfully.

EXPNO: 10E   **PROGRAM TO COMPUTE AREA OF CIRCLE USING PREPROCESSOR**
DIRECTIVES

Aim:

To write a C program to compute of a circle using pre-processor directives.

Algorithm:

1. Start
2. Define pi as 3.1415
3. Define circle_ area (or) as pi*r*r
4. Read radius
5. Call circle area function radius calculate area of circle.
6. Print area
7. Stop.

Sample Output:

Enter the radius: 6

Area= 113.094002

Result:

Thus the program is executed successfully.