LAB MANUAL

Subject: Programming in C Lab
Code: ESC-CSE-103

Department of Computer Science Engineering
RPS College of Engineering and Technology
Mahendergarh, Haryana - 123029
<table>
<thead>
<tr>
<th>PROGRAM NO.</th>
<th>NAME OF PROGRAM</th>
</tr>
</thead>
<tbody>
<tr>
<td>1</td>
<td>Write a C program to find the sum of individual digits of positive integer.</td>
</tr>
<tr>
<td>2</td>
<td>To generate the first n terms of the Fibonacci sequence.</td>
</tr>
<tr>
<td>3</td>
<td>Write a C program to generate all prime numbers between 1 and n.</td>
</tr>
<tr>
<td>4</td>
<td>Write a C program which takes two integer operands and one operator from the user, performs the operation and then prints the result. (Consider the operators +, -, *, /, % and use Switch Statement.)</td>
</tr>
<tr>
<td>5</td>
<td>Write a C program to find the factorial of a given integer using recursive function.</td>
</tr>
<tr>
<td>6</td>
<td>Write a C program to determine if the given string is a palindrome or not.</td>
</tr>
<tr>
<td>7</td>
<td>To write a C program to perform swapping using function.</td>
</tr>
<tr>
<td>8</td>
<td>To write a C program to get the largest element of an array.</td>
</tr>
<tr>
<td>9</td>
<td>Write a C program to store student information in structure and display it.</td>
</tr>
<tr>
<td>10</td>
<td>Write a program to create a file called emp.rec and store information about a person, in terms of his name, age and salary.</td>
</tr>
<tr>
<td>11</td>
<td>Write a program to illustrate how a file stored on the disk is read.</td>
</tr>
<tr>
<td>12</td>
<td>To write a C program to find sum of n elements entered by user. To perform this program, allocate memory dynamically using calloc() function.</td>
</tr>
</tbody>
</table>
LAB REQUIREMENTS

HARDWARE REQUIREMENT :

- Processor : Intel(R) corei3 – 8100CPU 3.60GHz
- RAM : 8GB
- HDD : 250GB SSD
- Operating System : Windows10

SOFTWARE REQUIREMENT

- Turbo C
- VS Code

INTRODUCTION ABOUT LAB:

- Steps involved in program development:-
  1. Writing and editing the program.
  2. Linking the program with the required library modules.
  3. Compiling the program.
  4. Executing the program.
ALGORITHM:

It is a method of representing the step by step process for solving a problem. Each step is called an instruction.
Characteristics of algorithm are:
- Finiteness: It terminates with finite number of steps.
- Definiteness: Each step of algorithm is exactly defined.
- Effectiveness: All the operations used in the algorithm can be performed exactly in a fixed duration of time.
- Input: An algorithm must have an input before the execution of program begins.
- Output: An algorithm has one or more outputs after the execution of the program.

ABOUT C LANGUAGE

C is a programming language developed by Dennis Ritchie at AT&T’s BELL Laboratory of USA in 1972. Because of its reliability, C is very popular. C is highly portable & it is well suited for structured programming. C program consists of collection of functions.

HISTORY OF C

The milestones in C's development as a language are listed below: 1.

1. BCPL - a user friendly OS providing powerful development tools developed from BCPL c. 1967. Assembler was tedious, long and error prone.
2. A new language “B” was attempted c. 1970 by Ken Thompson at Bell Labs.
3. UNIX was developed c. 1970 using “B” - DEC PDP-7 Assembly Language.
4. A totally new language “C”, a successor to “B” was developed c. 1971. 5. By 1973, UNIX OS was almost totally written in “C”.
AIM 1: Write a C program to find the sum of individual digits of positive integer.

Description: Summation of digits of a number
Ex: 1234
Summation =1+2+3+4=10

ALGORITHM:
Step 1: Start
Step 2: Read n
Step 3: Initialize sum ← 0
Step 4: while(n!=0)
    Begin
Step 5: r←n%10
Step 6: sum←sum+r
Step 7: n←n/10
    End
Step 8: Print “sum”
Step 9: Stop

PROGRAM:
#include<stdio.h>
#include<conio.h>
void main()
{
    int n,r,sum=0;
    clrscr();
    printf("ENTER A POSITIVE INTEGER \n");
    scanf("%d",&n);
    while(n!=0) { r=n%10;
    sum=sum+r; n=n/10;
    }
    printf("THE SUM OF INDIVIDUAL DIGITS OF A POSITIVE INTEGER IS..%.d",sum);
    getch();
}

OUTPUT:

ENTER A POSITIVE INTEGER

5 3 2 1

THE SUM OF INDIVIDUAL DIGITS OF A POSITIVE INTEGER IS..11
AIM 2: To generate the first \( n \) terms of the Fibonacci sequence.

**Description:** Initial Fibonacci numbers are 0 and 1. Next number can be generated by adding two numbers. So 0+1=1. Therefore next number can be generated by adding two previous. So Fibonacci series is 0 1 1 2 3 5 ……

**Algorithm:**
Step 1: Start
Step 2: Read \( n \)
Step 3: Initialize \( f_0 \leftarrow 0, f_1 \leftarrow 1, f \leftarrow 0 \)
Step 4: \( i=0 \)
Step 5: while \((i<=n)\) do as follows
   - \( f=f_0+f_1; \)
   - \( f_0=f_1; \)
   - \( f_1=f; \)
   - \( i=i+1; \)
   If not goto step 6
Step 6: Stop

**Program:**
```c
#include<stdio.h>
#include<conio.h>
void main()
{
    int f0,f1,f,n,i;
    clrscr();
    printf("ENTER THE VALUE FOR n \n");
    scanf("%d",&n);
    f0=0;
    f1=1;
    printf("FIBONACCI SEQUENCE FOR THE FIRST %d TERMS:\n",n);
    i=0;
    while(i<n)
    {
        printf("%d\n",f0);
        f=f0+f1;
        f0=f1;
        f1=f;
        i=i+1;
    }
}
```

**Output**
ENTER THE VALUE FOR \( n \)
10
FIBONACCI SEQUENCE FOR THE FIRST 10 TERMS:
0 1 1 2 3 5 8 13 21 34
AIM 3: Write a C program to generate all prime numbers between 1 and n.

Description:
Prime number is a number which is exactly divisible by one and itself only Ex: 2, 3,5,7,........;

Algorithm:
Step 1: start
2: read n Step
3: initialize i=1,c=0
Step 4: if i<=n goto step 5 If not goto step 10
Step 5: initialize j=1
Step 6: if j<=i do the following. If no goto step 7
   i) if i%j==0 increment c
   ii) increment j
   iii) goto Step 6
Step 7: if c== 2 print i
Step 8: increment i
Step 9: goto step 4
Step 10: stop.

Program: #include<stdio.h>
#include<conio.h>
void main()
{
    int n,i,fact,j;
clrscr();
printf("enter the number:");
scanf("%d", &n);

for (i = 1; i <= n; i++)
{
    fact = 0; // THIS LOOP WILL CHECK A NO TO BE PRIME NO. OR NOT.
    for (j = 1; j <= i; j++)
    {
        if (i % j == 0)
            fact++;
    }
    if (fact == 2)
        printf("n %d", i);
}

getch();

**Output:**

Enter the number : 5

2 3 5
AIM 4: Write a C program which takes two integer operands and one operator from the user, performs the operation and then prints the result. (Consider the operators +, -, *, /, % and use Switch Statement.)

Algorithm:

Step 1: Read a, b
Step 2: Print “Menu Options”
Step 3: do Begin
Step 4: Read ch
Step 5: switch(ch)

Begin
Step 6:
  case 1:
  Begin
  Calculate c = a + b
  Print “c”
  break;
  End
  case 2:
  Begin
  Calculate c = a - b
  Print “c”
  break;
  End
  case 3:
  Begin
  Calculate c = a * b
  End
Print “c”
break;
End

case 4:
Begin
Calculate c = a/b
Print “c”
break;
End

case 5:
Begin
Calculate c = a%b
Print “c”
break;
End
default: Print “Invalid choice”
End.

Program:

#include<stdio.h>
#include<conio.h>

void main()
{
int a,b,c,ch;
clrsr();
printf("ENTER TWO VALUES FOR a & b\n");
}
scanf("%d %d",&a,&b);
while(1)
{
    printf("MENU OPTIONS \n");
    printf("************\n");
    printf("1.Addition\n");
    printf("2.Subtraction\n");
    printf("3.Multiplication\n");
    printf("4.Division\n");
    printf("5.Modulus\n");
    printf("6.Exit\n");
    printf("\n");
    printf("ENTER UR CHOICE\n");
    scanf("%d",&ch); switch(ch)
{
    case 1:
        c=a+b;
        printf("The addition of %d and %d is..%d\n",a,b,c);
        break;
    case 2:
        c=a-b;
        printf("The subtraction of %d and %d is..%d\n",a,b,c);
        break;
    case 3:
        c=a*b;
        printf("The multiplication of %d and %d is..%d\n",a,b,c);
        break;
    case 4:
        c=a/b;
        printf("The division of %d and %d is..%d\n",a,b,c);
        break;
    case 5:
        c=a%b;
        printf("The modulus of %d and %d is..%d\n",a,b,c);
        break;
    case 6:
        printf("EXITING\n");
        break;
    default:
        printf("Invalid choice\n");
}
}
printf("The multiplication of %d and %d is..%d\n",a,b,c);
break;
case 4: c=a/b;
printf("The division of %d and %d is..%d\n",a,b,c);
break;
case 5: c=a%b;
printf("The modulus of %d and %d is..%d\n",a,b,c);
break;
case 6:exit(0); default:printf("INVALID CHOICE\n");
}
}
getch();

INPUT:

ENTER TWO VALUES FOR a & b: 20  16

ENTER UR CHOICE

1

The addition of 20 and 16 is..36
AIM 5: Write a C program to find the factorial of a given integer using recursive function.

ALGORITHM:
main program

Step 1:  start
Step 2: read n
Step 3: call sub program as f=fact(n)
Step 4: print f value
Step 5: stop

Sub program:

Step 1: initialize the f
Step 2: if n = 0 or n == 1
return 1 to main program if not goto STEP 3
Step 3: return n*fact(n-1) to main program

PROGRAM:

#include<stdio.h>
#include<conio.h>

int fact(int);

void main()
{
    int n,res;
    clrscr();
    printf("ENTER A NUMBER:\n");
    scanf("%d",&n); res=fact(n);
    printf("THE FACTORIAL OF A GIVEN NUMBER IS..%d",res);
    getch();
}
int fact(int n) 
{
    int r;
    if(n==0)
      return(1);
    else
    {
      r=n*fact(n-1);
      return(r);
    }
}

OUTPUT:

ENTER A VALUE FOR n

5

THE FACTORIAL OF A GIVEN NUMBER IS..120
AIM 6: Write a C Program to Determine If the Given String Is a Palindrome Or Not

Description:
Palindrome means string on reversal should be same as original Ex: madam on reversal is also madam

Algorithm:
Step 1: start
Step 2: read string A
Step 3: copy string A into B
Step 4: reverse string B
Step 5: compare A & B If A equals B to goto step 6 Else goto step 7
Step 6: print given string A is palindrom
Step 7: print given string is not palindroma
Step 8: stop

Program:
#include <stdio.h>
#include <string.h>

void main()
{
char string[25], reverse_string[25] = {0};
int i, length = 0, flag = 0;
printf("Enter a string \n");
gets(string);
gets(string);
for (i = 0; string[i] != \0; i++)
{
    length++;
}
Enter a string  madam

The length of the string 'madam' = 5

madam is a palindrome
AIM 7: To write a C program to perform swapping using function.

Algorithm:

1. Start the program
2. Declare and get the two integer variables a and b.
3. call the swap() function
   3.1 In swap definition use the temporary variable and assign temp = a
   3.2 a = b
   3.3 b = temp
4. Print the a and b value.
5. Display the result
6. Stop the program.

Program:

```c
#include<stdio.h>
#include<conio.h>

void swap(int a, int b);

void main()
{
    int a, b, r;
    clrscr();
    printf("Enter value for a & b: ");
    scanf("%d%d", &a, &b);
    swap(a, b);
    getch();
}

void swap(int a, int b)
```

{  
    int temp;  
    temp=a;  
    a=b;  
    b=temp;  
    printf("after swapping the value for a & b is : %d %d",a,b);  
}

**Output:**

enter value for a&b: 4 5  

after swapping the value for a &b : 5 4
AIM 8: To write a C program to get the largest element of an array.

Algorithm:

1. Start the program
2. Read the number of array elements as size.
3. Read array elements array \( i = 0, 1, 2, 3, \ldots n-1 \)
4. Assume first element array[0] to be maximum
5. Compare each element array[i] with maximum
6. If (maximum > array[i]) then maximum = array[i]
7. Display the result.
8. Stop the program.

Program:

```c
int main()
{
    int array[100], maximum, size, c, location = 1;
    printf("Enter the number of elements in array\n");
    scanf("%d", &size);
    printf("Enter %d integers\n", size);
    for (c = 0; c < size; c++)
        scanf("%d", &array[c]);
    maximum = array[0];
    for (c = 1; c < size; c++)
    {
        if (array[c] > maximum)
            maximum = array[c];
    }
    // Display the result
    printf("The largest element is: %d\n", maximum);
    return 0;
}
```

location = c+1;
}
}

printf("Maximum element is present at location %d and it's value is %d.\n", location, maximum);
return 0;
}

Output: Enter the number of elements in array
5
Enter 5 integers
4 5 6 8 2
Maximum element present at location 4 and its value is 8
AIM 9: Write a C program to Store Student Information in Structure and Display it.

Algorithm:
1. Start the program
2. Declare the variables in the structure data type.
3. Read the values of the variables in the structure.
4. Display the result
5. Stop the program

Program:

```
#include <stdio.h>
struct Student
{
 char name[50];
 int roll; float marks;
} s[10];
int main()
{
 int i;
 printf("Enter information of students:\n");
 for(i=0; i<10; ++i)
 {
 s[i].roll = i+1;
 printf("\nFor roll number%d,\n",s[i].roll);
 printf("Enter name: ");
 scanf("%s",s[i].name);
 printf("Enter marks: ");
 scanf("%f",&s[i].marks);
 printf("\n");
 }
 printf("Displaying Information:\n\n");
 for(i=0; i<10; ++i)
 {
 printf("\nRoll number: %d\n",i+1);
 printf("Name: ");
 puts(s[i].name);
 printf("Marks: %.1f\n",s[i].marks);
 printf("\n");
 }
 return 0;
}
```
Output:

Enter information of students:
For roll number1,
Enter name: Tom
Enter marks: 98
For roll number2,
Enter name: Jerry
Enter marks: 89
Displaying Information:
Roll number: 1
Name: Tom
Marks: 98
For roll number2,
Enter name: Jerry
Enter marks: 89
AIM 10: Write a program to create a file called emp.rec and store information about a person, in terms of his name, age and salary.

```c
#include <stdio.h>
void main()
{
    FILE *fptr;
    char name[20];
    int age;
    float salary;
    fptr = fopen("emp.rec", "w");
    if (fptr == NULL)
    { 
        printf("File does not exists \n");
        return;
    }
    printf("Enter the name \n");
    scanf("%s", name);
    fprintf(fptr, "Name    = %s
", name);
    printf("Enter the age\n");
    scanf("%d", &age);
    fprintf(fptr, "Age     = %d
", age);
    printf("Enter the salary\n");
    scanf("%f", &salary);
    fprintf(fptr, "Salary  = %.2f
", salary);
    fclose(fptr);
}
```

**Output:** On file

Enter the name
Shiv
Name=Shiv
Enter the age
25
Age=25
Enter the Salary:
200000
Salary=200000
AIM 11: Write a program to illustrate how a file stored on the disk is read.

Program:

#include <stdio.h>
#include <stdlib.h>
void main()
{
    FILE *fptr;
    char filename[15];
    char ch;
    printf("Enter the filename to be opened \n");
    scanf("%s", filename); /* open the file for reading */
    fptr = fopen(filename, "r");
    if (fptr == NULL)
    {
        printf("Cannot open file \n");
        exit(0);
    }
    ch = fgetc(fptr);
    while (ch != EOF)
    {
        printf ("%c", ch);
        ch = fgetc(fptr);
    }
    fclose(fptr);
}

Output:

Enter the file name to be opened:

File1.txt

C programming
AIM 12: To write a C program to find sum of n elements entered by user. To perform this program, allocate memory dynamically using calloc() function.

Program:

```c
#include <stdio.h>
#include <stdlib.h>

int main()
{
    int i, n;
    int *a;

    printf("Number of elements to be entered:");
    scanf("%d",&n);
    a = (int*)calloc(n, sizeof(int));

    printf("Enter %d numbers:
",n);
    for( i=0 ; i < n ; i++ )
    {
        scanf("%d",a[i]);
    }

    printf("The numbers entered are: ");
    for( i=0 ; i < n ; i++ )
    {
        printf("%d ",a[i]);
    }

    return(0);
}
```

Output:
Number of elements to be entered: 3

Enter 3 numbers: 22 55 14

The numbers entered are: 22 55 14