Ramakrishna Mission Vivekananda Centenary College Rahara, Kolkata-118

LAB ASSIGNMENT COPY

Reg No – A01-1112-117-032-2023

Roll – 737

Paper Code - UGCMSMC01

Paper Name - Major Course 1

Semester – 1

Year - 2023-24

Programming Fundamentals using C Lab

Dept-CS   Sem-1st

C Program – Basic	page	Remarks
1. Write a c program to print the sum of two numbers where the numbers will be given by the user.		
2. Write a c program to swap two numbers using third variable or without third variable		
3. Write a c program to find the size of int, float, double, and char		
4. Write a c program to print the ASCII value of a character		
Control Flow and Loop		
5. Write a c program to check whether a given number is even or odd.		
6. Write a c program to find largest number among three numbers using ternary operator and smallest among them using nested if-else.		
7. Write a c program to check leap year of a given year.		
8. Write a c program to make a simple calculator.		
9. Write a c program to check whether a given number is prime or not		
10. Write a c program to calculate the sum of n natural numbers. The inputs will be provided by the user.		
11. Write a c program to find factorial of a given number.		
12. Write a c program to reverse a given number.		
13. Write a c program to print the day of 1st January of any year inputted by the user, considering the first January of 1900 is Monday.		
14. Write a c program to display Armstrong numbers between 1 to 1000		
15. Write a C program to check a given number is palindrome or not.		
16. Write a C program to print the following pattern for n number of lines where n will be given by the user: * * * * * * * * * *		

17. Write a C program to print the following pattern for n number of lines where n will be given by the user:
   *
  * *
 * * *
* * * *		

Roll-737
Sem-1
Year-2023

18.	Write a C program to print the following pattern for n number of lines where n will be given by the user: 2 4 6 12 14 16 32 34 36 38		
19.	Write a C program to print the following pattern for n number of lines where n will be given by the user: 1 23 456 78910		
	Functions and pointers		
20.	Write a C program to find out the area and perimeter of a rectangle using two different functions.		

21.	Write a C Program to find the cube and square root of a number using two different functions without using the standard math library functions.		
22.	Write a C program to calculate simple interest using function. The amount, rate of interest and number of year's term will be given by the user.		
23.	Write a C program to swap two numbers using call by reference		
	Arrays & String		
24.	Write a c Program to search a particular element in an Array.		
25.	Write a c Program to Find the Maximum and Minimum in an Array.		
26.	Write a c program to get n number of integers from the user and store them in an array. Now, do the addition and the multiplication of the entire array element and show the results. 0<n<100		
27.	Write a c program to multiply two matrices		
28.	Write a c program to create a square matrix of size m dynamically. Value of m will be provided by the user. Calculate the sum of all the left diagonal elements of the sum of all the right diagonal elements of the matrix. Print the two sums.		
29.	Write a c program to find the length of a string without strlen() function		
30.	Write a c program to compare two strings lexicographically		
31.	Write a c program to reverse a string without the library function.		
	Structures and Unions		
32.	Write a c program to create your data type to store data of 10 books and print the data in proper format.		
33.	Write a c program to store student records as structures and sort them by age or roll number.		
	File and Command line Arguments		
34.	Write a c program to copy one file into another file where the file names will be provided through command line. The input will be like: mycopy<source file><destination file>		
35.	Write a c program to read and write stored student record into a file.		

Roll-737
Sem-1
Year-2023

1. Write a c program to print the sum of two numbers where the numbers will be given by the user.

   CODE–
   ```c
   #include <stdio.h>
   int main() {
     int num1, num2, sum;
     printf("Enter two numbers: ");
     scanf("%d %d", &num1, &num2);
     sum = num1 + num2;
     printf("The sum of the two numbers is %d\n", sum);
     return 0;
   }
   ```
   OUTPUT–
   Enter two numbers: 5 13
   The sum of the two numbers is 18

2. Write a c program to swap two numbers using third variable or without third variable

   CODE_1–
   ```c
   #include <stdio.h>
   int main() {
     int num1, num2, temp;
     printf("Enter two numbers: ");
     scanf("%d %d", &num1, &num2);
     temp = num1;
     num1 = num2;
     num2 = temp;
     printf("After swapping, the first number is %d and the second number is %d\n", num1, num2);
     return 0;
   }
   ```
   OUTPUT–
   Enter two numbers: 13 19
   After swapping, the first number is 19 and the second number is 13

   CODE_2–
   ```c
   #include <stdio.h>
   int main() {
     int num1, num2;
     printf("Enter two numbers: ");
     scanf("%d %d", &num1, &num2);
     num1 = num1 + num2;
     num2 = num1 – num2;
     num1 = num1 – num2;
   ```

Roll-737
Sem-1
Year-2023

```
  printf("After swapping, the first number is %d and the second number is %d\n",
num1, num2);
  return 0;
}
```
OUTPUT–
Enter two numbers: 36 98
After swapping, the first number is 98 and the second number is 36

3.   Write a c program to find the size of int, float, double, and char

CODE–
```
#include <stdio.h>
int main() {
  int int_size = sizeof(int);
  float float_size = sizeof(float);
  double double_size = sizeof(double);
  char char_size = sizeof(char);
  printf("The size of int is %lu bytes\n", int_size);
  printf("The size of float is %lu bytes\n", float_size);
  printf("The size of double is %lu bytes\n", double_size);
  printf("The size of char is %lu bytes\n", char_size);
  return 0;
}
```
OUTPUT–
The size of int is 4 bytes
The size of float is 0 bytes
The size of double is 0 bytes
The size of char is 1 bytes

4.   Write a c program to print the ASCII value of a character

CODE–
```
#include <stdio.h>
int main() {
  char c;
  printf("Enter a character: ");
  scanf("%c", &c);
  printf("The ASCII value of %c is %d\n", c, c);
  return 0;
}
```
OUTPUT–
Enter a character: r
The ASCII value of r is 114

5.   Write a c program to check whether a given number is even or odd.

```
CODE-
#include <stdio.h>
int main() {
  int num;
  printf("Enter a number: ");
  scanf("%d", &num);
  if (num % 2 == 0) {
    printf("%d is an even number.\n", num);
  } else {
    printf("%d is an odd number.\n", num);
  }
  return 0;
}
OUTPUT-
Enter a number: 25
25 is an odd number.
```

6.   Write a c program to find largest number among three numbers using ternary operator and smallest among them using nested if-else.

```
CODE_1-
#include<stdio.h>
int main(){
   int a , b , c , g;
   printf("Value of a , b , c = ");
   scanf("%d%d%d",&a,&b,&c);
   g=(a>b)?((a>c)?a:c):((b>c)?b:c);
   printf("Largest = %d",g);
}
OUTPUT-
Value of a , b , c = 23 29 37
Largest = 37
```

```
CODE_2-
#include <stdio.h>
int main() {
  int num1, num2, num3, smallest;
  printf("Enter three numbers: ");
  scanf("%d %d %d", &num1, &num2, &num3);
  if (num1 < num2 && num1 < num3) {
    smallest = num1;
  } else if (num2 < num1 && num2 < num3) {
    smallest = num2;
  } else {
    smallest = num3;
  }
```

Roll-737
Sem-1
Year-2023

```
  printf("The smallest number is %d\n", smallest);
  return 0;
}
```
OUTPUT–
Enter three numbers: 39 38 19
The smallest number is 19

7.   Write a c program to check leap year of a given year.

CODE–
```c
#include <stdio.h>
int main() {
  int year;
  printf("Enter a year: ");
  scanf("%d", &year);
  if ((year % 4 == 0 && year % 100 != 0) || year % 400 == 0) {
    printf("%d is a leap year.\n", year);
  } else {
    printf("%d is not a leap year.\n", year);
  }
  return 0;
}
```
OUTPUT–
Enter a year: 2023
2023 is not a leap year.

8.   Write a c program to make a simple calculator.

CODE–
```c
#include <stdio.h>
int main() {
  char operator;
  float num1, num2, result;
  printf("Enter an operator (+, -, *, /): ");
  scanf("%c", &operator);
  printf("Enter two numbers: ");
  scanf("%f %f", &num1, &num2);
  switch (operator) {
    case '+':
      result = num1 + num2;
      break;
    case '-':
      result = num1 - num2;
      break;
    case '*':
      result = num1 * num2;
      break;
    case '/':
```

```
      result = num1 / num2;
      break;
    default:
      printf("Invalid operator.\n");
      break;
  }

  printf("The result is: %f\n", result);
  return 0;
}
```
OUTPUT–
Enter an operator (+, –, *, /): /
Enter two numbers: 39 6
The result is: 6.500000

9. Write a c program to check whether a given number is prime or not

CODE–
```
#include<stdio.h>
int main(){
   int n,i,m=0,pn=0;
   printf("Enter a Number=");
   scanf("%d",&n);
   m=n/2;
   if(n>1){
      for (i = 2; i < m; i++)
      {
         if (n%i==0)
         {
            printf("Not a Prime Number");
            pn=1;
            break;
         }

      }
      if (pn==0)
      {
         printf("Prime Number");
      }
   }
   else
   {
      printf("Not a Prime Number");
   }
   return 0;
}
```
OUTPUT–
Enter a Number=11

Roll-737
Sem-1
Year-2023

Prime Number

10. Write a c program to calculate the sum of n natural numbers. The inputs will be provided by the user.

CODE–
```c
#include <stdio.h>
int main() {
  int n, i, sum;
  printf("Enter a number: ");
  scanf("%d", &n);
  sum = 0;
  for (i = 1; i <= n; i++) {
    sum += i;
  }
  printf("The sum of the first %d natural numbers is %d.\n", n, sum);
  return 0;
}
```
OUTPUT–
Enter a number: 18
The sum of the first 18 natural numbers is 171.

11. Write a c program to find factorial of a given number.

CODE–
```c
#include <stdio.h>
int main() {
  int num, fact;
  printf("Enter a number: ");
  scanf("%d", &num);
  fact = 1;
  for (int i = 1; i <= num; i++) {
    fact *= i;
  }
  printf("The factorial of %d is %d.\n", num, fact);
  return 0;
}
```
OUTPUT–
Enter a number: 5
The factorial of 5 is 120.

12. Write a c program to reverse a given number.

CODE–
```c
#include <stdio.h>
int main() {
  int num, rev, rem;
  printf("Enter a number: ");
```

```
     scanf("%d", &num);
     rev = 0;
     while (num != 0) {
       rem = num % 10;
       rev = rev * 10 + rem;
       num /= 10;
     }
     printf("The reversed number is %d.\n", rev);
     return 0;
}
```

OUTPUT-
Enter a number: 2369
The reversed number is 9632.

13. Write a c program to print the day of 1st January of any year inputted by the user, considering the first January of 1900 is Monday.

```
CODE-
#include<stdio.h>
int main(){
int year, year_gap, leap=0, total_day, ref_2=400, ref=1900, day;
printf("enter a year greater than 1900 = ");
scanf("%d",&year);
if(year>ref)
{
year_gap=year-ref;
for(;ref<year;ref++)
{
if(ref%400==0 || ref%100!=0 && ref%4==0)
{
leap++;
}
}
total_day=((year_gap-leap)*365)+(leap*366);
day=total_day%7;
switch(day)
{
case 0:
printf("Monday");
break;
case 1:
printf("Tuesday");
break;
case 2:
printf("Wednesday");
break;
case 3:
printf("Thursday");
```

```
break;
case 4:
printf("Friday");
break;
case 5:
printf("Saturday");
break;
case 6:
printf("Sunday");
break;
}
}
return 0;
}
```

OUTPUT–
enter a year greater than 1900 = 2023
Sunday

14.   Write a c program to display Armstrong numbers between 1 to 1000

CODE–
```
#include <stdio.h>
int main() {
  int num, sum;
  for (int i = 1; i <= 1000; i++) {
    num = i;
    sum = 0;
    while (num > 0) {
      int rem = num % 10;
      sum += rem * rem * rem;
      num /= 10;
    }
    if (sum == i) {
      printf("%d is an Armstrong number\n", i);
    }
  }
  return 0;
}
```

OUTPUT–
1 is an Armstrong number
153 is an Armstrong number
370 is an Armstrong number
371 is an Armstrong number
407 is an Armstrong number

15. Write a C program to check a given number is palindrome or not.

```
CODE-
#include <stdio.h>
int main() {
  int num, rev=0, rem, ex;
  printf("Enter a number: ");
  scanf("%d", &num);
  ex = num;
  while (num != 0) {
    rem = num % 10;
    rev = rev * 10 + rem;
    num /= 10;
  }
  if (ex == rev) {
    printf("%d is a palindrome number.\n", ex);
  } else {
    printf("%d is not a palindrome number.\n", ex);
  }
  return 0;
}
OUTPUT-
Enter a number: 121
121 is a palindrome number.
```

16. Write a C program to print the following pattern for n number of lines where n will be given by the user:

```
*

* *

* * *

* * * *
```

```
CODE-
#include <stdio.h>
int main() {
  int n;
  printf("Enter the number of lines: ");
  scanf("%d", &n);
  for (int i = 1; i <= n; i++) {
    for (int j = 1; j <= i; j++) {
      printf("*");
    }
    printf("\n");
  }
```

```
  return 0;
}
OUTPUT–
Enter the number of lines: 4
*
**
***
****
```

17.   Write a C program to print the following pattern for n number of lines where n will be given by the user:

```
   *

  * *

 * * *

* * * *
```

```
CODE–
#include <stdio.h>
int main() {
  int n;
  printf("Enter the number of lines: ");
  scanf("%d", &n);
  for (int i = 1; i <= n; i++) {
    for (int j = 1; j <= n – i; j++) {
      printf(" ");
    }
    for (int j = 1; j <= i; j++) {
      printf("*");
    }
    printf("\n");
  }
  return 0;
}
```

```
OUTPUT–
Enter the number of lines: 4
   *
  **
 ***
****
```

18. Write a C program to print the following pattern for n number of lines where n will be given by the user:

2

4 6

12 14 16

32 34 36 38

CODE-
```c
#include<stdio.h>
int main()
{
    int n,i,j,a=2;
    printf("Enter a Number:");
    scanf("%d",&n);
    for(i=1;i<=n;i++)
    {
            for(j=1;j<=i;j++)
            {
                    printf("%d",a);
                    printf(" ");
            if(i!=j)
            {
               a=a+2;
                 }
         }
        a=a*2;
            printf("\n");
    }
    return 0;
}
```
OUTPUT-
Enter a Number:4
2
4 6
12 14 16
32 34 36 38

19. Write a C program to print the following pattern for n number of lines where n will be given by the user:

1

2 3

```
4 5 6

7 8 9 10

CODE-
#include <stdio.h>
int main() {
  int n;
  printf("Enter the number of lines: ");
  scanf("%d", &n);
  int num = 1;
  for (int i = 1; i <= n; i++) {
    for (int j = 1; j <= i; j++) {
      printf("%d ", num);
      num++;
    }
    printf("\n");
  }
  return 0;
}
OUTPUT-
Enter the number of lines: 4
1
2 3
4 5 6
7 8 9 10
```

20. Write a C program to find out the area and perimeter of a rectangle using two different functions.

```
CODE-
#include <stdio.h>
int area(int length, int breadth) {
  return length * breadth;
}
int perimeter(int length, int breadth) {
  return 2 * (length + breadth);
}
int main() {
  int length, breadth;
  printf("Enter the length of the rectangle: ");
  scanf("%d", &length);
  printf("Enter the breadth of the rectangle: ");
  scanf("%d", &breadth);
  int area_of_rectangle = area(length, breadth);
  int perimeter_of_rectangle = perimeter(length, breadth);
  printf("The area of the rectangle is: %d\n", area_of_rectangle);
  printf("The perimeter of the rectangle is: %d\n", perimeter_of_rectangle);
```

```
  return 0;
}
OUTPUT–
Enter the length of the rectangle: 5
Enter the breadth of the rectangle: 6
The area of the rectangle is: 30
The perimeter of the rectangle is: 22
```

21.   Write a C Program to find the cube and square root of a number using two different functions without using the standard math library functions.

```
CODE–
#include <stdio.h>
int cube(int num) {
  int cube = num * num * num;
  return cube;
}
int squareRoot(int num) {
  int sqrt = 0;
  for (int i = 1; i * i <= num; i++) {
    sqrt = i;
  }
  return sqrt;
}
int main() {
  int number;
  printf("Enter a number: ");
  scanf("%d", &number);
  int cubeResult = cube(number);
  printf("Cube of %d is: %d\n", number, cubeResult);
  int sqrtResult = squareRoot(number);
  printf("Square root of %d is: %d\n", number, sqrtResult);
  return 0;
}
OUTPUT–
Enter a number: 25
Cube of 25 is: 15625
Square root of 25 is: 5
```

22.   Write a C program to calculate simple interest using function. The amount, rate of interest and number of year's term will be given by the user.

```
CODE–
#include <stdio.h>
float calculate_simple_interest(float principal, float rate, int years) {
  return (principal * rate * years) / 100;
}
int main() {
```

```
float principal, rate, interest;
int years;
printf("Enter principal amount: ");
scanf("%f", &principal);
printf("Enter rate of interest: ");
scanf("%f", &rate);
printf("Enter number of years: ");
scanf("%d", &years);
interest = calculate_simple_interest(principal, rate, years);
printf("Simple interest: %.2f\n", interest);
return 0;
}
```
OUTPUT–
Enter principal amount: 25000
Enter rate of interest: 5
Enter number of years: 5
Simple interest: 6250.00

23.   Write a C program to swap two numbers using call by reference

CODE–
```
#include<stdio.h>
void swap(int *x , int *y)
    {
            *x=*x+*y;
            *y=*x-*y;
            *x=*x-*y;
    }
int main()
    {
            int a , b ;
            printf("Enter the value of a & b = ");
            scanf("%d%d",&a,&b);
            swap(&a,&b);
            printf("Current value of a = %d\n",a);
            printf("Current value of b = %d\n",b);
    }
```
OUTPUT–
Enter the value of a & b = 5 9
Current value of a = 9
Current value of b = 5

24.   Write a c Program to search a particular element in an Array.

CODE–
```
#include <stdio.h>
int main() {
 int array[100], element, size, found = 0;
```

```c
int i ;
printf("Enter the size of the array: ");
scanf("%d", &size);
printf("Enter the elements of the array: ");
for (i = 0; i < size; i++) {
  scanf("%d", &array[i]);
}
printf("Enter the element to search: ");
scanf("%d", &element);
for (i = 0; i < size; i++) {
  if (array[i] == element) {
    found = 1;
    break;
  }
}
if (found == 1) {
  printf("The element is present in the array at index %d.\n", i);
  printf("The element is %d.\n", array[i]);
} else {
  printf("The element is not present in the array.\n");
}
return 0;
}
```

OUTPUT–
Enter the size of the array: 5
Enter the elements of the array: 1 2 3 4 5
Enter the element to search: 3
The element is present in the array at index 2.
The element is 3.

25. Write a c Program to Find the Maximum and Minimum in an Array.

CODE–
```c
#include <stdio.h>
int main() {
  int array[100], size, max, min;
  printf("Enter the size of the array: ");
  scanf("%d", &size);
  printf("Enter the elements of the array: ");
  for (int i = 0; i < size; i++) {
    scanf("%d", &array[i]);
  }
  max = array[0];
  min = array[0];
  for (int i = 1; i < size; i++) {
    if (array[i] > max) {
```

```
      max = array[i];
    }

    if (array[i] < min) {
      min = array[i];
    }
  }
  printf("The maximum element in the array is %d.\n", max);
  printf("The minimum element in the array is %d.\n", min);
  return 0;
}
```

OUTPUT–
Enter the size of the array: 5
Enter the elements of the array: 1 2 3 4 5
The maximum element in the array is 5.
The minimum element in the array is 1.

26. Write a c program to get n number of integers from the user and store them in an array. Now, do the addition and the multiplication of the entire array element and show the results. 0<n<100

```c
CODE-
#include <stdio.h>
int main() {
    int n;
    printf("Enter the number of integers (less than 100): ");
    scanf("%d", &n);
    if (n <= 0 || n >= 100) {
        printf("Please enter a valid number between 1 and 99.\n");
        return 1;
    }
    int arr[n];
    printf("Enter %d integers:\n", n);
    for (int i = 0; i < n; i++) {
        scanf("%d", &arr[i]);
    }
    int sum = 0;
    int product = 1;
    for (int i = 0; i < n; i++) {
        sum += arr[i];
        product *= arr[i];
    }
    printf("Sum of the entered integers: %d\n", sum);
    printf("Product of the entered integers: %d\n", product);
    return 0;
}
OUTPUT-
Enter the number of integers (less than 100): 5
Enter 5 integers:
21
23
18
19
123
Sum of the entered integers: 204
Product of the entered integers: 20317878
```

27. Write a c program to multiply two matrices.

```c
CODE-
#include <stdio.h>
#define MAX_SIZE 10
void matrixMultiplication(int mat1[MAX_SIZE][MAX_SIZE], int
mat2[MAX_SIZE][MAX_SIZE], int result[MAX_SIZE][MAX_SIZE], int rows1, int cols1,
int cols2) {
```

```c
        for (int i = 0; i < rows1; i++) {
            for (int j = 0; j < cols2; j++) {
                result[i][j] = 0;
                for (int k = 0; k < cols1; k++) {
                    result[i][j] += mat1[i][k] * mat2[k][j];
                }
            }
        }
    }
    void displayMatrix(int matrix[MAX_SIZE][MAX_SIZE], int rows, int cols) {
        printf("Resultant Matrix:\n");
        for (int i = 0; i < rows; i++) {
            for (int j = 0; j < cols; j++) {
                printf("%d ", matrix[i][j]);
            }
            printf("\n");
        }
    }
    int main() {
        int mat1[MAX_SIZE][MAX_SIZE], mat2[MAX_SIZE][MAX_SIZE],
    result[MAX_SIZE][MAX_SIZE];
        int rows1, cols1, rows2, cols2;
        printf("Enter the number of rows and columns for first matrix (max 10 x 10):
    ");
        scanf("%d %d", &rows1, &cols1);
        printf("Enter elements of first matrix:\n");
        for (int i = 0; i < rows1; i++) {
            for (int j = 0; j < cols1; j++) {
                scanf("%d", &mat1[i][j]);
            }
        }
        printf("Enter the number of rows and columns for second matrix (max 10 x
    10): ");
        scanf("%d %d", &rows2, &cols2);
        if (cols1 != rows2) {
            printf("Matrix multiplication not possible.\n");
            return 1;
        }
        printf("Enter elements of second matrix:\n");
        for (int i = 0; i < rows2; i++) {
            for (int j = 0; j < cols2; j++) {
                scanf("%d", &mat2[i][j]);
            }
        }
        matrixMultiplication(mat1, mat2, result, rows1, cols1, cols2);
        displayMatrix(result, rows1, cols2);
        return 0;
    }
```

Roll-737
Sem-1
Year-2023

OUTPUT–
Enter the number of rows and columns for first matrix (max 10 x 10): 2 2
Enter elements of first matrix:
5 6 8 9
Enter the number of rows and columns for second matrix (max 10 x 10): 2 2
Enter elements of second matrix:
7 8 6 2
Resultant Matrix:
71 52
110 82

28. Write a c program to create a square matrix of size m dynamically. Value of m will be provided by the user. Calculate the sum of all the left diagonal elements of the sum of all the right diagonal elements of the matrix. Print the two sums.

CODE–
```c
#include <stdio.h>
#include <stdlib.h>
int main() {
    int m;
    printf("Enter the size of the square matrix: ");
    scanf("%d", &m);
    if (m <= 0) {
        printf("Invalid size for matrix.\n");
        return 1;
    }
    int **matrix = (int **)malloc(m * sizeof(int *));
    for (int i = 0; i < m; i++) {
        matrix[i] = (int *)malloc(m * sizeof(int));
    }
    printf("Enter elements of the square matrix:\n");
    for (int i = 0; i < m; i++) {
        for (int j = 0; j < m; j++) {
            scanf("%d", &matrix[i][j]);
        }
    }
    int leftDiagonalSum = 0;
    int rightDiagonalSum = 0;
    for (int i = 0; i < m; i++) {
        leftDiagonalSum += matrix[i][i];
        rightDiagonalSum += matrix[i][m - i - 1];
    }
    printf("Sum of left diagonal elements: %d\n", leftDiagonalSum);
    printf("Sum of right diagonal elements: %d\n", rightDiagonalSum);
    // Freeing dynamically allocated memory
    for (int i = 0; i < m; i++) {
        free(matrix[i]);
    }
```

```
        free(matrix);
        return 0;
    }
    OUTPUT–
    Enter the size of the square matrix: 3 3
    Enter elements of the square matrix:
    7 8 9 4 5 6 1 2 3
    Sum of left diagonal elements: 9
    Sum of right diagonal elements: 18
```

29. Write a c program to find the length of a string without strlen() function.

```
    CODE–
    #include <stdio.h>
    int stringLength(char str[]) {
        int length = 0;
        while (str[length] != '\0') {
            length++;
        }
        return length;
    }
    int main() {
        char inputString[100];
        printf("Enter a string: ");
        scanf("%s", inputString);
        int len = stringLength(inputString);
        printf("Length of the string: %d\n", len);
        return 0;
    }
    OUTPUT–
    Enter a string: Hello
    Length of the string: 5
```

30. Write a c program to compare two strings lexicographically.

```
    CODE–
    #include <stdio.h>
    int compareStrings(char str1[], char str2[]) {
        int i = 0;
        while (str1[i] != '\0' || str2[i] != '\0') {
            if (str1[i] != str2[i]) {
                return str1[i] – str2[i];
            }
            i++;
        }
        return 0;
    }
    int main() {
```

```c
        char string1[100], string2[100];
        printf("Enter first string: ");
        scanf("%s", string1);
        printf("Enter second string: ");
        scanf("%s", string2);
        int result = compareStrings(string1, string2);
        if (result == 0) {
           printf("Strings are equal.\n");
        } else if (result < 0) {
           printf("String 1 is lexicographically smaller than String 2.\n");
        } else {
           printf("String 1 is lexicographically greater than String 2.\n");
        }
        return 0;
    }
```

OUTPUT–
Enter first string: Hi
Enter second string: Hello
String 1 is lexicographically greater than String 2.

31.   Write a c program to reverse a string without the library function.

CODE–
```c
#include <stdio.h>
void reverseString(char str[]) {
   int length = 0;
   while (str[length] != '\0') {
      length++;
   }
   int start = 0;
   int end = length – 1;
   while (start < end) {
      char temp = str[start];
      str[start] = str[end];
      str[end] = temp;
      start++;
      end––;
   }
}
int main() {
   char inputString[100];
   printf("Enter a string: ");
   scanf("%s", inputString);
   printf("Original String: %s\n", inputString);
   reverseString(inputString);
   printf("Reversed String: %s\n", inputString);
   return 0;
}
```

OUTPUT–
Enter a string: Mate
Original String: Mate
Reversed String: etaM

32. Write a c program to create your data type to store data of 10 books and print the data in proper format.

CODE–
```c
#include <stdio.h>
#define MAX_BOOKS 10
#define MAX_TITLE_LENGTH 100
#define MAX_AUTHOR_LENGTH 50
struct Book {
    char title[MAX_TITLE_LENGTH];
    char author[MAX_AUTHOR_LENGTH];
    int year;
    float price;
};
int main() {
    struct Book books[MAX_BOOKS];
    printf("Enter information for 10 books:\n");
    for (int i = 0; i < MAX_BOOKS; i++) {
        printf("Book %d\n", i + 1);
        printf("Enter title: ");
        scanf("%s", books[i].title);
        printf("Enter author: ");
        scanf("%s", books[i].author);
        printf("Enter year of publication: ");
        scanf("%d", &books[i].year);
        printf("Enter price: ");
        scanf("%f", &books[i].price);
    }
    printf("\nDisplaying information of all 10 books:\n");
    for (int i = 0; i < MAX_BOOKS; i++) {
        printf("Book %d\n", i + 1);
        printf("Title: %s\n", books[i].title);
        printf("Author: %s\n", books[i].author);
        printf("Year of publication: %d\n", books[i].year);
        printf("Price: $%.2f\n", books[i].price);
        printf("\n");
    }
    return 0;
}
```
OUTPUT–
Enter information for 10 books:
Book 1
Enter title: abc

Enter author: zxd
Enter year of publication: 2023
Enter price: 200
Book 2
Enter title: fer
Enter author: ree
Enter year of publication: 2012
Enter price: 365
Book 3
Enter title: wqw
Enter author: dss
Enter year of publication: 2022
Enter price: 230
Book 4
Enter title: iii
Enter author: qwq
Enter year of publication: 2015
Enter price: 599
Book 5
Enter title: qwe
Enter author: qwr
Enter year of publication: 2015
Enter price: 365
Book 6
Enter title: fgr
Enter author: rtg
Enter year of publication: 1932
Enter price: 1230
Book 7
Enter title: fgf
Enter author: gfh
Enter year of publication: 2016
Enter price: 533
Book 8
Enter title: rfr
Enter author: tyh
Enter year of publication: 2003
Enter price: 2366
Book 9
Enter title: rty
Enter author: ere
Enter year of publication: 2013
Enter price: 366
Book 10
Enter title: qwr
Enter author: ert
Enter year of publication: 2022
Enter price: 500

Displaying information of all 10 books:
Book 1
Title: abc
Author: zxd
Year of publication: 2023
Price: $200.00

Book 2
Title: fer
Author: ree
Year of publication: 2012
Price: $365.00

Book 3
Title: wqw
Author: dss
Year of publication: 2022
Price: $230.00

Book 4
Title: iii
Author: qwq
Year of publication: 2015
Price: $599.00

Book 5
Title: qwe
Author: qwr
Year of publication: 2015
Price: $365.00

Book 6
Title: fgr
Author: rtg
Year of publication: 1932
Price: $1230.00

Book 7
Title: fgf
Author: gfh
Year of publication: 2016
Price: $533.00

Book 8
Title: rfr
Author: tyh
Year of publication: 2003

Price: $2366.00

Book 9
Title: rty
Author: ere
Year of publication: 2013
Price: $366.00

Book 10
Title: qwr
Author: ert
Year of publication: 2022
Price: $500.00

33. Write a c program to create your data type to store data of 10 books and print the data in proper format.

CODE–
```c
#include <stdio.h>
#include <string.h>
#define MAX_STUDENTS 5
#define MAX_NAME_LENGTH 50
struct Student {
    int rollNumber;
    char name[MAX_NAME_LENGTH];
    int age;
};
void swap(struct Student *a, struct Student *b) {
    struct Student temp = *a;
    *a = *b;
    *b = temp;
}
void sortByRollNumber(struct Student students[], int n) {
    for (int i = 0; i < n – 1; i++) {
        for (int j = 0; j < n – i – 1; j++) {
            if (students[j].rollNumber > students[j + 1].rollNumber) {
                swap(&students[j], &students[j + 1]);
            }
        }
    }
}
void sortByAge(struct Student students[], int n) {
    for (int i = 0; i < n – 1; i++) {
        for (int j = 0; j < n – i – 1; j++) {
            if (students[j].age > students[j + 1].age) {
                swap(&students[j], &students[j + 1]);
            }
        }
```

```c
        }
    }
    void displayStudents(struct Student students[], int n) {
        printf("Student Records:\n");
        for (int i = 0; i < n; i++) {
            printf("Roll Number: %d, Name: %s, Age: %d\n", students[i].rollNumber,
students[i].name, students[i].age);
        }
    }
    int main() {
        struct Student students[MAX_STUDENTS];
        printf("Enter student details for %d students:\n", MAX_STUDENTS);
        for (int i = 0; i < MAX_STUDENTS; i++) {
            printf("Enter details for student %d:\n", i + 1);
            printf("Roll Number: ");
            scanf("%d", &students[i].rollNumber);
            printf("Name: ");
            scanf("%s", students[i].name);
            printf("Age: ");
            scanf("%d", &students[i].age);
        }
        int choice;
        printf("\nEnter sorting criteria:\n");
        printf("1. Sort by Roll Number\n");
        printf("2. Sort by Age\n");
        printf("Enter your choice (1 or 2): ");
        scanf("%d", &choice);
        if (choice == 1) {
            sortByRollNumber(students, MAX_STUDENTS);
            printf("\nSorted by Roll Number:\n");
            displayStudents(students, MAX_STUDENTS);
        } else if (choice == 2) {
            sortByAge(students, MAX_STUDENTS);
            printf("\nSorted by Age:\n");
            displayStudents(students, MAX_STUDENTS);
        } else {
            printf("\nInvalid choice.\n");
        }
        return 0;
    }
```

OUTPUT–
Enter student details for 5 students:
Enter details for student 1:
Roll Number: 737
Name: Abhirup
Age: 18
Enter details for student 2:
Roll Number: 788

Name: Ajay
Age: 19
Enter details for student 3:
Roll Number: 736
Name: Sanjay
Age: 21
Enter details for student 4:
Roll Number: 764
Name: Gaurav
Age: 19
Enter details for student 5:
Roll Number: 744
Name: Arun
Age: 18

Enter sorting criteria:
1. Sort by Roll Number
2. Sort by Age
Enter your choice (1 or 2): 1

Sorted by Roll Number:
Student Records:
Roll Number: 736, Name: Sanjay, Age: 21
Roll Number: 737, Name: Abhirup, Age: 18
Roll Number: 744, Name: Arun, Age: 18
Roll Number: 764, Name: Gaurav, Age: 19
Roll Number: 788, Name: Ajay, Age: 19

34. Write a c program to copy one file into another file where the file names will be provided through command line. The input will be like: mycopy<source file><destination file>.

CODE–
```c
#include <stdio.h>
#include <stdlib.h>
int main()
{
    FILE *sourceFile;
    FILE *destFile;
    char sourcePath[100];
    char destPath[100];
    char ch;
    printf("Enter source file path: ");
    scanf("%s", sourcePath);
    printf("Enter destination file path: ");
    scanf("%s", destPath);
    sourceFile  = fopen(sourcePath, "r");
```

```
        destFile    = fopen(destPath,   "w");
        if (sourceFile == NULL || destFile == NULL)
        {
            printf("\nUnable to open file.\n");
            printf("Please check if file exists and you have read/write privilege.\n");
            exit(EXIT_FAILURE);
        }
        ch = fgetc(sourceFile);
        while (ch != EOF)
        {
            fputc(ch, destFile);
            ch = fgetc(sourceFile);
        }
        printf("\nFiles copied successfully.\n");
        fclose(sourceFile);
        fclose(destFile);
        return 0;
    }
```

OUTPUT–
Enter source file path: data.exe
Enter destination file path: copy.exe

Files copied successfully.

35.   Write a c program to read and write stored student record into a file.

CODE–
```
#include<stdio.h>
#include<stdlib.h>
struct student
{
int rollno;
char name[50];
int m1,m2,m3;
}s1;
int main()
{
    FILE *fp;
    fp=fopen(" D:\C_Programs\read and write stored student record.exe","w");
    if(fp==NULL)
    {
    printf("File could not open");
    exit(0);
    }
    printf("Enter student details :: \n");
```

Roll-737
Sem-1
Year-2023

```
            printf("\nName :: ");
            scanf("%s",&s1.name);
            printf("Roll No :: ");
            scanf("%d",&s1.rollno);
            printf("Marks1 :: ");
            scanf("%d",&s1.m1);
            printf("Marks2 :: ");
            scanf("%d",&s1.m2);
            printf("Marks3 :: ");
            scanf("%d",&s1.m3);
            fwrite(&s1,sizeof(s1),1,fp);
            printf("\nRecord has been added successfully !!\n");
            fclose(fp);
            return 0;
        }
        OUTPUT–
        Enter student details ::

        Name :: abhirup datta khan
        Roll No :737: Marks1 :80: Marks2 :85: Marks3 :: 79
        Record has been added successfully !!
```