

NATIONAL INSTITUTE OF TECHNOLOGY KARNATAKA

SURATHKAL, MANGALORE - 575 025

Course Code – CS111

Course Name – Computer Programming Lab

Lab - 07

Date – July 27, 2021

Submitted To

Marwa Mohiddin Ma’am

Department of Computer Science and Engineering

National Institute of Technology Karnataka, Surathkal

Submitted By

Md Rakib Hasan

Roll – 201CS132

Department of Computer Science and Engineering

**POINTER**

**Question – 1**

Program to count frequency of every character present in a line of text

**Answer**

#include <stdio.h>

#define MAX\_SIZE 100

int main()

{

    char line[MAX\_SIZE];

    char \*ptr = line;

    printf("Enter line: ");

    gets(line);

    int i = 0, c, j;

    while (\*(ptr + i))

    {

        c = 1;

        j = i + 1;

        if (\*(ptr + i) != -1)

        {

            while (\*(ptr + j))

            {

                if (\*(ptr + i) == \*(ptr + j))

                {

                    c++;

                    //replacing character with -1 which is already counted

                    \*(ptr + j) = -1;

                }

                j++;

            }

            printf("'%c' = %d\n", \*(ptr + i), c);

        }

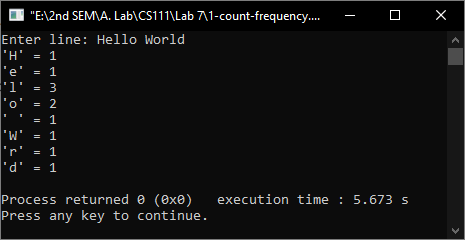
        i++;

    }

    return 0;

}

**Output**



**Question – 2**

Program to swape two number

**Answer**

#include <stdio.h>

int main()

{

    int n1, n2, temp;

    int \*p1 = &n1, \*p2 = &n2;

    printf("Enter num1: ");

    scanf("%d", &n1);

    printf("Enter num2: ");

    scanf("%d", &n2);

    printf("Before swipe num1 = %d & num2 = %d\n", n1, n2);

    //swapping

    temp = \*p1;

    \*p1 = \*p2;

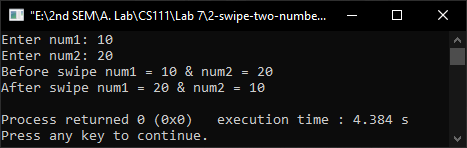
    \*p2 = temp;

    printf("After swipe num1 = %d & num2 = %d\n", n1, n2);

    return 0;

}

**Output**



**Question – 3**

Program to find area and circumference of a circle

**Answer**

#include<stdio.h>

#include<math.h>

#define PI acos(-1)

void calculation(double r, double \*area, double \*circumferemce);

int main()

{

    double r;

    double \*p, area, circumference;

    p = &r;

    printf("Enter radius of the circle: ");

    scanf("%lf", &r);

    calculation(r, &area, &circumference);

    printf("Area = %lf\n", area);

    printf("Circumference = %lf\n", circumference);

    return 0;

}

//calculating area and circumference

void calculation(double r, double \*area, double \*circumferemce)

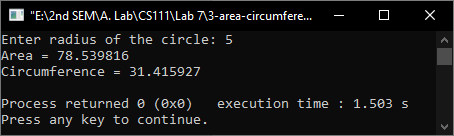
{

    \*area = PI \* r \* r;

    \*circumferemce = 2 \* PI \* r;

}

**Output**



**Question – 4**

Write functions for the following string operations

a) Concatenation b) Comparison c) Length d) Copy e) Reverse

**Answer**

#include <stdio.h>

#include <string.h>

#define MAX\_SIZE 100

//functions

void concat();

void comp();

void length();

void copy();

void rev();

int main()

{

    int select;

    printf("Menu: \n");

    printf("1. Concatenation\n");

    printf("2. Comparison\n");

    printf("3. Length\n");

    printf("4. Copy\n");

    printf("5. Reverse\n");

    printf("Select Operation: ");

    scanf("%d", &select);

    switch (select)

    {

    case 1:

        concat();

        break;

    case 2:

        comp();

        break;

    case 3:

        length();

        break;

    case 4:

        copy();

        break;

    case 5:

        rev();

        break;

    default:

        printf("Wrong Input. Program Closed");

        break;

    }

    return 0;

}

void concat()

{

    char str1[MAX\_SIZE], str2[MAX\_SIZE];

    printf("Enter first string: ");

    gets(str1);

    gets(str1);

    printf("Enter second string: ");

    gets(str2);

    char \*s1 = str1, \*s2 = str2;

    while (\*s1)

    {

        s1++;

    }

    while (\*s2)

    {

        \*s1 = \*s2;

        s2++;

        s1++;

    }

    \*s1 = '\0'; //string should end with \0

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

}

void comp()

{

    char str1[MAX\_SIZE], str2[MAX\_SIZE];

    printf("Enter first string: ");

    gets(str1);

    gets(str1);

    printf("Enter second string: ");

    gets(str2);

    char \*s1 = str1, \*s2 = str2;

    int cm = 0;

    while (\*s1 && \*s2)

    {

        if (\*s1 != \*s2)

        {

            cm = \*s1 > \*s2 ? 1 : 2;

        }

        s1++;

        s2++;

    }

    if (cm == 0)

    {

        printf("They both are same");

    }

    else if (cm == 1)

    {

        printf("%s is lexicographically smaller than %s", str2, str1);

    }

    else if (cm == 2)

    {

        printf("%s is lexicographically smaller than %s", str1, str2);

    }

}

void length()

{

    char str1[MAX\_SIZE];

    printf("Enter string: ");

    gets(str1);

    gets(str1);

    char \*s1 = str1;

    int len = 0;

    while (\*s1)

    {

        len++;

        s1++;

    }

    printf("Length = %d", len);

}

void copy()

{

    char str1[MAX\_SIZE], str2[MAX\_SIZE];

    printf("Enter string: ");

    gets(str1);

    gets(str1);

    char \*s1 = str1, \*s2 = str2;

    while (\*s1)

    {

        \*s2 = \*s1;

        s1++;

        s2++;

    }

    \*s2 = '\0'; // string should end with \0

    printf("Copied string: %s", str2);

}

void rev()

{

    char str1[MAX\_SIZE], temp;

    printf("Enter string: ");

    gets(str1);

    gets(str1);

    char \*start = str1, \*end = str1;

    int i, len = strlen(str1);

    end = end + (len - 1);

    for (i = 0; i < len / 2; i++)

    {

        temp = \*start;

        \*start = \*end;

        \*end = temp;

        start++;

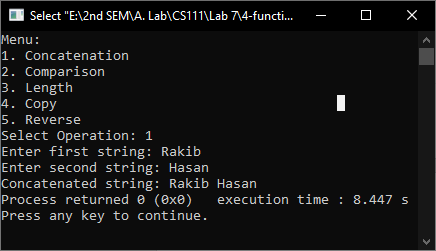
        end--;

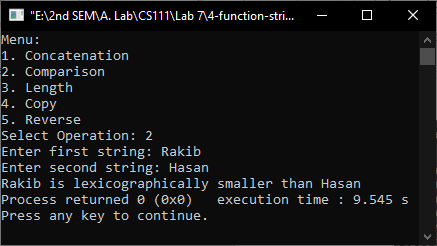
    }

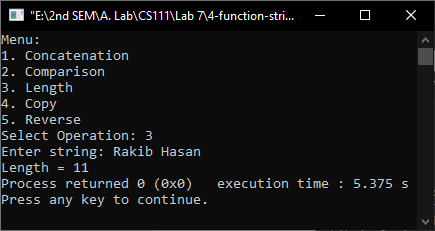
    printf("Reversed string: %s", str1);

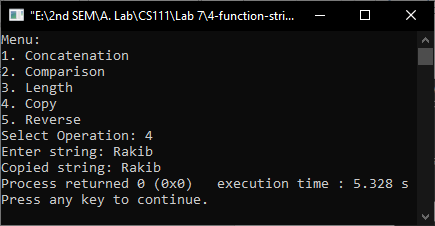
}

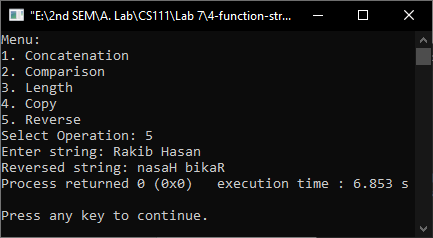
**Output**











**Question – 5**

Write a program to display the greatest of N numbers – use malloc() function

**Answer**

#include <stdio.h>

#include <stdlib.h>

int main()

{

    int n, i, temp;

    printf("Enter total element: ");

    scanf("%d", &n);

    int \*arr;

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

    if (arr == NULL)

    {

        printf("Memory not allocated");

        exit(0);

    }

    else

    {

        printf("Enter element: ");

        for (i = 0; i < n; i++)

        {

            scanf("%d", (arr + i));

        }

        temp = arr[0];

        for (i = 0; i < n; i++)

        {

            if (arr[i] > temp)

            {

                temp = arr[i];

            }

        }

    }

free(arr);

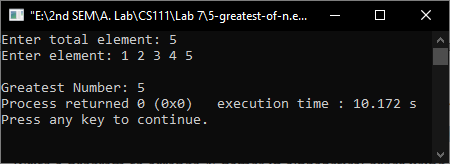
    arr = NULL;

    printf("\nGreatest Number: %d", temp);

    return 0;

}

**Output**



**Qusetion – 6**

Write a program to arrange N names in alphabetical order using dynamic memory allocation

**Answer**

#include <stdio.h>

#include <stdlib.h>

#include <string.h>

int main()

{

    int n, i, j;

    char \*\*names, temp[100];

    printf("Enter total name: ");

    scanf("%d", &n);

    names = (char \*\*)malloc(n \* sizeof(char \*));

    for (i = 0; i < n; i++)

    {

        names[i] = (char \*)malloc(100 \* sizeof(char));

    }

    printf("Enter name: ");

    getchar();

    for (i = 0; i < n; i++)

    {

        gets(names[i]);

    }

    for (i = 0; i < n - 1; i++)

    {

        for (j = i + 1; j < n; j++)

        {

            if (strcmp(names[i], names[j]) > 0) //sorting

            {

                strcpy(temp, names[i]);

                strcpy(names[i], names[j]);

                strcpy(names[j], temp);

            }

        }

    }

    printf("Names in alphabetical order: ");

    for (i = 0; i < n; i++)

    {

        printf("%s ", names[i]);

        free(names[i]);

        names[i] = NULL;

    }

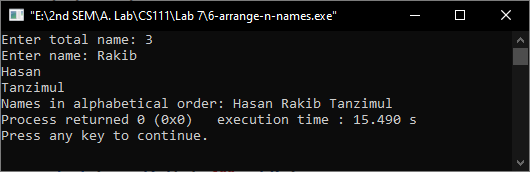
    free(names);

    names = NULL;

    return 0;

}

**Output**

****