A Micro Project Report

on

Problem Solving using C Language

Submitted by

K.Navya Madhuri (23471A05HR)

DEPARTMENT OF COMPUTER SCIENCE AND ENGINEERING

NARASARAOPETA ENGINEERING COLLEGE: NARASARAOPET

(AUTONOMOUS)

Accredited by NAAC with A+ Grade and NBA under Tier-1

NIRF rank in the band of 201-300 and is an ISO 9001:2015 certified Approved by AICTE, New Delhi, Permanently affiliated to JNTU Kakinada, Approved by AICTE, Accredited by NBA and accredited 'A+' grade by NAAC Narasaraopet-522601, Palnadu(Dt.), Andhra Pradesh, India

2024-2025

NARASARAOPETA ENGINEERING COLLEGE: NARASARAOPET

(AUTONOMOUS)

DEPARTMENT OF COMPUTER SCIENCE AND ENGINEERING

CERTIFICATE

This is to certify that K.Navya Madhuri, Roll No: 23471A05HR, a Second Year Student of the Department of Computer Science and Engineering, has completed the Micro Project Satisfactorily in “Problem Solving using C Language" for the Academic Year 2024-2025..

Project Co-Ordinator
Dr. Rama Krishna. Eluri, M.Tech., Ph.D.
M.Tech., Ph.D.
Professor

HEAD OF THE DEPARTMENT
Dr. S. N. Tirumala Rao,
Asst. Professor

INDEX

Frequency Count of Digits in String

AIM:

Write a C program to given a string, consisting of alphabets and digits, find the frequency of each digit in the given string.

```
 #include<stdio.h>
#include<string.h>
int main()
 {
 char str[100];
 int freq[10] = {0};
printf("Enter a string: ");
fgets(str, sizeof(str), stdin);
 for (int i = 0; i < strlen(str); i++)
 {
 if (str[i] >= '0' && str[i] <= '9')
 {
 freq[str[i] - '0']++; }
 }
 printf("Frequency of digits in the string:\n");
 for (int i = 0; i < 10; i++) {
 printf("Digit '%d': %d times\n", i, freq[i]);
 }
 return 0;
}
```

Input:

Enter a string:a1b2c3d1234

OUTPUT:

```
Enter a string: a1b2c3d1234

Frequencyofdigitsinthestring: Digit '0': 0

times

Digit'1':2times

Digit'2':2times

Digit'3':2times

Digit'4':1times

Digit'5':0times

Digit'6':0times

Digit'7':0times

Digit'8':0times

Digit'9':0times
```

Aim :

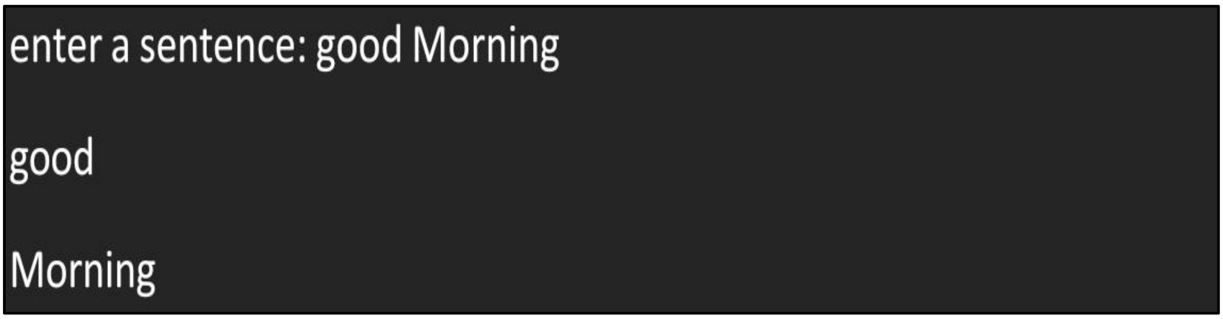
C program given a Sentence, Print Each Word of the Sentence in a New Line .

```
#include <stdio.h

 int main()

 {

  char word[100];

  printf("enter a sentence:");

  while(scanf("%19s",word)==1)

  {

  printf("%s\n",word);

  }

  return 0;

 }
```

Input:

enter a sentence: good Morning

output :



```
enter a sentence: good Morning

good

Morning
```

Areas of Triangle in Sorted Order

Aim :

You are given Triangles, specifically, their sides . Print them in the Same style but sorted by their areas form the smallest one to the largest. It is Guaranteed that all the areas are different.

```
#include <stdio.h>

#include <math.h>

#define MAX_TRIANGLES 100

typedef struct {

 double a, b, c;

 double area;

} Triangle;

double calculate_area(double a, double b, double c)

 { double s = (a + b + c) / 2.0;

 return sqrt(s * (s - a) * (s - b) * (s - c));

}

int compare_areas(const void *a, const void *b)

 { Triangle *triangleA = (Triangle *)a;

 Triangle *triangleB = (Triangle *)b;

 if (triangleA->area < triangleB->area)

return -1;

 if (triangleA->area > triangleB->area)
return 1;
 return 0;
}
int main()
 { int n;
 printf("Enter the number of triangles: ");
 scanf("%d", &n);
 Triangle triangles [MAX_TRIANGLES];
 for (int i = 0; i < n; i++) {
 printf("Enter the sides of triangle %d (a b c): ", i + 1);
 scanf("%lf %lf %lf", &triangles[i].a, &triangles[i].b, &triangles[i].c);
triangles[i].area = calculate_area(triangles[i].a, triangles[i].b, triangles
[i].c);
 }
 qsort(triangles, n, sizeof(Triangle), compare_areas);
 printf("\nTriangles sorted by area:\n");
 for (int i = 0; i < n; i++) {
 printf("Triangle %d: sides = (%.2f, %.2f, %.2f), area = %.2f\n",
 i + 1, triangles[i].a, triangles[i].b, triangles[i].c, triangles[i].area);
 }
 return 0;
}
```

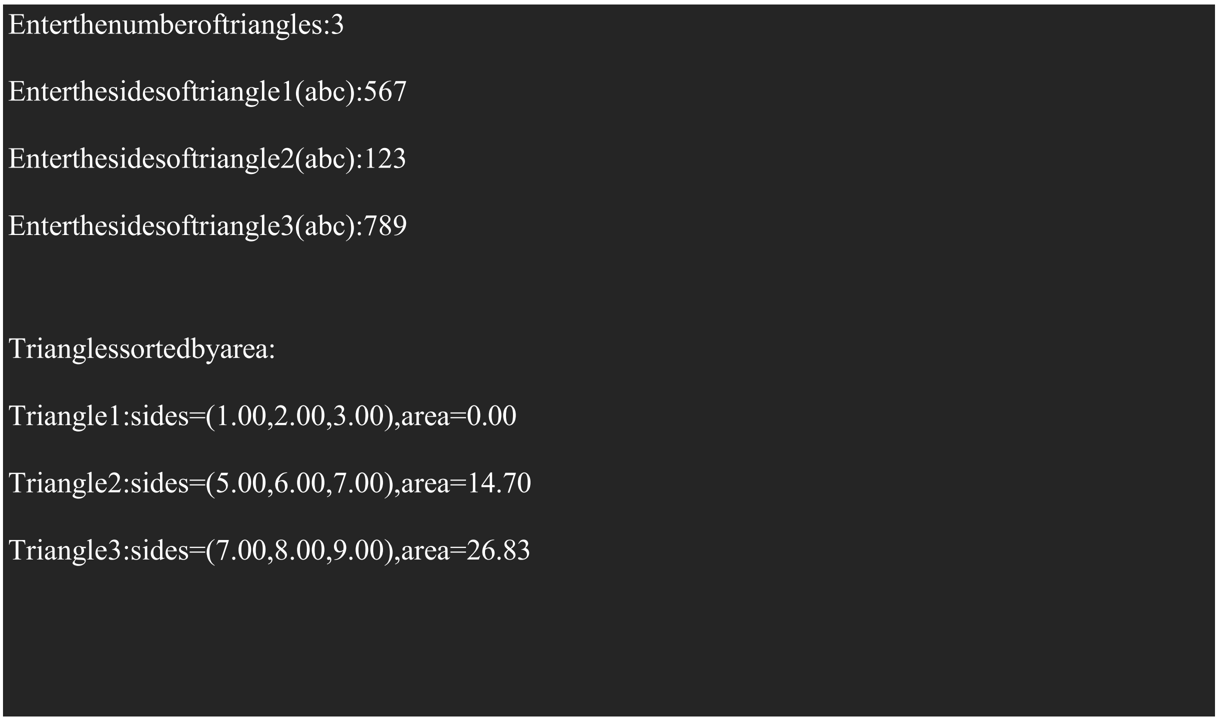
Input:

```
Enter the number of triangles:3
Enter the sides of triangle1(abc):567
```

Enter the sides of triangle2(abc):123

Enter the sides of triangle3(abc):789

OUTPUT:



```
Enterthenumberoftriangles:3

Enterthesidesoftriangle1(abc):567

Enterthesidesoftriangle2(abc):123

Enterthesidesoftriangle3(abc):789


Trianglessortedbyarea:

Triangle1:sides=(1.00,2.00,3.00),area=0.00

Triangle2:sides=(5.00,6.00,7.00),area=14.70

Triangle3:sides=(7.00,8.00,9.00),area=26.83
```

Match-Stick Game

AIM:

Write a program for a matchstick game being playedbetween the computer and a user.

Your program shouldensure that the computer always wins. Rules

for the game are as follows:

-There are 21 matchsticks.

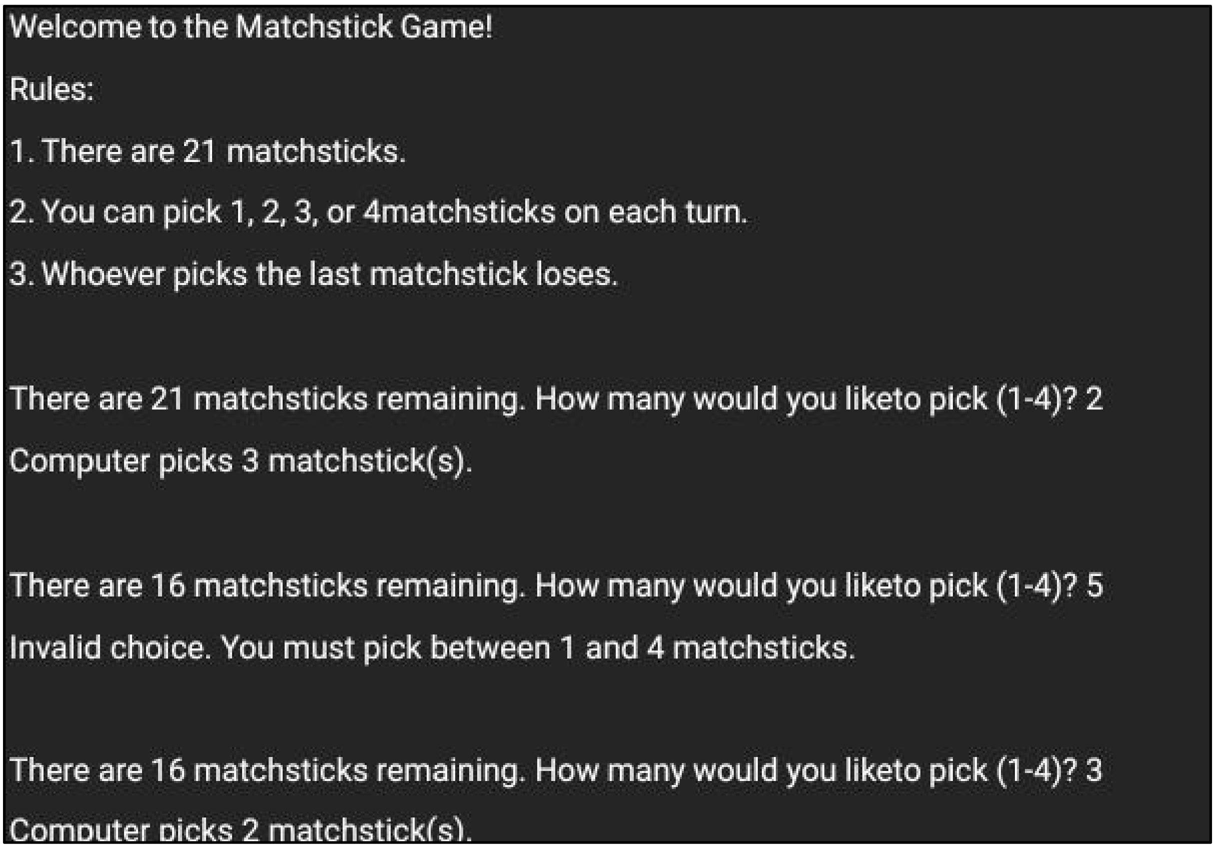
-The computer asks the player to pick 1,2,3,or 4 matchsticks.

-After the person picks, the computer does its picking.

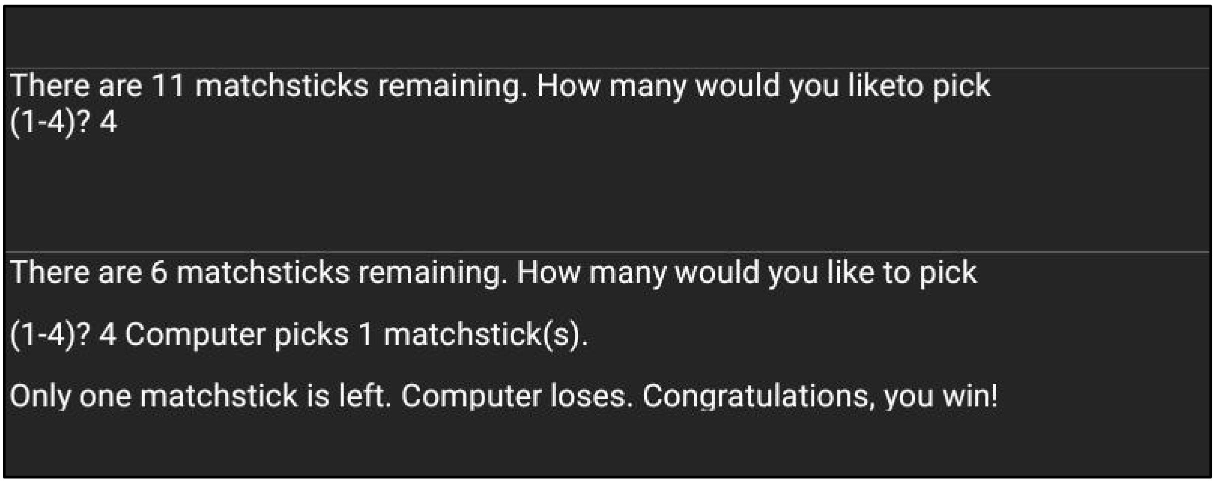
-Whoever is forced to pick up the last matchstick loses the game.

```
#include <stdio.h>
int main()
 {
 int matchsticks = 21, user_pick, computer_pick;
 printf("Welcome to the Matchstick Game!\n");
 printf("Rules:\n1. There are 21 matchsticks.\n2. You can pick 1, 2, 3, or 4matchsticks on each turn.\n3. Whoever picks the last matchstick loses.\n");
 while (matchsticks > 1) {
 printf("\nThere are %d matchsticks remaining. How many would you liketo pick (1-4)? ", matchsticks);
 scanf("%d", &user_pick);
 if (user_pick < 1 || user_pick > 4) {
 printf("Invalid choice. You must pick between 1 and 4 matchsticks.\n");
 continue;
 }
 matchsticks -= user_pick;
 if (matchsticks == 1) {
 printf("Only one matchstick is left. You lose!\n");
 break;
 }
 computer_pick = 5 - user_pick;
 matchsticks -= computer_pick;
 printf("Computer picks %d matchstick(s).\n", computer_pick);
 if (matchsticks == 1) {
 printf("Only one matchstick is left. Computer loses. Congratulations, you win!\n");
 break;
}
}
return 0;
}
```

Output :



```
Welcome to the Matchstick Game!
Rules:
1. There are 21 matchsticks.
2. You can pick 1, 2, 3, or 4matchsticks on each turn.
3. Whoever picks the last matchstick loses.

There are 21 matchsticks remaining. How many would you liketo pick (1-4)? 2
Computer picks 3 matchstick(s).

There are 16 matchsticks remaining. How many would you liketo pick (1-4)? 5
Invalid choice. You must pick between 1 and 4 matchsticks.

There are 16 matchsticks remaining. How many would you liketo pick (1-4)? 3
Computer picks 2 matchstick(s).
```



```
There are 11 matchsticks remaining. How many would you liketo pick
(1-4)? 4


There are 6 matchsticks remaining. How many would you like to pick

(1-4)? 4 Computer picks 1 matchstick(s).

Only one matchstick is left. Computer loses. Congratulations, you win!
```