A Micro Project Report

on

Problem Solving using C Language

Submitted by

Gogula Srilakshmi Swathi (23471A05FO)

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 Gogula Srilakshmi Swathi, Roll No: 23471A05FO, 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
Mr. M. Venkata Rao, M.Tech.
Asst. Professor

HEAD OF THE DEPARTMENT
Dr. S. N. Tirumala Rao, M.Tech., Ph.D.
Professor

INDEX

S.No	Description
1.	Read two complex numbers (a+bi) from the keyboard and perform the following operations with the help of structures and also use malloc() to create the object. 1. Addition of two complex numbers. 2. Subtraction of two complex numbers. 3. Multiplication of two complex numbers. 4. Addition of N complex numbers.

Operation of complex numbers

AIM:

Read two complex numbers (a+bi) from the keyboard and perform the following operations with the help of structures and also use malloc() to create the object

1. Addition of two complex numbers

```
#include <stdio.h>
#include<stdlib.h>
struct complex
{
   float real;
   float imag;
};
struct complex addcomplex(struct complex*c1,struct complex*c2)
{
struct complex result;
result.real=c1->real+c2->real;
result.imag=c1->imag+c2->imag;
return result;
}
int main()
{
   struct complex *c1,*c2,result;
c1=(struct complex*)malloc(sizeof(struct complex*));
c2=(struct complex*)malloc(sizeof(struct complex*));
if(c1==NULL || c2==NULL)
{
printf("memory allocation is failed!\n");
return -1;
}
```

```
printf("Enter the real and imaginary part of first complex(a+bi):\n");
printf("Real part:");
scanf("%f",&c1->real);
printf("Imaginary part:");
scanf("%f",&c1->imag);
printf("Enter the real and imaginary part of second complex(a+bi):\n");
printf("Real part:");
scanf("%f",&c2->real);
printf("Imaginary part:");
scanf("%f",&c2->imag);
result=addcomplex(c1,c2);
printf("\n sum of the given complex
number:%.2f+%.2fi\n",result.real,result.imag);
free(c1);
free(c2);
return 0;
}
```

OUTPUT:-

INPUT:-

Enter the real and imaginary part of first complex(a+bi):

Real part:5.0

Imaginary part:4.0

Enter the real and imaginary part of second complex(a+bi):

Real part:3.0

Imaginary part:2.0

OUTPUT:-

Sum of the given complex number:8.0+6.0i

2.Subtraction of two complex numbers

```
#include<stdio.h>
#include<stdlib.h>
struct complex
{
   float real;
   float imag;
};
struct complex subtractcomplex(struct complex*c1,struct complex*c2)
{
struct complex result;
result.real=c1->real-c2->real;
result.imag=c1->imag-c2->imag;
return result;
}
int main()
{
   struct complex *c1,*c2,result;
c1=(struct complex*)malloc(sizeof(struct complex*));
c2=(struct complex*)malloc(sizeof(struct complex*));
if(c1==NULL || c2==NULL)
{
printf("memory allocation is failed!\n");
return -1;
}
printf("Enter the real and imaginary part of first complex(a+bi):\n");
```

```
printf("Real part:");
scanf("%f",&c1->real);
printf("Imaginary part:");
scanf("%f",&c1->imag);
printf("Enter the real and imaginary part of second complex(a+bi):\n");
printf("Real part:");
scanf("%f",&c2->real);
printf("Imaginary part:");
scanf("%f",&c2->imag);
result=subtractcomplex(c1,c2);
printf("\n difference of the given complex number:%.2f-
%.2fi\n",result.real,result.imag);
free(c1);
free(c2);
return 0;
}
```

OUTPUT:-

INPUT:-

Enter the real and imaginary part of first complex(a+bi):

Real part:3.0

Imaginary part:4.0

Enter the real and imaginary part of second complex(a+bi):

Real part:1.0

Imaginary part:2.0

OUTPUT:-

difference of the given complex number:2.00-2.00i

3.Multiplication of two complex numbers

```
#include <stdio.h>
#include<stdlib.h>
struct complex
{
   float real;
   float imag;
};
struct complex multiplycomplex(struct complex*c1,struct complex*c2)
{
struct complex result;
result.real=c1->real*c2->real;
result.imag=c1->imag*c2->imag;
return result;
}
int main()
{
   struct complex *c1,*c2,result;
c1=(struct complex*)malloc(sizeof(struct complex*));
c2=(struct complex*)malloc(sizeof(struct complex*));
if(c1==NULL || c2==NULL)
{
printf("memory allocation is failed!\n");
return -1;
}
```

```
printf("Enter the real and imaginary part of first complex(a+bi):\n");
printf("Real part:");
scanf("%f",&c1->real);
printf("Imaginary part:");
scanf("%f",&c1->imag);
printf("Enter the real and imaginary part of second complex(a+bi):\n");
printf("Real part:");
scanf("%f",&c2->real);
printf("Imaginary part:");
scanf("%f",&c2->imag);
result=multiplycomplex(c1,c2);
printf("\n multiply of the given complex
number:%.2f*%.2fi\n",result.real,result.imag);
free(c1);
free(c2);
return 0;
}
```

OUTPUT:-

INPUT:-

Enter the real and imaginary part of first complex(a+bi):

Real part:3.0

Imaginary part:5.0

Enter the real and imaginary part of second complex(a+bi):

Real part:1.0

Imaginary part:4.0

OUTPUT:-

multiply of the given complex number:3.00*20.00i

4. Addition of N complex numbers

```
#include <stdio.h>
#include <stdlib.h>
typedef struct
 {
   float real;
   float imag;
} Complex;
Complex addComplex(Complex c1, Complex c2)
 {
   Complex result;
   result.real = c1.real + c2.real;
   result.imag = c1.imag + c2.imag;
   return result;
}

int main()
 {
   int n, l;
   printf("Enter the number of complex numbers: ");
   scanf("%d", &n);
   Complex *complexNumbers = (Complex *)malloc(n * sizeof(Complex));
   if (complexNumbers == NULL)
 {
       printf("Memory allocation failed!\n");
       return 1;
```

```
    }
    for (i = 0; i < n; i++)
 {
        printf("Enter complex number %d (real and imaginary parts):\n", i + 1);
        printf("Real part: ");
        scanf("%f", &complexNumbers[i].real);
        printf("Imaginary part: ");
        scanf("%f", &complexNumbers[i].imag);
    }
    Complex sum = {0.0, 0.0};
    for (i = 0; i < n; i++)
{
        sum = addComplex(sum, complexNumbers[i]);
    }
    printf("\nThe sum of the complex numbers is: %.2f + %.2fi\n", sum.real,
sum.imag);
    free(complexNumbers);
    return 0;
}
```

OUTPUT:-

INPUT:-

Enter the number of complex numbers: 4

Enter complex number 1 (real and imaginary parts):

Real part: 6

Imaginary part: 4

Enter complex number 2 (real and imaginary parts):

Real part: 7

Imaginary part: 3

Enter complex number 3 (real and imaginary parts):

Real part: 5

Imaginary part: 7

Enter complex number 4 (real and imaginary parts):

Real part: 8

Imaginary part: 4

OUTPUT:-

The sum of the complex numbers is: 26.00 + 18.00i