A Micro Project Report

on

Problem Solving using C Language

Submitted by

Konda Reddy (23471A05F5)



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 Konda Reddy, Roll No: 23471A05F5, 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 HEAD OF THE DEPARTMENT

Dr. Rama Krishna. Eluri, M.Tech., Ph.D. Dr. S. N. Tirumala Rao, M.Tech.,

Ph.D. Asst. Professor Professor

# INDEX

S.No

Description

1.

C Program to addition two rational numbers.

2.

C Program to subtraction of two rational numbers.

3.

C program to multiplication of two rational numbers.

4.

C program to division of two rational numbers.

5.

Write a C program to There is a structure called employee that holds information like employee code, name, date of joining. Write a program to create an array of the structure and enter some data into it. Then ask the user to enter current date. Display the names of those employees whose tenure is 3 or more than 3 years according to the given current date.

/

1. Read two rational numbers (a/b form) from the keyboard do the following operations with the help of structures

AIM: addition two rational numbersn two rational number

SOURCE CODE:

#include<stdio.h>

struct rational

{

int nume;

int denom;

};

int gcd(int a,int b)

{

while(b!=0)

{

int temp=b;

b=a%b;

a=temp;

}

return a;

}

struct rational addrational(struct rational r1,struct rational r2)

{

struct rational result;

int common\_divisor;

result.nume=r1.nume\*r2.denom+r2.nume\*r1.denom;

result.denom=r1.denom\*r2.denom;

common\_divisor=gcd(result.nume,result.denom);

result.nume /=common\_divisor;

result.denom /=common\_divisor;

return result;

}

int main()

{

struct rational r1,r2,result;

clrscr();

printf("Enter first rational number ");

scanf("%d%d\n",&r1.nume,&r1.denom);

printf("Enter second rational number ");

scanf("%d%d\n",&r2.nume,&r2.denom);

if(r1.denom==0||r2.denom==0)

{

printf("denominator can not be zero\n");

return 1;

}

result=addrational(r1,r2);

printf("sum=%d/%d\n",result.nume,result.denom);

getch();

return 0;

}

Output:

Enter first rational number

4

4

Enter second rational number 4

5

sum=7/4

2.AIM: subtraction of two rational numbers

SOURCE CODE:

#include<stdio.h>

#include<conio.h>

struct rational

{

int nume;

int denom;

};

int gcd(int a,int b)

{

while(b!=0)

{

int temp=b;

b=a%b;

a=temp;

}

return a;

}

struct rational subrational(struct rational r1,struct rational r2)

{

struct rational result;

int common\_divisor;

result.nume=r1.nume\*r2.denom-r2.nume\*r1.denom;

result.denom=r1.denom\*r2.denom;

common\_divisor=gcd(result.nume,result.denom);

result.nume /=common\_divisor;

result.denom /=common\_divisor;

return result;

}

int main()

{

struct rational r1,r2,result;

clrscr();

printf("Enter first rational number\n ");

scanf("%d %d",&r1.nume,&r1.denom);

printf("Enter second rational number\n ");

scanf("%d %d",&r2.nume,&r2.denom);

if(r1.denom==0 || r2.denom==0)

{

printf("denominator can not be zero\n");

return 1;

}

result=subrational(r1,r2);

printf("Difference=%d/%d\n",result.nume,result.denom);

getch();

return 0;

}

Output:

Enter first rational number

3

2

Enter second rational number

3

4

Difference=3/4

3.AIM: multiplication of two rational numbers

SOURCE CODE:

#include<stdio.h>

struct rational

{

int nume;

int denom;

};

int gcd(int a,int b)

{

while(b!=0)

{

int temp=b;

b=a%b;

a=temp;

}

return a;

}

struct rational mulrational(struct rational r1,struct rational r2)

{

struct rational result;

int common\_divisor;

result.nume=r1.nume\*r2.nume;

result.denom=r1.denom\*r2.denom;

common\_divisor=gcd(result.nume,result.denom);

result.nume /=common\_divisor;

result.denom /=common\_divisor;

return result;

}

int main()

{

struct rational r1,r2,result;

clrscr();

printf("Enter first rational number\n ");

scanf("%d %d",&r1.nume,&r1.denom);

printf("Enter second rational number\n ");

scanf("%d %d",&r2.nume,&r2.denom);

if(r1.denom==0 || r2.denom==0)

{

printf("denominator can not be zero\n");

return 1;

}

result=mulrational(r1,r2);

printf("Product=%d/%d\n",result.nume,result.denom);

getch();

return 0;

}

Output:

Enter first rational number

2

3

Enter second rational number

3

4

Product=1/2

4.AIM: division of two rational numbers.

SOURCE CODE:

#include<stdio.h>

struct rational

{

int nume;

int denom;

};

int gcd(int a,int b)

{

while(b!=0)

{

int temp=b;

b=a%b;

a=temp;

}

return a;

}

struct rational divrational(struct rational r1,struct rational r2)

{

struct rational result;

int common\_divisor;

result.nume=r1.nume\*r2.denom;

result.denom=r1.denom\*r2.nume;

common\_divisor=gcd(result.nume,result.denom);

result.nume /=common\_divisor;

result.denom /=common\_divisor;

return result;

}

int main()

{

struct rational r1,r2,result;

clrscr();

printf("Enter first rational number\n ");

scanf("%d %d",&r1.nume,&r1.denom);

printf("Enter second rational number\n ");

scanf("%d %d",&r2.nume,&r2.denom);

if(r1.denom==0 || r2.denom==0)

{

printf("denominator can not be zero\n");

return 1;

}

result=divrational(r1,r2);

printf("Reminder=%d/%d\n",result.nume,result.denom);

getch();

return 0;

}

Output:

Enter first rational number

2

3

Enter second rational number

3

4

Reminder=8/9

5.AIM: There is a structure called employee that holds information like employee code, name, date of joining. Write a program to create an array of the structure and enter some data into it. Then ask the user to enter current date. Display the names of those employees whose tenure is 3 or more than 3 years according to the given current date.

SOURCE CODE:

#include<stdio.h>

struct Date

{

int day;

int month;

int year;

};

struct Employee

{

int emp\_code;

char name[30];

struct Date date\_of\_joining;

};

int calculatetenure(struct Date join\_date,struct Date current\_date)

{

int years=current\_date.year-join\_date.year;

if(current\_date.month<join\_date.month || (current\_date.month==join\_date.month && current\_date.day<join\_date.day))

{

years--;

}

return years;

}

int main()

{

int n,i;

struct Date current\_date;

struct Employee employees[10];

clrscr();

printf("enter no.of employees:");

scanf("%d",&n);

for(i=0;i<n;i++)

{

printf("enter the details of employee:%d\n",i+1);

printf("employee code:");

scanf("%d",&employees[i].emp\_code);

printf("name:");

scanf("%s",&employees[i].name);

printf("date of joining:");

scanf("%d %d %d",&employees[i].date\_of\_joining.day,&employees[i].date\_of\_joining.month,&employees[i].date\_of\_joining.year);

}

printf("current date:");

scanf("%d %d %d",&current\_date.day,&current\_date.month,&current\_date.year);

printf("employee with a tenure of 3 years\n");

for(i=0;i<n;i++)

{

int tenure=calculatetenure(employees[i].date\_of\_joining,current\_date);

if(tenure>=3)

{

printf("%s\n",employees[i].name);

}

else

{

printf("nobudy");

}

}

getch();

return 0;

}

Output:

Enter no.of employees:3

Enter the details of employee:1

Employee code:123

Name:kondareddy

Date of joining:25 04 2005

Enter the details of employee:2

Employee code:143

Name:naveen

Date of joining:05 03 2004

Enter the details of employee:3

Employee code:420

Name:siva

Date of joining:03 10 2006

Current date:24 05 2007

Employee with a tenure of 3 years

naveen