A Micro Project Report

on

Problem Solving using C Language

Submitted by

Nunsavath Raj Kalyan

(23471A0542)

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 Nunsavath Raj Kalyan, Roll No: 23471A0542, 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
Ms.CH.Rajini, M.Tech.
Asst. Professor

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

INDEX

Number of prime numbers

AIM:

C Program to count number of prime numbers in given minimum to maximum ranges

```
#include<stdio.h>
#include<conio.h>
void main()
{
int i,j,n1,n2,flag=0,sum=0;
clrscr();
printf("enter number1 value");
scanf("%d",&n1);
printf("enter numbe 2 value");
scanf("%d",&n2);
for(i=n1;i<=n2;i++)
{
 flag=0;
 for(j=1;j<=i;j++)
 {
    if(i%j==0)
     {
         flag+=1;
```

```
    }
  }
if(flag==2)
{
sum=sum+1;
}
}
printf("the total number of prime numbers are %d",sum);
}
```

OUTPUT:

Output

```
enter number1 value1
enter numbe 2 value30
the total number of prime numbers are 10
```

Output

```
enter number1 value1
enter numbe 2 value50
the total number of prime numbers are 15
```

ARMSTRONG NUMBERS

Aim: **C program to generate Armstrong numbers in given minimum to maximum ranges**

```
#include<stdio.h>
void main()
{
   int n,n1,n2,r,temp,sum;
   printf("Enter n1 value:");
   scanf("%d",&n1);
   printf("Enter n2 value:");
   scanf("%d",&n2);
   for(n=n1;n<=n2;n++)
   {
     temp=n;
      sum=0;
      while(temp!=0)
      {
         r=temp%10;
         sum=sum+(r*r*r);
         temp=temp/10;
```

```
        }
        if(sum==n)
        {
          printf("The armstrong numbers are:%d\n",n);
       }
}
}
```

OUTPUT:

```
Output
Enter n1 value:100
Enter n2 value:2000
The armstrong numbers are:153
The armstrong numbers are:370
The armstrong numbers are:371
The armstrong numbers are:407
```

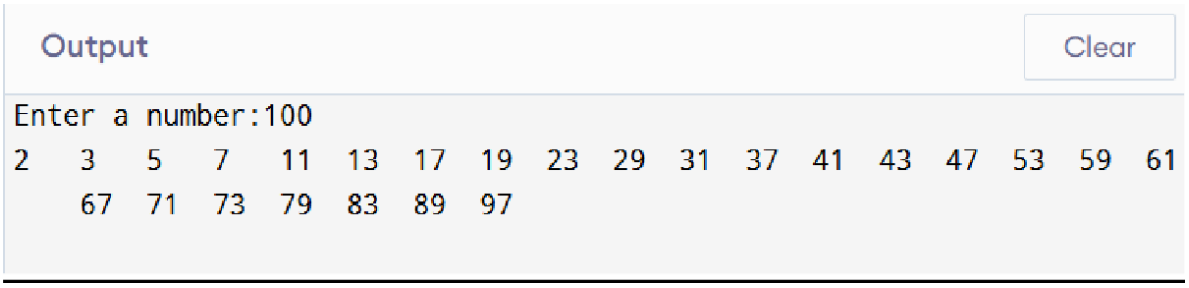
PRIME NUMBERS

AIM:

C program to generate first N prime numbers where n is given by user

```
#include<stdio.h>
int main()
{
   int i,j,num,flage=0;
   printf("Enter a number:");
   scanf("%d",&num);
   for(i=1;i<=num;i++)
   {
      flage=0;
      for(j=1;j<=i;j++)
      {
         if(i%j==0)
         {
            flage+=1;
         }
      }
      if(flage==2)
      printf("%d\t",i);
   }
}
```

OUTPUT:


Output
Clear
Enter a number:100
2 3 5 7 11 13 17 19 23 29 31 37 41 43 47 53 59 61
67 71 73 79 83 89 97

PERFECT NUMBERS

AIM:

C program to generate perfect numbers in given minimum to maximum ranges

```
#include<stdio.h>

void main()

{

int i,j,n1,n2,sum=0;

printf("enter n1 value");

scanf("%d",&n1);

printf("enter n2 value");

scanf("%d",&n2);

for(i=n1;i<=n2;i++)

{

 sum=0;

 for(j=1;j<i;j++)

 {

   if(i%j==0)

   {

        sum=sum+j;

   }

 }
```

```
if(sum==i)

printf("the perfect numbers are %d\n",i);

}

}
```

OUTPUT:

Output

```
enter n1 value1
enter n2 value60
the perfect numbers are 6
the perfect numbers are 28
```

Output

```
enter n1 value
1
enter n2 value
1000
the perfect numbers are 6
the perfect numbers are 28
the perfect numbers are 496
```