A Micro Project Report

on

Problem Solving using C Language

Submitted by

Shaik Mujavar Ajima (23471A05EN)

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 Shaik Mujavar Ajima, Roll No: 23471A05EN, 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.
Asst. Professor

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

INDEX

DECIMAL TO OCATL AND HEXA DECIMAL

AIM:

Write a C program convert Decimal to octal and hexadecimal

```
#include<stdio.h>

int main()

{

int n;

printf("Enter a number(Decimal): ");

scanf("%d", &n);

printf("Octal equivalent of %d(Decimal): %o\n", n, n);

printf("Hexadecimal equivalent of %d(Decimal): %x\n", n, n);

return 0;

}
```

OUTPUT:

```
Enter a number (Decimal): 255
Octal equivalent of 255(Decimal): 377
Hexadecimal equivalent of 255(Decimal): ff
```

DELETE NUMBER IN GIVE POSITION

AIM:

Write a C program delete number in given position in array.

```
#include <stdio.h>

#include <stdlib.h>



int main(void)

{

int i, n, index, arr[10];

printf("Enter the size of the array: ");

scanf("%d", &n);

printf("Enter the elements of the array: \n");

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

{

printf("arr[%d] = ", i);

scanf("%d", &arr[i]);

}
printf("Enter the index of the element to be deleted: ");

scanf("%d", &index);

if (index >= n+1)

{

printf (" \n Deletion is not possible in the array.");

}

else

{

for (i = index; i < n - 1; i++)

arr[i] = arr[i + 1];

printf("The array after deleting the element is: ");

for (i = 0; i < n - 1; i++)

printf("%d ", arr[i]);

return 0;

}

}
```

OUTPUT:

```
Enter the size of the array: 5

Enter the elements of the array:

arr[0] = 10
```

```
arr[1] = 20

arr[2] = 30

arr[3] = 40

arr[4] = 50

Enter the index of the element to be deleted: 2

The array after deleting the element is: 10 20 40 50
```

MERGE TWO ARRAYS

AIM:

Write a C program to merge two arrays.

```
#include <stdio.h>
int main()
{
int n1,n2,n3;
int a[10000], b[10000], c[20000];
printf("Enter the size of first array: ");
scanf("%d",&n1);
printf("Enter the array elements: ");
for(int i = 0; i < n1; i++)
scanf("%d", &a[i]);
printf("Enter the size of second array: ");
scanf("%d",&n2);
printf("Enter the array elements: ");
for(int i = 0; i < n2; i++)
scanf("%d", &b[i]);
n3 = n1 + n2;
for(int i = 0; i < n1; i++)
c[i] = a[i];
for(int i = 0; i < n2; i++)
c[i + n1] = b[i];

printf("The merged array: ");
for(int i = 0; i < n3; i++)
printf("%d ", c[i]);

printf("\nFinal array after sorting: ");
for(int i = 0; i < n3; i++){
int temp;
for(int j = i + 1; j < n3; j++) {
if(c[i] > c[j]) {
temp = c[i];
c[i] = c[j];
c[j] = temp;
}
}
}
for(int i = 0; i < n3 ; i++)
printf(" %d ",c[i]);
return 0;
}
```

OUTPUT:

```
Enter the size of first array: 3

Enter the array elements:

arr[0] = 5

arr[1] = 1

arr[2] = 3

Enter the size of second array: 3

Enter the array elements:

arr[0] = 4

arr[1] = 2

arr[2] = 6

The merged array: 5 1 3 4 2 6

Final array after sorting:  1  2  3  4  5  6
```

PERFECT NUMBERS IN GIVEN RANGE

AIM:

Write a C program to generate perfect numbers in given minimum to maximum range.

```
#include<stdio.h>

#include<conio.h>

void main()

{

int minimum, maximum, sum, i, j;

clrscr();

printf("Enter minimum: ");

scanf("%d", &minimum);

printf("Enter maximum: ");

scanf("%d", &maximum);


/* Generating Perfect Numbers */

for(i=minimum; i<=maximum; i++)

{
```

```
sum = 0;
for(j=1;j< i;j++)
{
if(i%j==0)
{
sum = sum + j;
}
}
if(sum == i)
{
printf("%d\t", i);
}
}
getch();
}
```

OUTPUT:

```
Enter minimum: 1
Enter maximum: 100
6   28
```