A Micro Project Report

on

Problem Solving using C Language

Submitted by

Shaik.Nagoor Bee(23471A05CW)

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 Skaik.Nagoor Bee Roll No: 23471A05CW, 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. Shaik Rafi, M.Tech., (Ph.D).

Asst. Professor

HEAD OF THE DEPARTMENT

Dr. S. N. Tirumala Rao, M.Tech., Ph.D.

Professor

INDEX

Records of n Students

AIM:

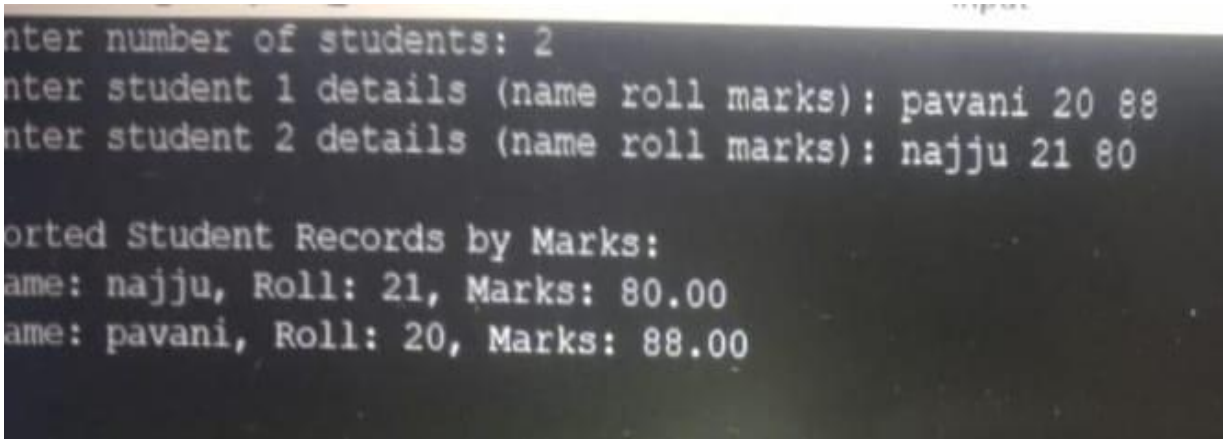
Read Records of n Students in Structure & Sort on the Basis of Marks in Ascending Order

```
#include <stdio.h>

struct Student
 {
    char name[100];
    int roll;
    float marks;
};

int main()
{
    int n;
    printf("Enter number of students: ");
    scanf("%d", &n);
    struct Student students[n];
    for (int i = 0; i < n; i++)
 {
        printf("Enter student %d details (name roll marks): ", i + 1);
        scanf("%s %d %f", students[i].name, &students[i].roll,
&students[i].marks);
   }
```

```
    for (int i = 0; i < n - 1; i++)
   {
        for (int j = i + 1; j < n; j++)
       {
           if (students[i].marks > students[j].marks)
            {
              struct Student temp = students[i];
              students[i] = students[j];
              students[j] = temp;
            }
        }
   }
    printf("\nSorted Student Records by Marks:\n");
     for (int i = 0; i < n; i++)
  {
       printf("Name: %s, Roll: %d, Marks: %.2f\n",
  students[i].name, students[i].roll, students[i].marks);
  }
    return 0;

}
```

INPUT:
Enter number of students:2
Enter students 1 details (name roll marks) : pavani 20 88
Enter students 2 details (name roll marks) : najju 21 80

OUTPUT:
Sorted student records by marks:
Name:najju Roll:21 Marks:80.00
Name:pavani Roll:20 marks:88.00



Employee Records

Aim:

Employee Record in Descending Order by Age in Structure

```
#include<stdio.h>
struct employee
{
 char name[30];
 int salary;
 int age;
};
int main()
{
 struct employee e[20], temp;
 int i,j,n;
 printf("Enter n:\n");
 scanf("%d",&n);
 for(i=0;i< n;i++)
 {
  printf("Enter name, salary and age of employee:\n");
  scanf("%s%d%d",e[i].name, &e[i].salary, &e[i].age);
 }
 for(i=0;i< n-1;i++)
```

```
{
 for(j=i+1;j< n;j++)
 {
  if(e[i].age< e[j].age)
  {
   temp = e[i];
   e[i] = e[j];
   e[j] = temp;
  }
 }
}
printf("Sorted records are:\n");
for(i=0;i< n;i++)
{
 printf("Name: %s\n", e[i].name);
 printf("Salary: %d\n", e[i].salary);
 printf("Age: %d\n\n", e[i].age);
}
return 0;
}
```

INPUT:

Enter n:2

Enter name. salary and age of employee: najju 8000 25

Enter name. salary and age of employee: pavani 90000 26

OUTPUT:

Sorted records are:

Name: pavani

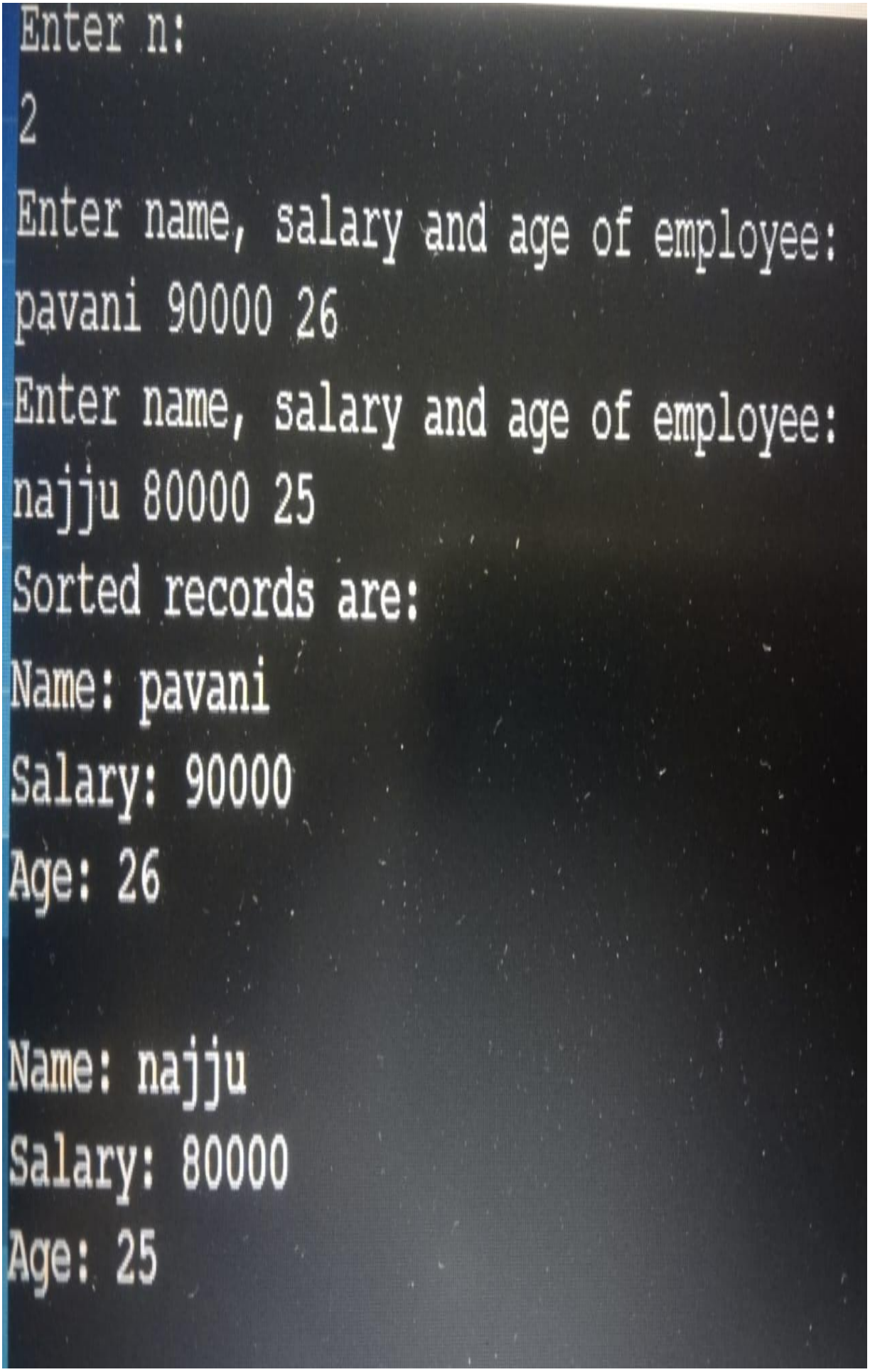
Salary: 90000

Age: 26

Name: najju

Salary:80000

Age:25


Enter n:
2
Enter name, salary and age of employee:
pavani 90000 26
Enter name, salary and age of employee:
najju 80000 25
Sorted records are:
Name: pavani
Salary: 90000
Age: 26
Name: najju
Salary: 80000
Age: 25

Merge Two Arrays

Aim:

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;
}
```

INPUT:

Enter the size of first array: 5

Enter the array elements: 12345

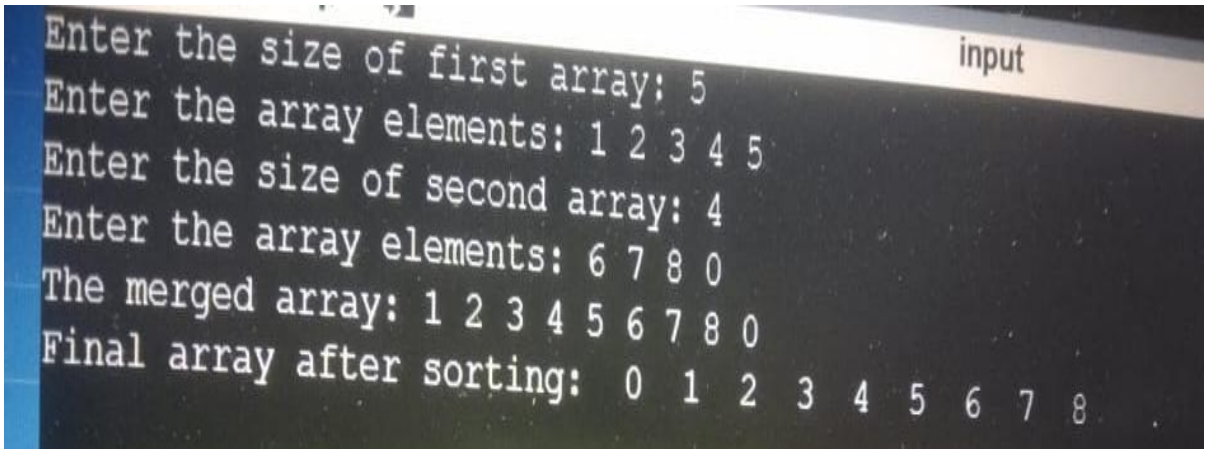
Enter the size of second array: 4

Enter the array elements: 678 0

OUTPUT:

The merged array: 123456780

Final array after sorting: 0 1 2 3 4 5 6 7 8



Perfect Numbers

Aim:

C program to Generate perfect Number in Give Minimum to Maximum Ranges

```
#include <stdio.h>
int main()
{
   int min, max, i, j, sum;
   printf("Enter minimum value: ");
   scanf("%d", &min);
   printf("Enter maximum value: ");
   scanf("%d", &max);
   printf("Perfect numbers between %d and %d:\n", min, max);
   for (i = min; i <= max; i++)
{
      sum = 0;
      for (j = 1; j < i; j++)
   {
```

```
            if (i % j == 0)
            {
                sum += j;
            }
        }
        if (sum == i)
        {
            printf("%d\n", i);
        }
    }
    return 0;
}
```

INPUT:

Enter minimum value:1

Enter maximum value: 1000

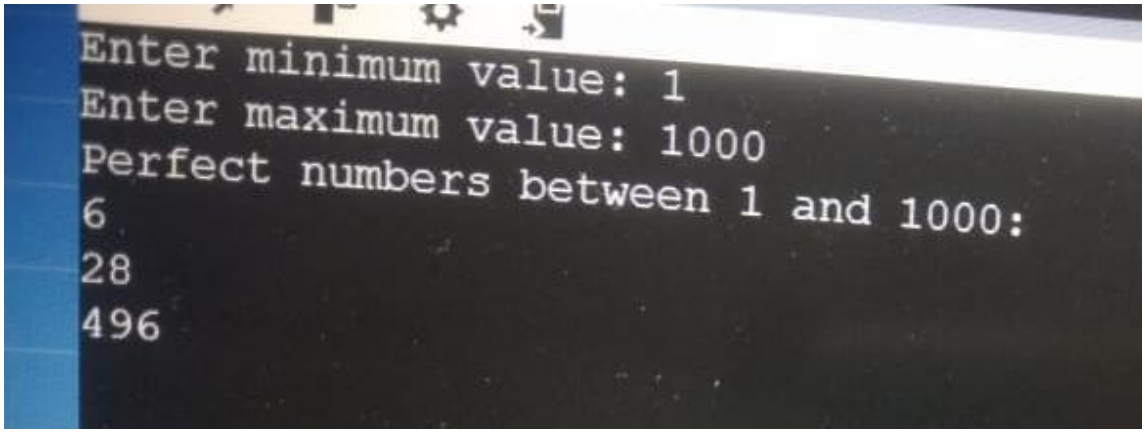
OUTPUT:

Perfect numbers between 1 and 1000:

6

28

496



Grace Marks of Students

Aim:

Write a program which to find the grace marks for a student using switch. The user should enter the class obtained by the student and the number of subjects he has failed in.

- If the student gets first class and the number of subjects he failed in is greater than 3, then he does not get any grace, If the number of subjects he failed in is less than or equal to 3 then the grace is of 5 marks per subject.

- If the student gets second class and the number of subjects he failed in is greater than 2, then he does not get any grace. If the number of subjects he failed in is less than or equal to 2 then the grace is of 4 marks per subject.

- If the student gets third class and the number of subjects he failed in is greater than 1, then he does not get any grace. If the number of subjects he failed in is equal to 1 then the grace is of 5 marks per subject.

```
#include<stdio.h>
#include<conio.h>
int main()
{
   int  grace,n, sub,marks;
   printf("Enter the class obtained by the student: ");
   scanf("%d", &n);
   printf("Enter the number of subject in which he has failed: ");
   scanf("%d", &sub);
```

```
switch(n)
{
   case 1:
      if(sub>3)
         grace=0;
      else
          grace=5;
   break;
   case 2:
      if(sub>2)
          grace =0;
      else
         grace =4;
   break;
  case 3:
      if(sub>1)
         grace = 0;
      else
         grace=5;
   break;
  default:
  printf("invalid input");
```

```
    }
    marks= sub* grace;
    printf("the total marks given is %d",marks);
}
```

INPUT:

Enter the class obtained by the student: 1

Enter the number of subject in which he has failed: 2

OUTPUT:

the total marks given is 10