A Micro Project Report

on

Problem Solving using C Language

Submitted by

AVULA MANIKANTA (23471A05BB)

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 AVULA MANIKANTA , Roll No: 23471A05BB, 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

S.No	Description
1.	C Program to Find second and Third Smallest Element from Array.
2.	C Program to Insert Number in Given Position in Array.
3.	C Program to Merge Two Arrays
4.	C Program to Find Standard Deviation.

second and Third Smallest Element from Array

AIM:

C Program to Find second and Third Smallest Element from Array.

Source code:

```
#include <stdio.h>
#include <limits.h>

void findSecondAndThirdSmallest(int arr[], int n) {
    if (n < 3) {
        printf("Array must have at least 3 elements.\n");
        return;
    }

    int first = INT_MAX, second = INT_MAX, third = INT_MAX;
    for (int i = 0; i < n; i++) {
        if (arr[i] < first) {
            third = second;
            second = first;
            first = arr[i];
        } else if (arr[i] < second && arr[i] != first) {
            third = second;
            second = arr[i];
        } else if (arr[i] < third && arr[i] != first && arr[i] != second)
{
            third = arr[i];
```

```
        }
    }

    if (second == INT_MAX || third == INT_MAX) {
        printf("Second or third smallest element does not exist.\n");
    } else {
        printf("The second smallest element is: %d\n", second);
        printf("The third smallest element is: %d\n", third);
    }
}

int main() {
    int arr[] = {12, 13, 1, 10, 34, 1};
    int n = sizeof(arr) / sizeof(arr[0]);

    findSecondAndThirdSmallest(arr, n);

    return 0;
}
```

Output:

The second smallest element is: 10

The third smallest element is: 12

Insert Number in Given Position in Array

AIM:

.C Program to Insert Number in Given Position in Array.

Source code:

```
#include <stdio.h>


int main() {
   int arr[100], n, pos, value;
   printf("Enter the number of elements in the array: ");
   scanf("%d", &n);
   printf("Enter %d elements: ", n);
   for(int i = 0; i < n; i++) {
      scanf("%d", &arr[i]);
   }
   printf("Enter the position to insert (1 to %d): ", n+1);
   scanf("%d", &pos);
   printf("Enter the value to insert: ");
   scanf("%d", &value);
   if(pos < 1 || pos > n+1) {
      printf("Invalid position!\n");
   } else {
      for(int i = n; i >= pos; i--) {
```

```
            arr[i] = arr[i - 1];
        }
        arr[pos - 1] = value;
        n++;
        printf("Array after insertion: ");
        for(int i = 0; i < n; i++) {
            printf("%d ", arr[i]);
        }
        printf("\n");
    }

    return 0;
}
```

Output:

Enter the number of elements in the array: 5

Enter 5 elements: 1 2 3 4 5

Enter the position to insert (1 to 6): 3

Enter the value to insert: 10

Array after insertion: 1 2 10 3 4 5

Merge Two Arrays

AIM:

C Program to Merge Two Arrays

Source code:

```
#include <stdio.h>


int main() {
   int arr1[50], arr2[50], merged[100];
   int n1, n2, i, j;
   printf("Enter the number of elements in the first array: ");
   scanf("%d", &n1);
   printf("Enter %d elements for the first array: ", n1);
   for(i = 0; i < n1; i++) {
      scanf("%d", &arr1[i]);
   }
   printf("Enter the number of elements in the second array: ");
   scanf("%d", &n2);
   printf("Enter %d elements for the second array: ", n2);
   for(i = 0; i < n2; i++) {
      scanf("%d", &arr2[i]);
   }
   for(i = 0; i < n1; i++) {
```

```
        merged[i] = arr1[i];
    }
    for(j = 0; j < n2; j++) {
        merged[i + j] = arr2[j];
    }
    printf("Merged array: ");
    for(i = 0; i < n1 + n2; i++) {
        printf("%d ", merged[i]);
    }
    printf("\n");

    return 0;
}
```

Output:

Enter the number of elements in the first array: 3

Enter 3 elements for the first array: 1 2 3

Enter the number of elements in the second array: 2

Enter 2 elements for the second array: 4 5

Merged array: 1 2 3 4 5

Standard Deviation.

C Program to Find Standard Deviation.

Source code:

```
#include <stdio.h>
#include <math.h>

int main() {
   int n;
   float data[100], mean = 0.0, sum = 0.0, standardDeviation = 0.0;
   printf("Enter the number of elements: ");
   scanf("%d", &n);
   printf("Enter %d elements: ", n);
   for(int i = 0; i < n; i++) {
      scanf("%f", &data[i]);
      sum += data[i];
   }
   mean = sum / n;
   for(int i = 0; i < n; i++) {
      standardDeviation += pow(data[i] - mean, 2);
   }
   standardDeviation = sqrt(standardDeviation / n);
```

```
    printf("Standard Deviation = %.2f\n", standardDeviation);


    return 0;
}
```

Output :

Enter the number of elements: 5

Enter 5 elements: 10 12 23 23 16

Standard Deviation = 5.57