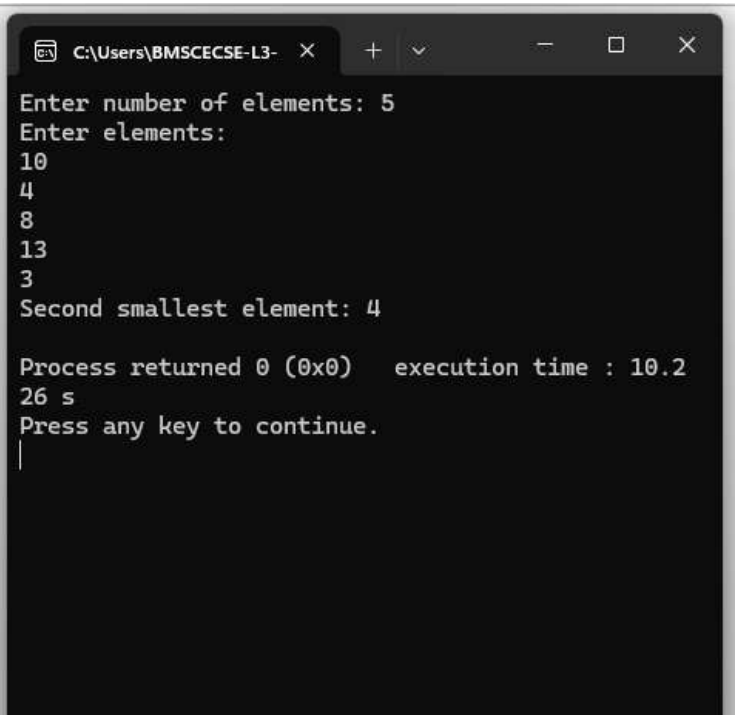
```c
/*#include <stdio.h>

void main() {
    int n, i;
    printf("Enter number of elements: ");
    scanf("%d", &n);

    int arr[n];
    printf("Enter elements:\n");
    for(i = 0; i < n; i++) {
        scanf("%d", &arr[i]);
    }

    int smallest = arr[0], sec= arr[0];

    for(i = 1; i < n; i++) {
        if(arr[i] < smallest) {
            sec = smallest;
            smallest = arr[i];
        } else if(arr[i] < sec && arr[i] != smallest) {
            sec = arr[i];
        }
    }

    printf("Second smallest element: %d\n", sec);
}
*/
```



```c
#include <stdio.h>

int main() {
    int n, i, j, sum = 0;

    printf("Enter order of square matrix: ");
    scanf("%d", &n);

    int matrix[n][n];

    printf("Enter matrix elements:\n");
    for(i = 0; i < n; i++) {
        for(j = 0; j < n; j++) {
            scanf("%d", &matrix[i][j]);
        }
    }

    for(i = 0; i < n; i++) {
        sum += matrix[i][i];
    }

    printf("Sum of left diagonal: %d\n", sum);
    return 0;
}
```

```
C:\Users\BMSCECSE-L3-
Enter order of square matrix: 3
Enter matrix elements:
1
0
4
5
2
3
2
3
6
Sum of left diagonal: 9

Process returned 0 (0x0)   execution time : 13.0
75 s
Press any key to continue.
```

Start here | array.c

```c

int main() {
    int rows, cols, i, j;

    printf("Enter number of rows and columns: ");
    scanf("%d %d", &rows, &cols);

    int matrix[rows][cols];

    printf("Enter matrix elements:\n");
    for(i = 0; i < rows; i++) {
        for(j = 0; j < cols; j++) {
            scanf("%d", &matrix[i][j]);
        }
    }

    for(i = 0; i < rows; i++) {
        int rowSum = 0;
        for(j = 0; j < cols; j++) {
            rowSum += matrix[i][j];
        }
        printf("Sum of row %d: %d\n", i + 1, rowSum);
    }

    for(j = 0; j < cols; j++) {
        int colSum = 0;
        for(i = 0; i < rows; i++) {
            colSum += matrix[i][j];
        }
        printf("Sum of column %d: %d\n", j + 1, colSum);
    }

    return 0;
}

```

```
C:\Users\BMSCECSE-L3-
Enter number of rows and columns: 2
2
Enter matrix elements:
2
3
6
0
Sum of row 1: 5
Sum of row 2: 6
Sum of column 1: 8
Sum of column 2: 3

Process returned 0 (0x0)   execution time : 15.8
14 s
Press any key to continue.
```

here × array.c ×

```c
#include <stdio.h>

int main() {
    int n, i, j, count = 0;

    printf("Enter number of elements: ");
    scanf("%d", &n);

    int arr[n];
    int visited[n];

    printf("Enter elements:\n");
    for(i = 0; i < n; i++) {
        scanf("%d", &arr[i]);
        visited[i] = 0;
    }

    for(i = 0; i < n; i++) {
        if(visited[i] == 1)
            continue;

        int duplicate = 0;
        for(j = i + 1; j < n; j++) {
            if(arr[i] == arr[j]) {
                visited[j] = 1;
                duplicate = 1;
            }
        }

        if(duplicate)
            count++;
    }

    printf("Total number of duplicate elements: %d\n", count);
    return 0;
}
```

```
C:\Users\BMSCECSE-L3-35\Dc
Enter number of elements: 5
Enter elements:
2
3
3
5
6
Total number of duplicate elements: 1

Process returned 0 (0x0)   execution time : 11.898 s
Press any key to continue.
```