Reg. No: 21BCE1297 **Name: Vidhi Shah** **Date: 08/03/22**

PPS5

Q1

Aim:

Write a 'C' program to sort the array of n elements using Selection sort. Get the user input for 'n'.

Procedure:

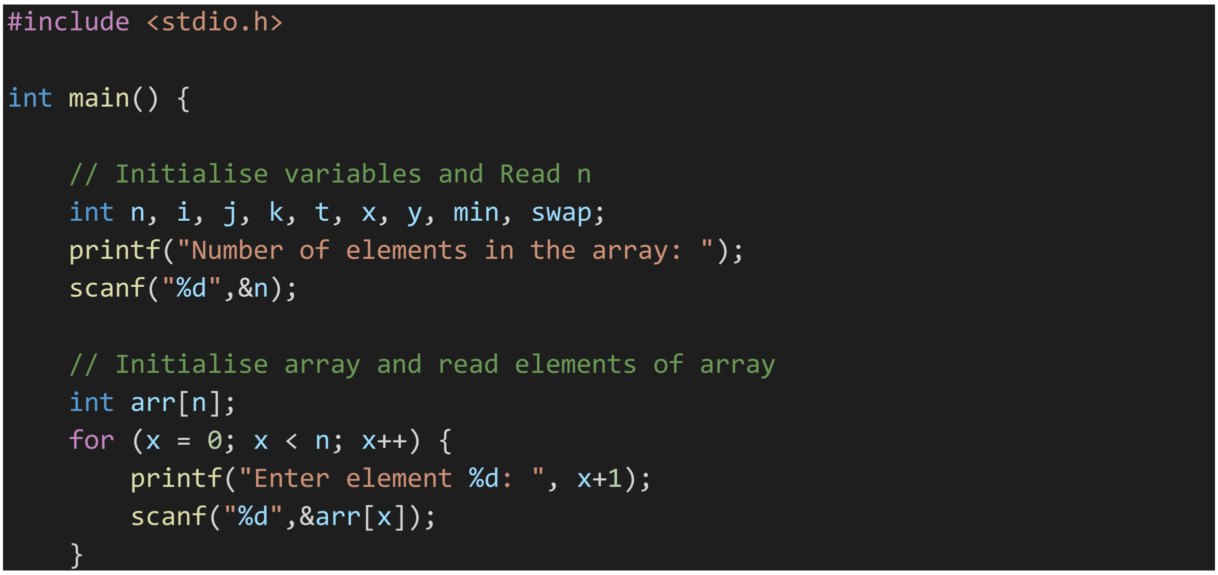
Input:
Number of elements, n
Next n lines contain n numbers

Output:
Sorted Array

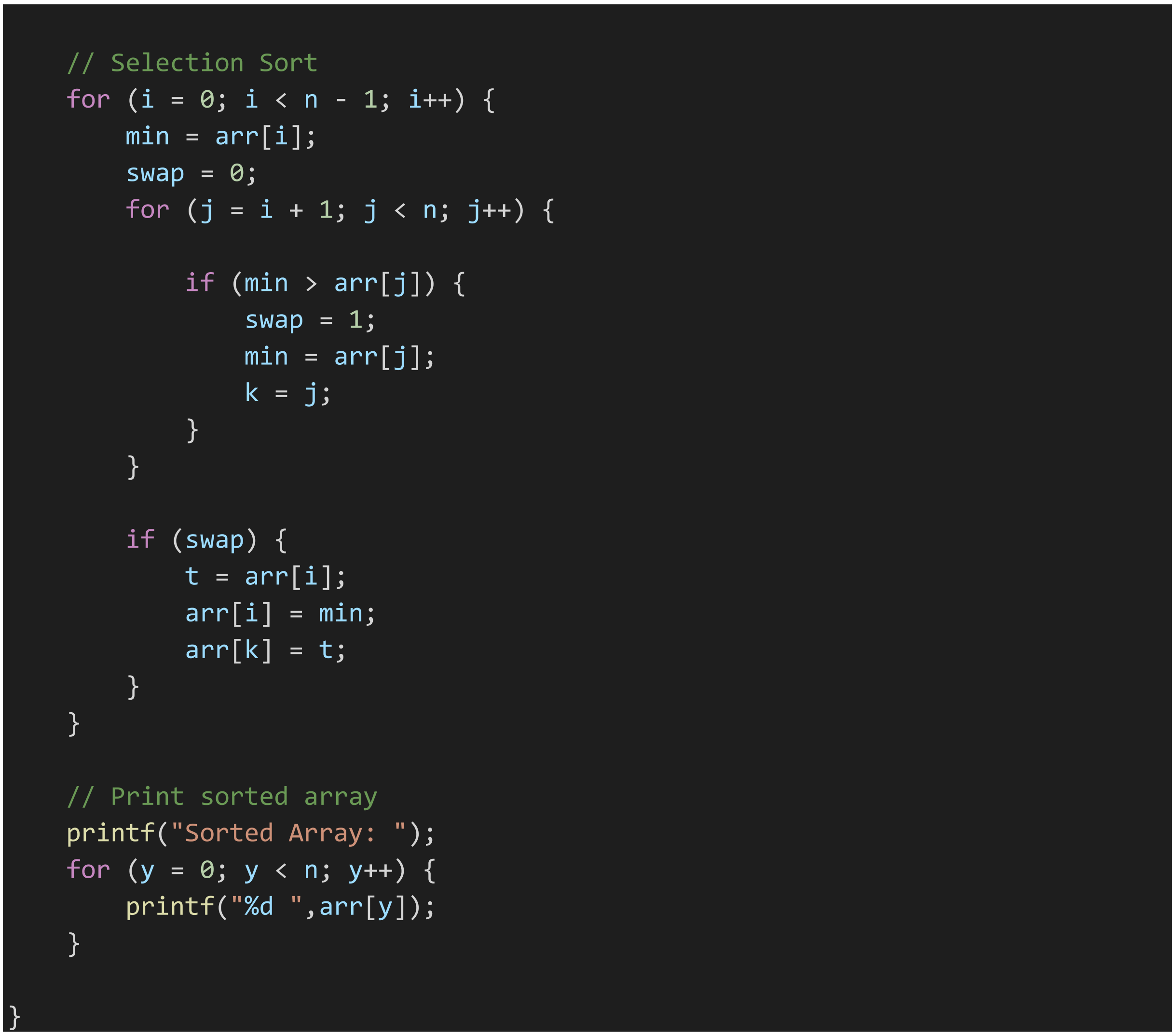
Algorithm:

```
Step 1: Read n
Step 2: Initialise array of size n
Step 3: Use for loop to read elements in the array
Step 4: Selection Sort Algorithm:
        Step A: Iterate through n-1 elements using counter i
        Step B: For each iteration set min to ith element of the array
        Step C: Check if current element is greater than subsequent elements
                If yes, set min to the subsequent element and store the index of that element
        Step D: If min is less than initial element, swap the elements
Step 5: Display sorted Array
```

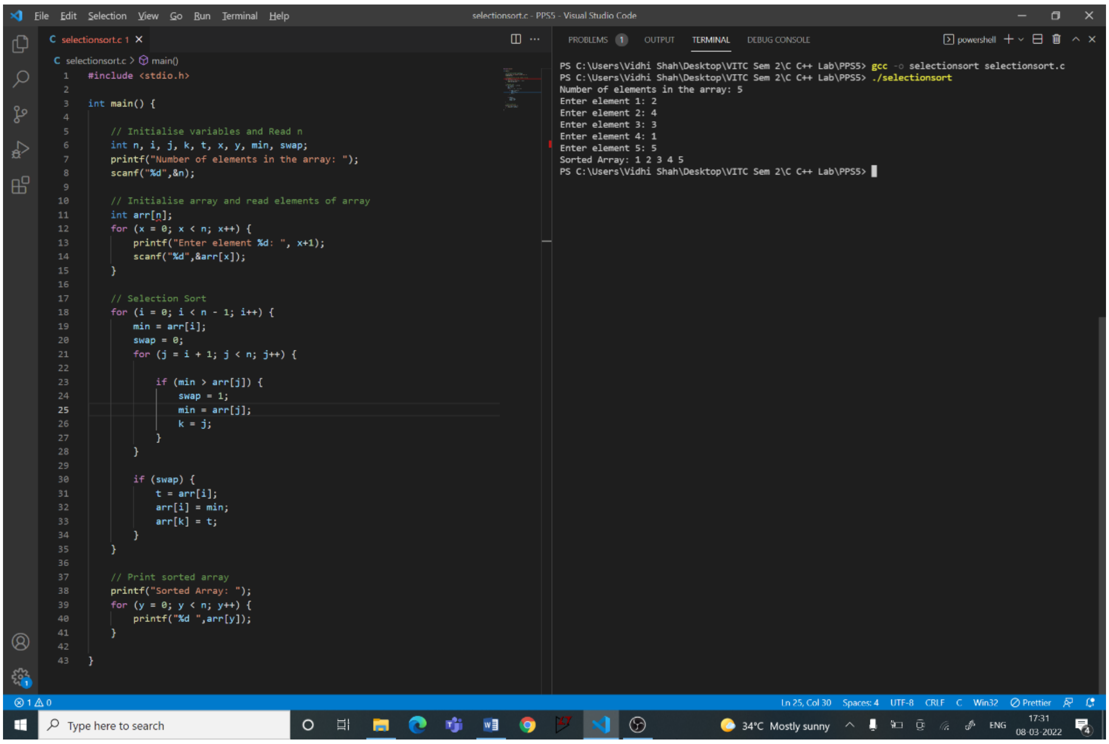
Code:



```c
#include <stdio.h>

int main() {

    // Initialise variables and Read n
    int n, i, j, k, t, x, y, min, swap;
    printf("Number of elements in the array: ");
    scanf("%d",&n);

    // Initialise array and read elements of array
    int arr[n];
    for (x = 0; x < n; x++) {
        printf("Enter element %d: ", x+1);
        scanf("%d",&arr[x]);
    }
```



```c
    // Selection Sort
    for (i = 0; i < n - 1; i++) {
        min = arr[i];
        swap = 0;
        for (j = i + 1; j < n; j++) {

            if (min > arr[j]) {
                swap = 1;
                min = arr[j];
                k = j;
            }
        }

        if (swap) {
            t = arr[i];
            arr[i] = min;
            arr[k] = t;
        }
    }

    // Print sorted array
    printf("Sorted Array: ");
    for (y = 0; y < n; y++) {
        printf("%d ",arr[y]);
    }

}
```



```c
#include <stdio.h>

int main() {

    // Initialise variables and Read n
    int n, i, j, k, t, x, y, min, swap;
    printf("Number of elements in the array: ");
    scanf("%d",&n);

    // Initialise array and read elements of array
    int arr[n];
    for (x = 0; x < n; x++) {
        printf("Enter element %d: ", x+1);
        scanf("%d",&arr[x]);
    }

    // Selection Sort
    for (i = 0; i < n - 1; i++) {
        min = arr[i];
        swap = 0;
        for (j = i + 1; j < n; j++) {

            if (min > arr[j]) {
                swap = 1;
                min = arr[j];
                k = j;
            }
        }

        if (swap) {
            t = arr[i];
            arr[i] = min;
            arr[k] = t;
        }
    }

    // Print sorted array
    printf("Sorted Array: ");
    for (y = 0; y < n; y++) {
        printf("%d ",arr[y]);
    }

}
```

```
PS C:\Users\Vidhi Shah\Desktop\VITC Sem 2\C C++ Lab\PPS5> gcc -o selectionsort selectionsort.c
PS C:\Users\Vidhi Shah\Desktop\VITC Sem 2\C C++ Lab\PPS5> ./selectionsort
Number of elements in the array: 5
Enter element 1: 2
Enter element 2: 4
Enter element 3: 3
Enter element 4: 1
Enter element 5: 5
Sorted Array: 1 2 3 4 5
PS C:\Users\Vidhi Shah\Desktop\VITC Sem 2\C C++ Lab\PPS5> 
```

Q2

Aim:

Write a ‘C’ program to eliminate the duplicate elements from an array of n elements. Get the user input for ‘n’.

Procedure:

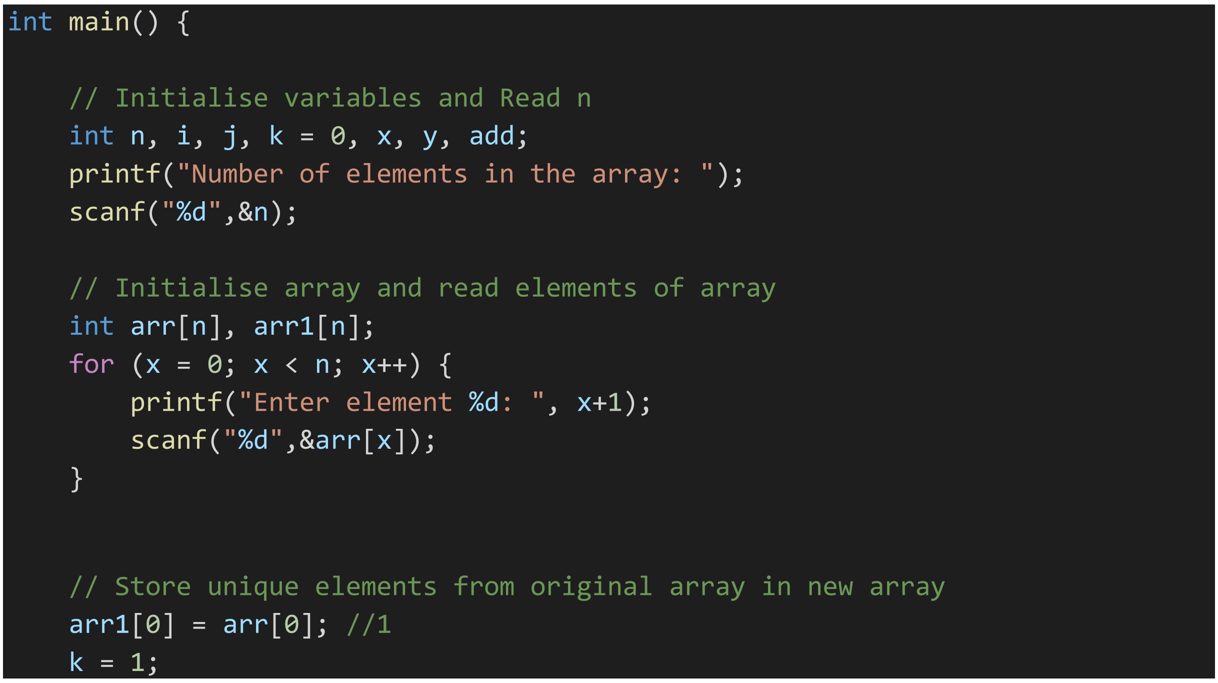
Input:
Number of elements, n
Next n lines contain n numbers

Output:
Array with no duplicate elements

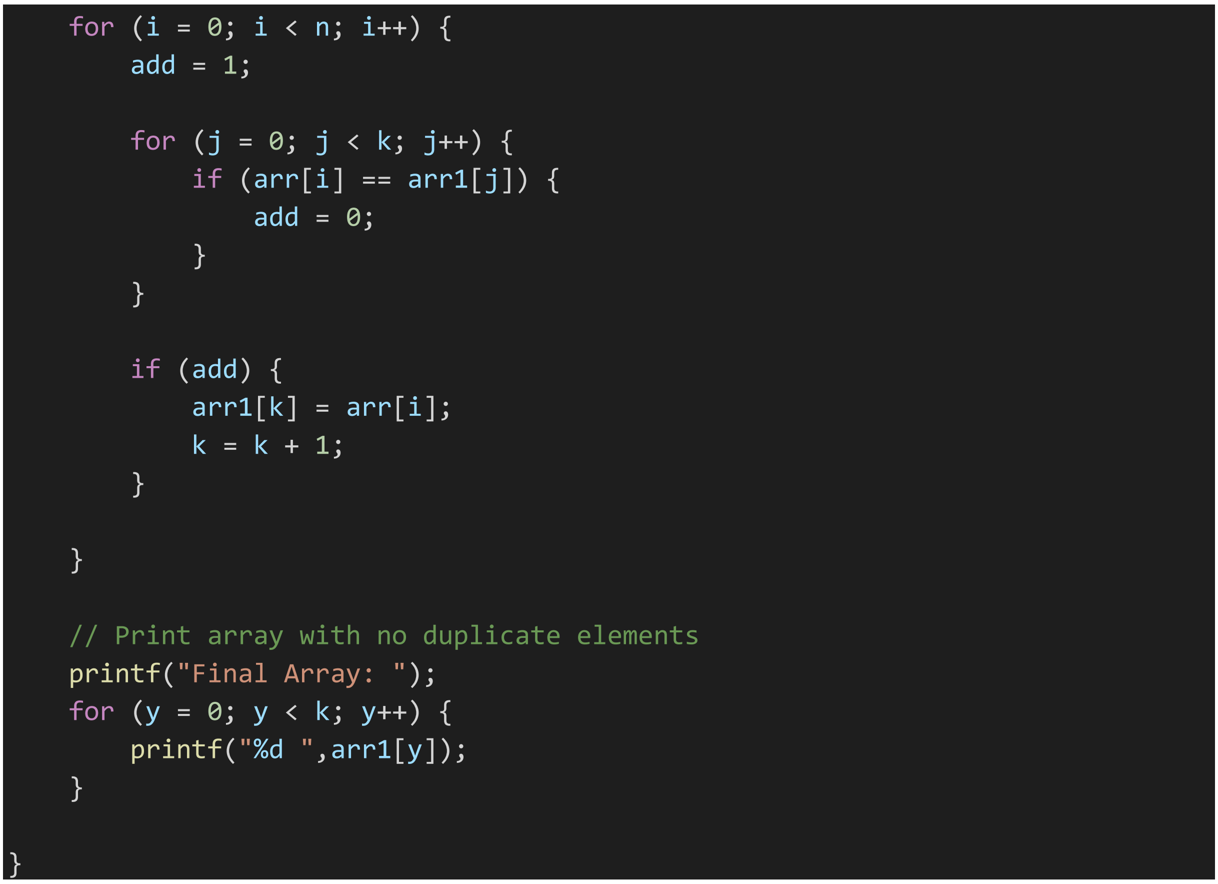
Algorithm:

```
Step 1: Read n
Step 2: Initialise original array (arr) and new array (arr1) of size n
Step 3: Use for loop to read elements in the arr
Step 4: Elimination of duplicate elements algorithm:
        Step A: Assign first element of arr as first element of arr1, set k to 1
        Step B: Iterate through n elements of arr
        Step C: For each iteration set add to 1
        Step D: Check if current element of arr is equal to any element of arr1
                If yes, set add to 0
        Step E: If add is 1, add the current element of arr to arr 1 at $k^{th}$ position, k = k + 1
Step 5: Display new array with no duplicate elements
```

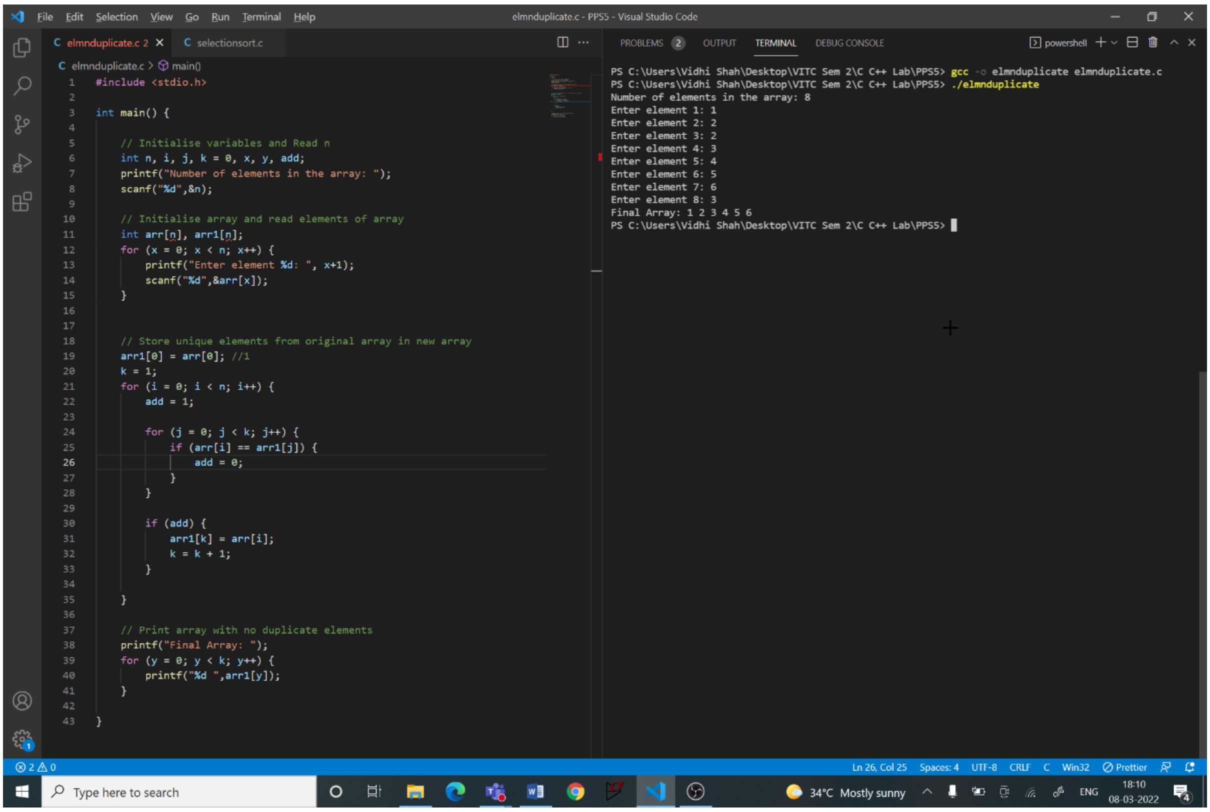
Code:



```c
int main() {

    // Initialise variables and Read n
    int n, i, j, k = 0, x, y, add;
    printf("Number of elements in the array: ");
    scanf("%d",&n);

    // Initialise array and read elements of array
    int arr[n], arr1[n];
    for (x = 0; x < n; x++) {
        printf("Enter element %d: ", x+1);
        scanf("%d",&arr[x]);
    }


    // Store unique elements from original array in new array
    arr1[0] = arr[0]; //1
    k = 1;
```



```
    for (i = 0; i < n; i++) {
        add = 1;

        for (j = 0; j < k; j++) {
            if (arr[i] == arr1[j]) {
                add = 0;
            }
        }

        if (add) {
            arr1[k] = arr[i];
            k = k + 1;
        }

    }

    // Print array with no duplicate elements
    printf("Final Array: ");
    for (y = 0; y < k; y++) {
        printf("%d ",arr1[y]);
    }

}
```



```
#include <stdio.h>

int main() {

    // Initialise variables and Read n
    int n, i, j, k = 0, x, y, add;
    printf("Number of elements in the array: ");
    scanf("%d",&n);

    // Initialise array and read elements of array
    int arr[n], arr1[n];
    for (x = 0; x < n; x++) {
        printf("Enter element %d: ", x+1);
        scanf("%d",&arr[x]);
    }


    // Store unique elements from original array in new array
    arr1[0] = arr[0]; //1
    k = 1;
    for (i = 0; i < n; i++) {
        add = 1;

        for (j = 0; j < k; j++) {
            if (arr[i] == arr1[j]) {
                add = 0;
            }
        }

        if (add) {
            arr1[k] = arr[i];
            k = k + 1;
        }

    }

    // Print array with no duplicate elements
    printf("Final Array: ");
    for (y = 0; y < k; y++) {
        printf("%d ",arr1[y]);
    }

}
```

```
PS C:\Users\Vidhi Shah\Desktop\VITC Sem 2\C C++ Lab\PPS5> gcc -o elmnduplicate elmnduplicate.c
PS C:\Users\Vidhi Shah\Desktop\VITC Sem 2\C C++ Lab\PPS5> ./elmnduplicate
Number of elements in the array: 8
Enter element 1: 1
Enter element 2: 2
Enter element 3: 2
Enter element 4: 3
Enter element 5: 4
Enter element 6: 5
Enter element 7: 6
Enter element 8: 3
Final Array: 1 2 3 4 5 6
PS C:\Users\Vidhi Shah\Desktop\VITC Sem 2\C C++ Lab\PPS5> 
```

Q3

Aim:

Write a ‘C’ program perform matrix addition for n x n matrix. Get the user input for ‘n’.

Procedure:

Input:

Number of rows and columns, n
Matrix A and Matrix B with n*n elements

Output:

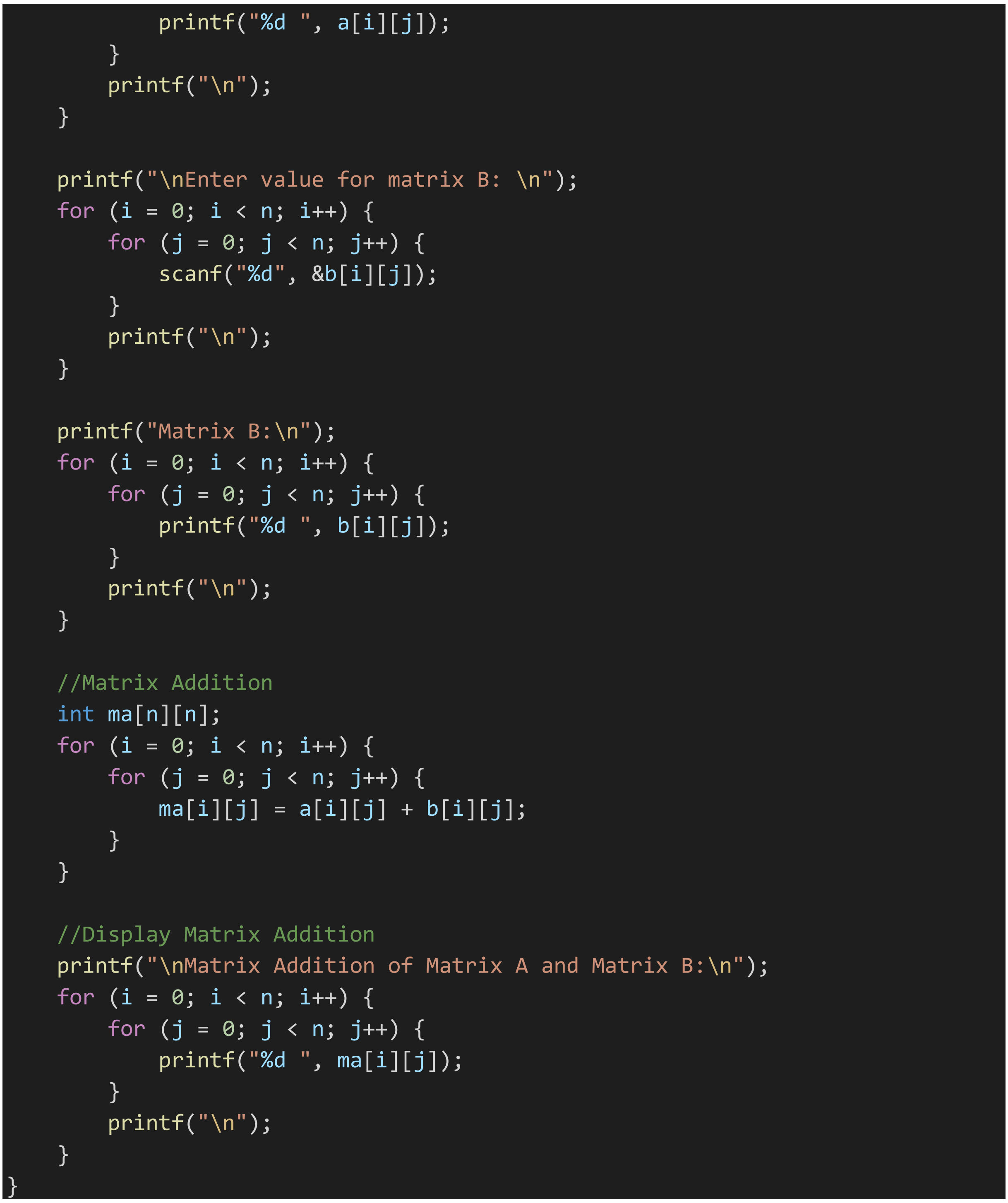
Matrix which is addition of Matrix A and Matrix B

Algorithm:

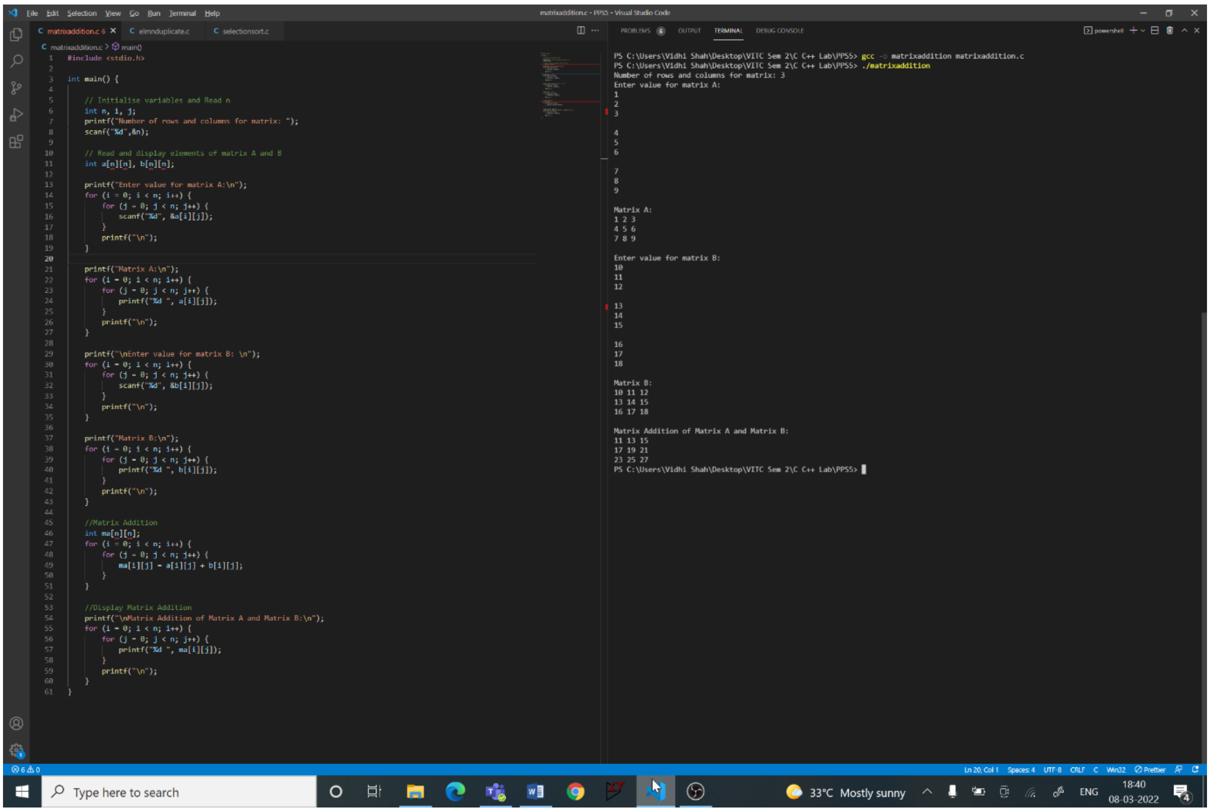
```
Step 1: Read n
Step 2: Initialise matrix A and matrix B of size n*n
Step 3: Use nested for loop to read elements in the matrix A and matrix B
Step 4: Use nested for loop to print elements of matrix A and matrix B
Step 5: Initialise addition matrix, MA of size n*n
Step 5: Use nested for loops with counter i and j
        MA[i][j] = A[i][j] + B[i][j]
Step 6: Display addition matrix using nested for loops
```

Code:

```c
#include <stdio.h>

int main() {

    // Initialise variables and Read n
    int n, i, j;
    printf("Number of rows and columns for matrix: ");
    scanf("%d",&n);

    // Read and display elements of matrix A and B
    int a[n][n], b[n][n];

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

    printf("Matrix A:\n");
    for (i = 0; i < n; i++) {
        for (j = 0; j < n; j++) {
```



```c
            printf("%d ", a[i][j]);
        }
        printf("\n");
    }

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

    printf("Matrix B:\n");
    for (i = 0; i < n; i++) {
        for (j = 0; j < n; j++) {
            printf("%d ", b[i][j]);
        }
        printf("\n");
    }

    //Matrix Addition
    int ma[n][n];
    for (i = 0; i < n; i++) {
        for (j = 0; j < n; j++) {
            ma[i][j] = a[i][j] + b[i][j];
        }
    }

    //Display Matrix Addition
    printf("\nMatrix Addition of Matrix A and Matrix B:\n");
    for (i = 0; i < n; i++) {
        for (j = 0; j < n; j++) {
            printf("%d ", ma[i][j]);
        }
        printf("\n");
    }
}
```


File Edit Selection View Go Run Terminal Help
matrixaddition.c - PPS5 - Visual Studio Code
matrixaddition.c 6
elmnduplicate.c
selectionsort.c
matrixaddition.c > main()
PROBLEMS 6
OUTPUT
TERMINAL
DEBUG CONSOLE
powershell
Ln 20, Col 1 Spaces: 4 UTF-8 CRLF C Win32 Prettier
Type here to search
33°C Mostly sunny
ENG
18:40
08-03-2022

```c
#include <stdio.h>

int main() {

    // Initialise variables and Read n
    int n, i, j;
    printf("Number of rows and columns for matrix: ");
    scanf("%d",&n);

    // Read and display elements of matrix A and B
    int a[n][n], b[n][n];

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

    printf("Matrix A:\n");
    for (i = 0; i < n; i++) {
        for (j = 0; j < n; j++) {
            printf("%d ", a[i][j]);
        }
        printf("\n");
    }

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

    printf("Matrix B:\n");
    for (i = 0; i < n; i++) {
        for (j = 0; j < n; j++) {
            printf("%d ", b[i][j]);
        }
        printf("\n");
    }

    //Matrix Addition
    int ma[n][n];
    for (i = 0; i < n; i++) {
        for (j = 0; j < n; j++) {
            ma[i][j] = a[i][j] + b[i][j];
        }
    }

    //Display Matrix Addition
    printf("\nMatrix Addition of Matrix A and Matrix B:\n");
    for (i = 0; i < n; i++) {
        for (j = 0; j < n; j++) {
            printf("%d ", ma[i][j]);
        }
        printf("\n");
    }
}
```

```
PS C:\Users\Vidhi Shah\Desktop\VITC Sem 2\C C++ Lab\PPS5> gcc -o matrixaddition matrixaddition.c
PS C:\Users\Vidhi Shah\Desktop\VITC Sem 2\C C++ Lab\PPS5> ./matrixaddition
Number of rows and columns for matrix: 3
Enter value for matrix A:
1
2
3

4
5
6

7
8
9

Matrix A:
1 2 3
4 5 6
7 8 9

Enter value for matrix B:
10
11
12

13
14
15

16
17
18

Matrix B:
10 11 12
13 14 15
16 17 18

Matrix Addition of Matrix A and Matrix B:
11 13 15
17 19 21
23 25 27
PS C:\Users\Vidhi Shah\Desktop\VITC Sem 2\C C++ Lab\PPS5>
```