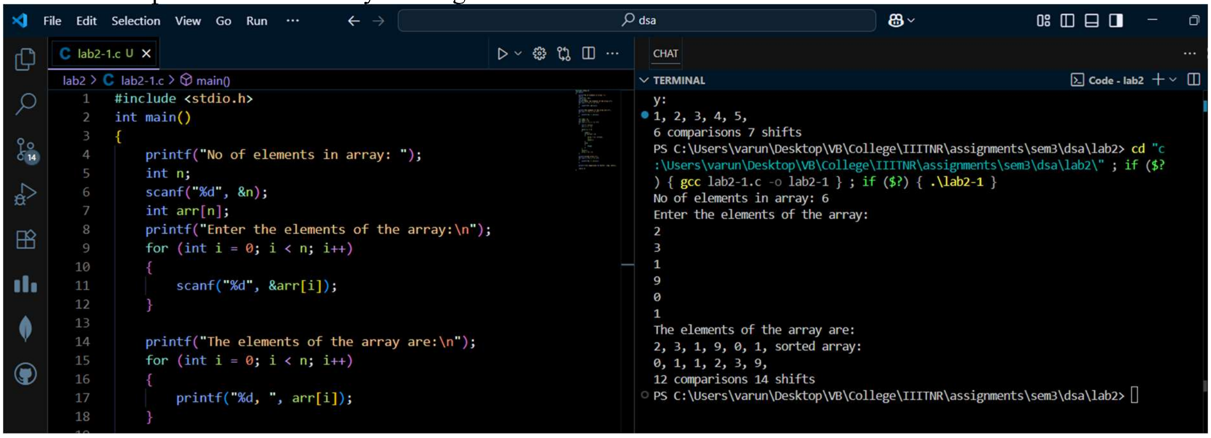
VARUN BHARGAVA – 241010282
DATA STRUCTURES TASK-2

Task 1: Insertion Sort:

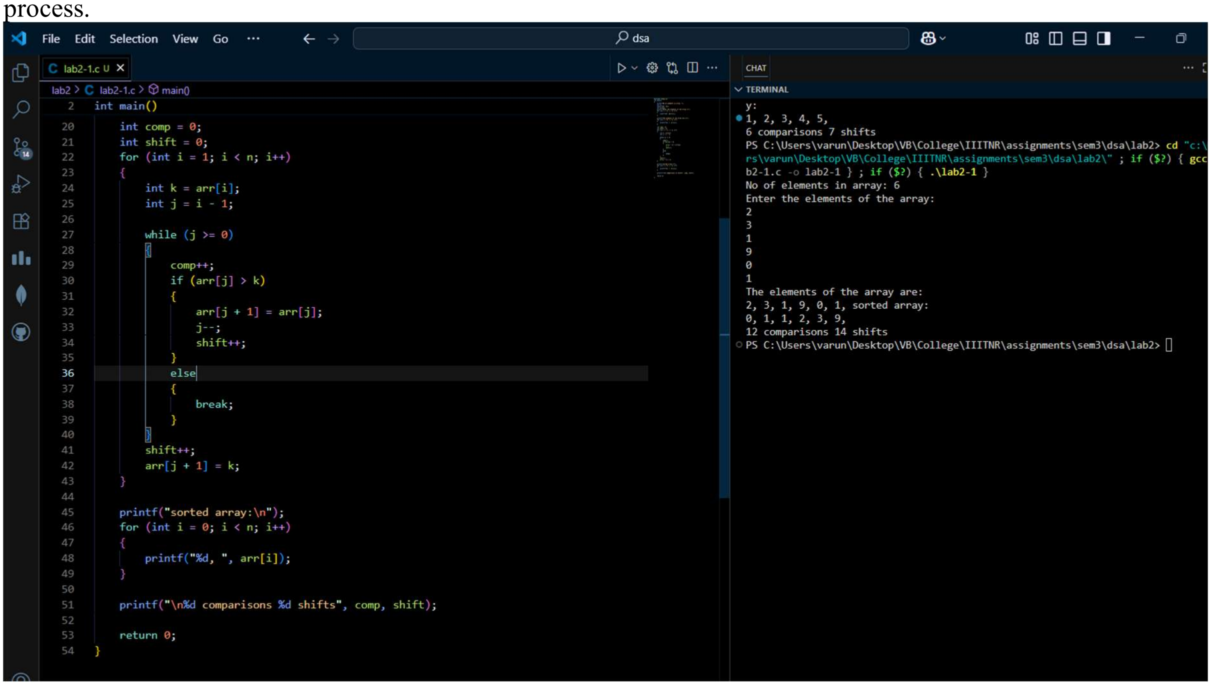
(https://github.com/varunnnb/dsa-sem3-iiitnr/blob/main/lab2/lab2-1.c)
Write a program to perform the following operations using Insertion Sort:
• Take user input to create an array of integers.



```c
#include <stdio.h>
int main()
{
    printf("No of elements in array: ");
    int n;
    scanf("%d", &n);
    int arr[n];
    printf("Enter the elements of the array:\n");
    for (int i = 0; i < n; i++)
    {
        scanf("%d", &arr[i]);
    }

    printf("The elements of the array are:\n");
    for (int i = 0; i < n; i++)
    {
        printf("%d, ", arr[i]);
    }
```

```
y:
1, 2, 3, 4, 5,
6 comparisons 7 shifts
PS C:\Users\varun\Desktop\VB\College\IIITNR\assignments\sem3\dsa\lab2> cd "c
:\Users\varun\Desktop\VB\College\IIITNR\assignments\sem3\dsa\lab2\" ; if ($?
) { gcc lab2-1.c -o lab2-1 } ; if ($?) { .\lab2-1 }
No of elements in array: 6
Enter the elements of the array:
2
3
1
9
0
1
The elements of the array are:
2, 3, 1, 9, 0, 1, sorted array:
0, 1, 1, 2, 3, 9,
12 comparisons 14 shifts
PS C:\Users\varun\Desktop\VB\College\IIITNR\assignments\sem3\dsa\lab2> 
```

• Sort the array using Insertion Sort in ascending order.
• Display the sorted array.
• Track and display the number of comparisons and shifts performed during the sorting process.



```c
int main()
    int comp = 0;
    int shift = 0;
    for (int i = 1; i < n; i++)
    {
        int k = arr[i];
        int j = i - 1;

        while (j >= 0)
        {
            comp++;
            if (arr[j] > k)
            {
                arr[j + 1] = arr[j];
                j--;
                shift++;
            }
            else
            {
                break;
            }
        }
        shift++;
        arr[j + 1] = k;
    }

    printf("sorted array:\n");
    for (int i = 0; i < n; i++)
    {
        printf("%d, ", arr[i]);
    }

    printf("\n%d comparisons %d shifts", comp, shift);

    return 0;
}
```

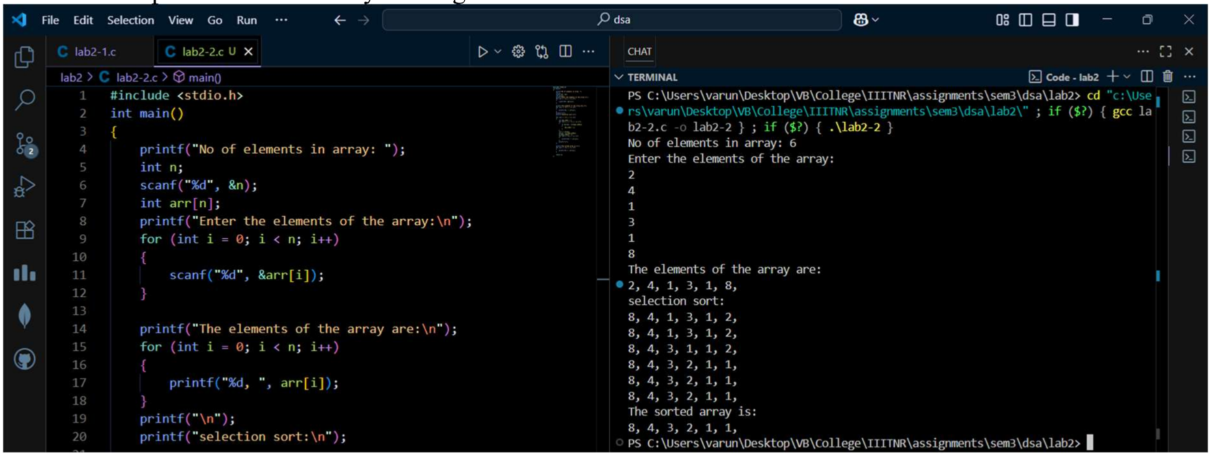
```
y:
1, 2, 3, 4, 5,
6 comparisons 7 shifts
PS C:\Users\varun\Desktop\VB\College\IIITNR\assignments\sem3\dsa\lab2> cd "c:\
rs\varun\Desktop\VB\College\IIITNR\assignments\sem3\dsa\lab2\" ; if ($?) { gcc
b2-1.c -o lab2-1 } ; if ($?) { .\lab2-1 }
No of elements in array: 6
Enter the elements of the array:
2
3
1
9
0
1
The elements of the array are:
2, 3, 1, 9, 0, 1, sorted array:
0, 1, 1, 2, 3, 9,
12 comparisons 14 shifts
PS C:\Users\varun\Desktop\VB\College\IIITNR\assignments\sem3\dsa\lab2> 
```

Task 2: Selection Sort:

(https://github.com/varunnnb/dsa-sem3-iiitnr/blob/main/lab2/lab2-2.c)

Write a program to perform the following operations using Selection Sort:

• Take user input to create an array of integers.



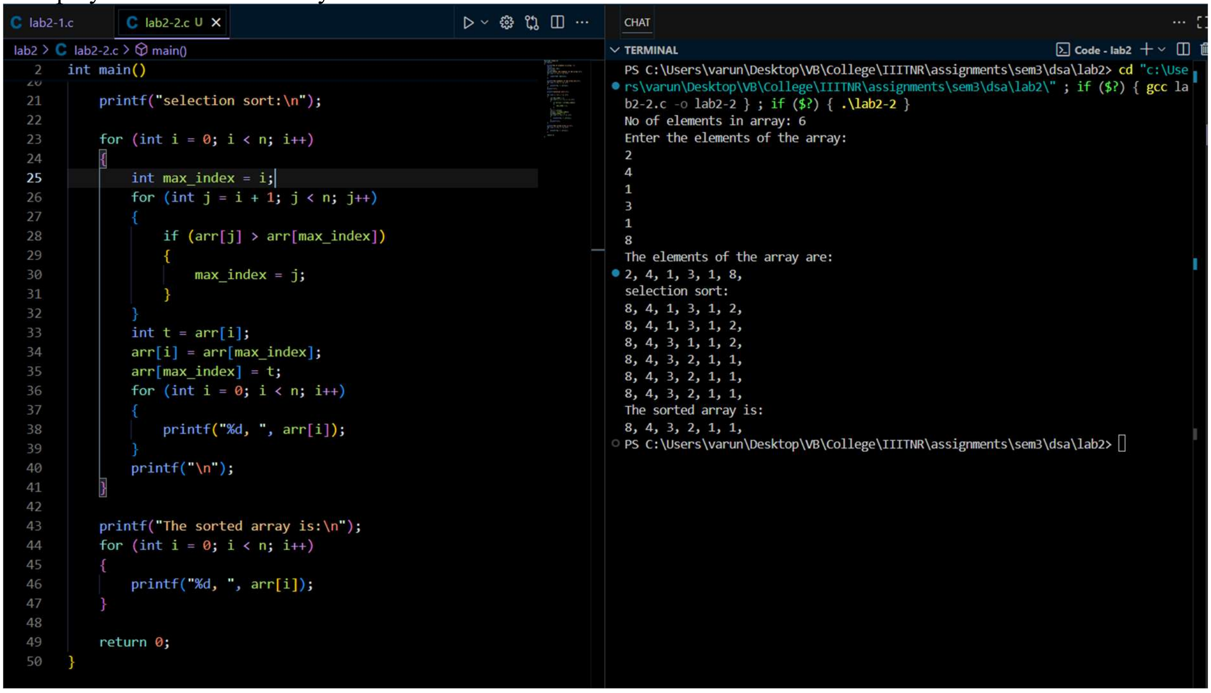
```c
#include <stdio.h>
int main()
{
    printf("No of elements in array: ");
    int n;
    scanf("%d", &n);
    int arr[n];
    printf("Enter the elements of the array:\n");
    for (int i = 0; i < n; i++)
    {
        scanf("%d", &arr[i]);
    }

    printf("The elements of the array are:\n");
    for (int i = 0; i < n; i++)
    {
        printf("%d, ", arr[i]);
    }
    printf("\n");
    printf("selection sort:\n");
```

```
PS C:\Users\varun\Desktop\VB\College\IIITNR\assignments\sem3\dsa\lab2> cd "c:\Use
rs\varun\Desktop\VB\College\IIITNR\assignments\sem3\dsa\lab2\" ; if ($?) { gcc la
b2-2.c -o lab2-2 } ; if ($?) { .\lab2-2 }
No of elements in array: 6
Enter the elements of the array:
2
4
1
3
1
8
The elements of the array are:
2, 4, 1, 3, 1, 8,
selection sort:
8, 4, 1, 3, 1, 2,
8, 4, 1, 3, 1, 2,
8, 4, 3, 1, 1, 2,
8, 4, 3, 2, 1, 1,
8, 4, 3, 2, 1, 1,
8, 4, 3, 2, 1, 1,
The sorted array is:
8, 4, 3, 2, 1, 1,
PS C:\Users\varun\Desktop\VB\College\IIITNR\assignments\sem3\dsa\lab2>
```

• Sort the array using Selection Sort in descending order.

• Print the array after each pass to show the intermediate steps of sorting.

• Display the final sorted array.



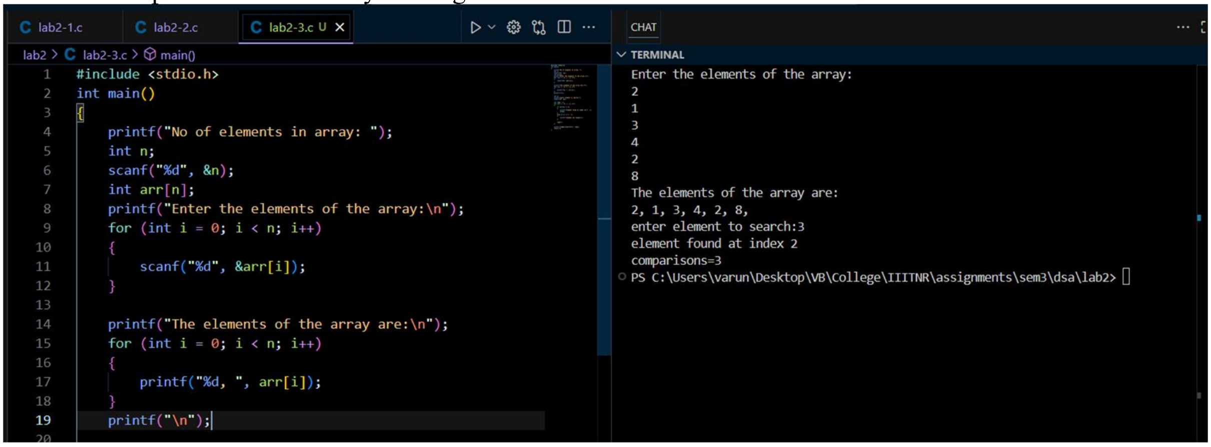
```c
int main()
    printf("selection sort:\n");

    for (int i = 0; i < n; i++)
    {
        int max_index = i;
        for (int j = i + 1; j < n; j++)
        {
            if (arr[j] > arr[max_index])
            {
                max_index = j;
            }
        }
        int t = arr[i];
        arr[i] = arr[max_index];
        arr[max_index] = t;
        for (int i = 0; i < n; i++)
        {
            printf("%d, ", arr[i]);
        }
        printf("\n");
    }

    printf("The sorted array is:\n");
    for (int i = 0; i < n; i++)
    {
        printf("%d, ", arr[i]);
    }

    return 0;
}
```

```
PS C:\Users\varun\Desktop\VB\College\IIITNR\assignments\sem3\dsa\lab2> cd "c:\Use
rs\varun\Desktop\VB\College\IIITNR\assignments\sem3\dsa\lab2\" ; if ($?) { gcc la
b2-2.c -o lab2-2 } ; if ($?) { .\lab2-2 }
No of elements in array: 6
Enter the elements of the array:
2
4
1
3
1
8
The elements of the array are:
2, 4, 1, 3, 1, 8,
selection sort:
8, 4, 1, 3, 1, 2,
8, 4, 1, 3, 1, 2,
8, 4, 3, 1, 1, 2,
8, 4, 3, 2, 1, 1,
8, 4, 3, 2, 1, 1,
8, 4, 3, 2, 1, 1,
The sorted array is:
8, 4, 3, 2, 1, 1,
PS C:\Users\varun\Desktop\VB\College\IIITNR\assignments\sem3\dsa\lab2>
```

Task 3: Linear Search:

(https://github.com/varunnnb/dsa-sem3-iiitnr/blob/main/lab2/lab2-3.c)
Write a program to perform the following operations using Linear Search:

- Take user input to create an array of integers.

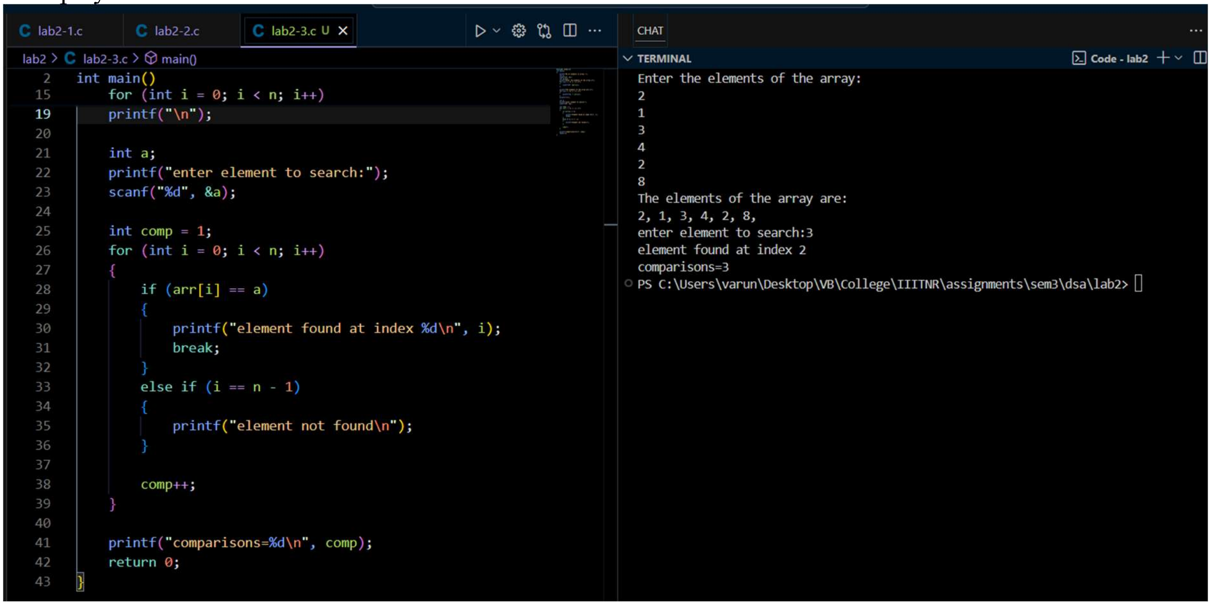


```c
#include <stdio.h>
int main()
{
    printf("No of elements in array: ");
    int n;
    scanf("%d", &n);
    int arr[n];
    printf("Enter the elements of the array:\n");
    for (int i = 0; i < n; i++)
    {
        scanf("%d", &arr[i]);
    }

    printf("The elements of the array are:\n");
    for (int i = 0; i < n; i++)
    {
        printf("%d, ", arr[i]);
    }
    printf("\n");
```

```
Enter the elements of the array:
2
1
3
4
2
8
The elements of the array are:
2, 1, 3, 4, 2, 8,
enter element to search:3
element found at index 2
comparisons=3
PS C:\Users\varun\Desktop\VB\College\IIITNR\assignments\sem3\dsa\lab2>
```

- Take input for the element to be searched in the array.
- Perform Linear Search:

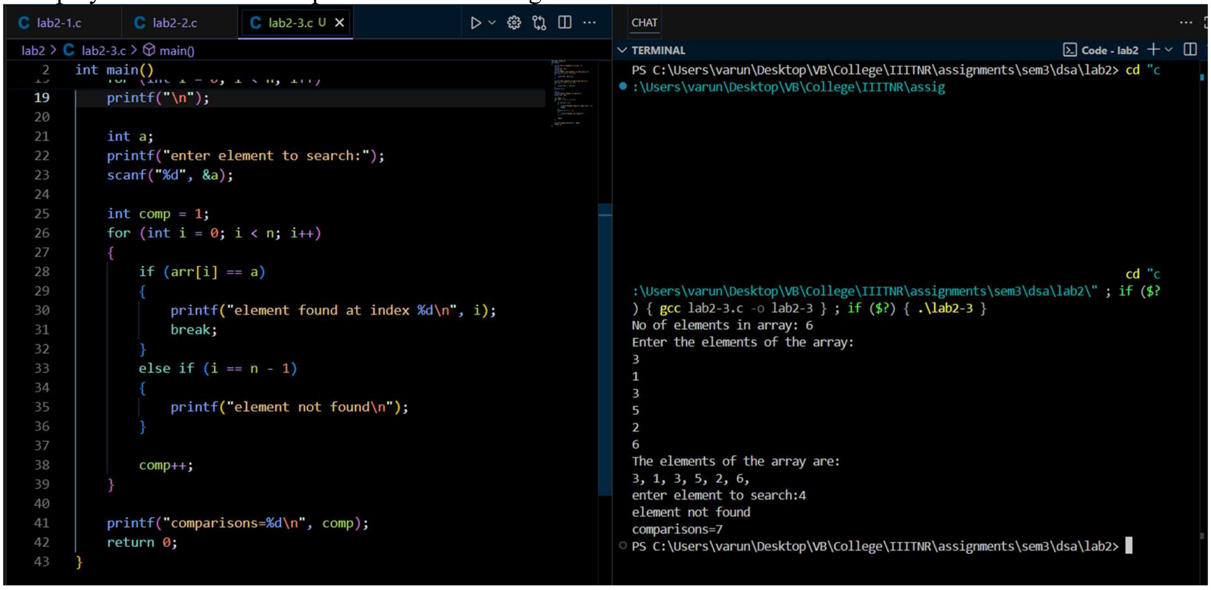
– Display the index of the element if found.



```c
int main()
    for (int i = 0; i < n; i++)
    printf("\n");

    int a;
    printf("enter element to search:");
    scanf("%d", &a);

    int comp = 1;
    for (int i = 0; i < n; i++)
    {
        if (arr[i] == a)
        {
            printf("element found at index %d\n", i);
            break;
        }
        else if (i == n - 1)
        {
            printf("element not found\n");
        }

        comp++;
    }

    printf("comparisons=%d\n", comp);
    return 0;
}
```

```
Enter the elements of the array:
2
1
3
4
2
8
The elements of the array are:
2, 1, 3, 4, 2, 8,
enter element to search:3
element found at index 2
comparisons=3
PS C:\Users\varun\Desktop\VB\College\IIITNR\assignments\sem3\dsa\lab2>
```

– If not found, display   Element not found in the array.
• Display the number of comparisons made during the search.



```c
int main()
    printf("\n");

    int a;
    printf("enter element to search:");
    scanf("%d", &a);

    int comp = 1;
    for (int i = 0; i < n; i++)
    {
        if (arr[i] == a)
        {
            printf("element found at index %d\n", i);
            break;
        }
        else if (i == n - 1)
        {
            printf("element not found\n");
        }

        comp++;
    }

    printf("comparisons=%d\n", comp);
    return 0;
}
```

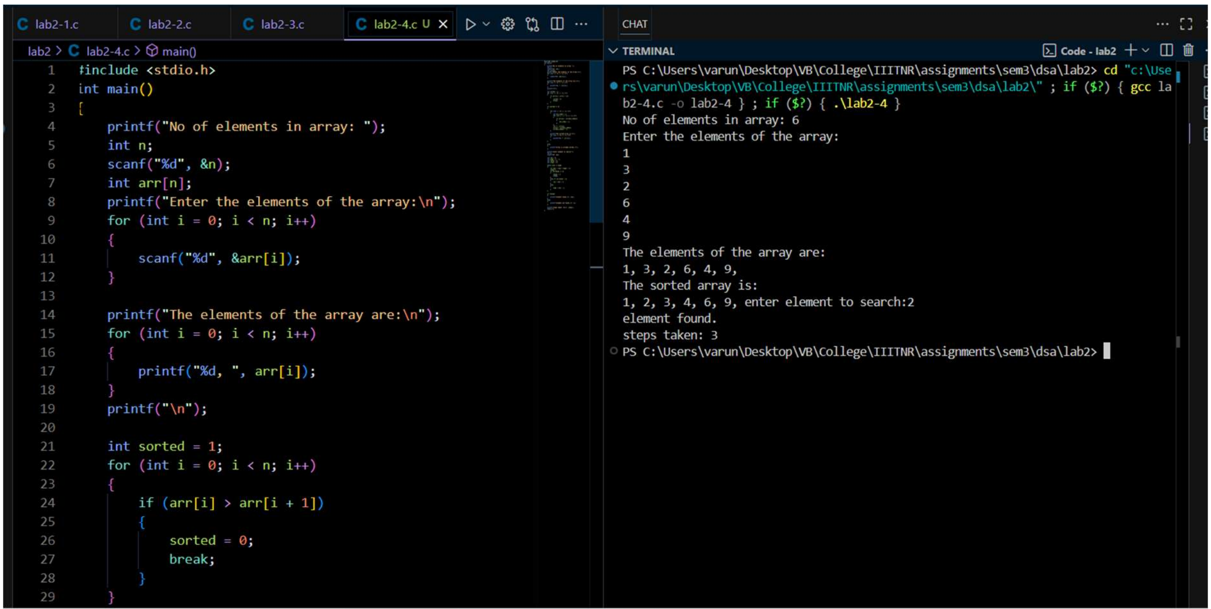
```
PS C:\Users\varun\Desktop\VB\College\IIITNR\assignments\sem3\dsa\lab2> cd "c
:\Users\varun\Desktop\VB\College\IIITNR\assig



                                                                        cd "c
:\Users\varun\Desktop\VB\College\IIITNR\assignments\sem3\dsa\lab2\" ; if ($?
) { gcc lab2-3.c -o lab2-3 } ; if ($?) { .\lab2-3 }
No of elements in array: 6
Enter the elements of the array:
3
1
3
5
2
6
The elements of the array are:
3, 1, 3, 5, 2, 6,
enter element to search:4
element not found
comparisons=7
PS C:\Users\varun\Desktop\VB\College\IIITNR\assignments\sem3\dsa\lab2>
```

Task 4: Binary Search:

(https://github.com/varunnnb/dsa-sem3-iiitnr/blob/main/lab2/lab2-4.c)

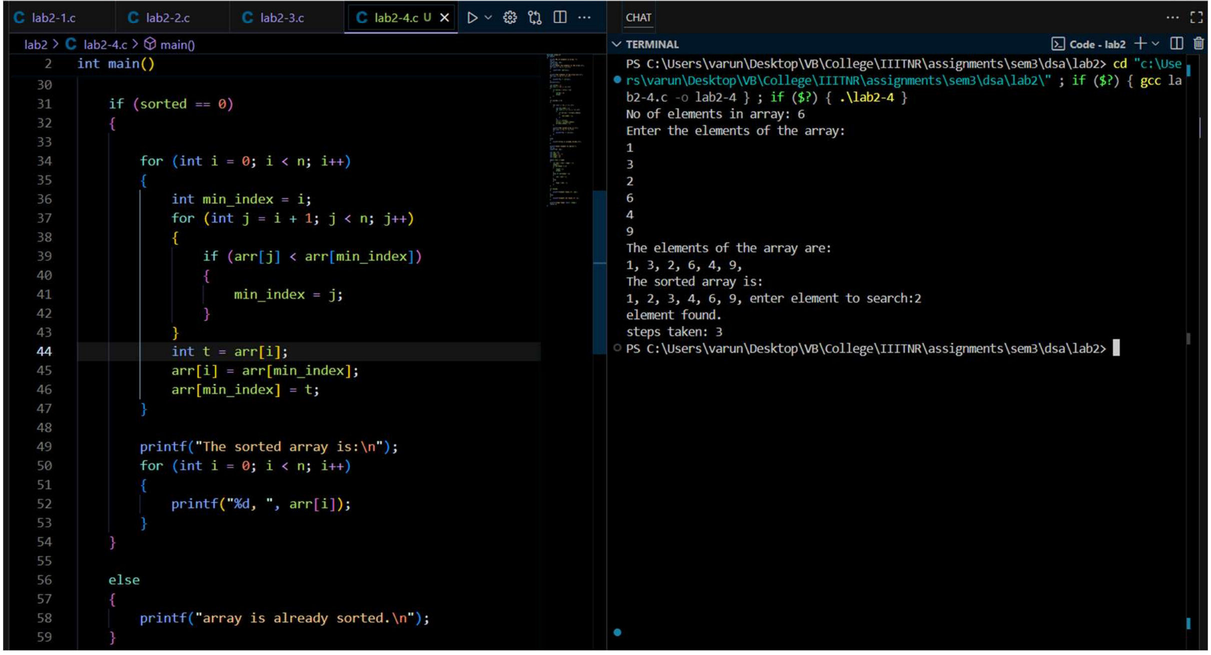
Write a program to perform the following operations using Binary Search:

• Take user input to create a sorted array of integers.



```c
#include <stdio.h>
int main()
{
    printf("No of elements in array: ");
    int n;
    scanf("%d", &n);
    int arr[n];
    printf("Enter the elements of the array:\n");
    for (int i = 0; i < n; i++)
    {
        scanf("%d", &arr[i]);
    }

    printf("The elements of the array are:\n");
    for (int i = 0; i < n; i++)
    {
        printf("%d, ", arr[i]);
    }
    printf("\n");

    int sorted = 1;
    for (int i = 0; i < n; i++)
    {
        if (arr[i] > arr[i + 1])
        {
            sorted = 0;
            break;
        }
    }
```

```
PS C:\Users\varun\Desktop\VB\College\IIITNR\assignments\sem3\dsa\lab2> cd "c:\Use
rs\varun\Desktop\VB\College\IIITNR\assignments\sem3\dsa\lab2\" ; if ($?) { gcc la
b2-4.c -o lab2-4 } ; if ($?) { .\lab2-4 }
No of elements in array: 6
Enter the elements of the array:
1
3
2
6
4
9
The elements of the array are:
1, 3, 2, 6, 4, 9,
The sorted array is:
1, 2, 3, 4, 6, 9, enter element to search:2
element found.
steps taken: 3
PS C:\Users\varun\Desktop\VB\College\IIITNR\assignments\sem3\dsa\lab2>
```

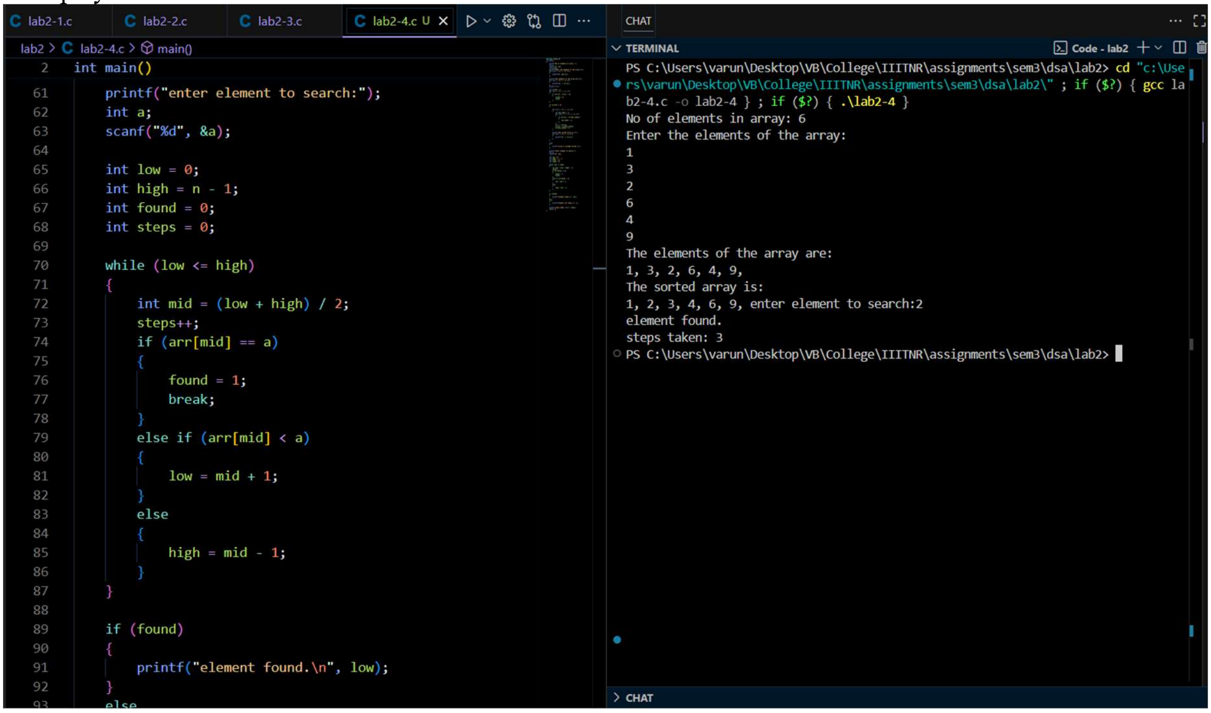


```c
int main()

    if (sorted == 0)
    {

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

        printf("The sorted array is:\n");
        for (int i = 0; i < n; i++)
        {
            printf("%d, ", arr[i]);
        }
    }

    else
    {
        printf("array is already sorted.\n");
    }
```

```
PS C:\Users\varun\Desktop\VB\College\IIITNR\assignments\sem3\dsa\lab2> cd "c:\Use
rs\varun\Desktop\VB\College\IIITNR\assignments\sem3\dsa\lab2\" ; if ($?) { gcc la
b2-4.c -o lab2-4 } ; if ($?) { .\lab2-4 }
No of elements in array: 6
Enter the elements of the array:
1
3
2
6
4
9
The elements of the array are:
1, 3, 2, 6, 4, 9,
The sorted array is:
1, 2, 3, 4, 6, 9, enter element to search:2
element found.
steps taken: 3
PS C:\Users\varun\Desktop\VB\College\IIITNR\assignments\sem3\dsa\lab2>
```

- Take input for the element to be searched in the array.
- Perform Binary Search:

– Display the index of the element if found.

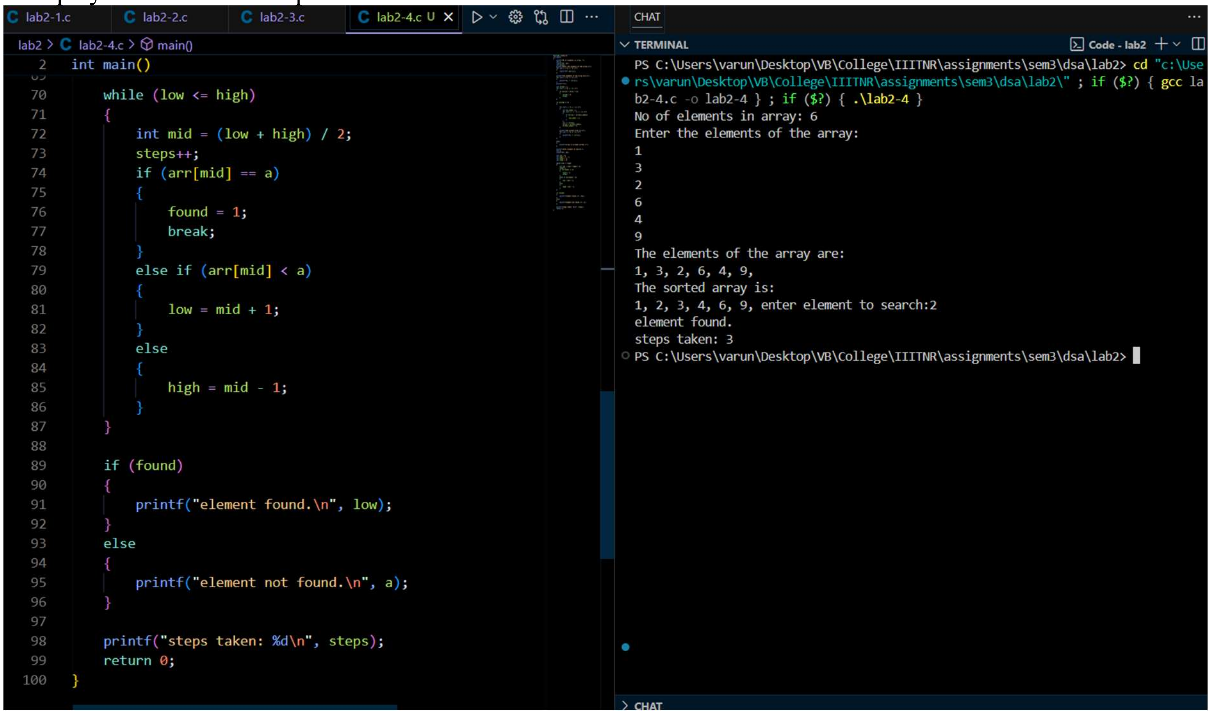


```c
int main()
    printf("enter element to search:");
    int a;
    scanf("%d", &a);

    int low = 0;
    int high = n - 1;
    int found = 0;
    int steps = 0;

    while (low <= high)
    {
        int mid = (low + high) / 2;
        steps++;
        if (arr[mid] == a)
        {
            found = 1;
            break;
        }
        else if (arr[mid] < a)
        {
            low = mid + 1;
        }
        else
        {
            high = mid - 1;
        }
    }

    if (found)
    {
        printf("element found.\n", low);
    }
    else
```

```
PS C:\Users\varun\Desktop\VB\College\IIITNR\assignments\sem3\dsa\lab2> cd "c:\Use
rs\varun\Desktop\VB\College\IIITNR\assignments\sem3\dsa\lab2\" ; if ($?) { gcc la
b2-4.c -o lab2-4 } ; if ($?) { .\lab2-4 }
No of elements in array: 6
Enter the elements of the array:
1
3
2
6
4
9
The elements of the array are:
1, 3, 2, 6, 4, 9,
The sorted array is:
1, 2, 3, 4, 6, 9, enter element to search:2
element found.
steps taken: 3
PS C:\Users\varun\Desktop\VB\College\IIITNR\assignments\sem3\dsa\lab2>
```

– If not found, display an appropriate message.

- Display the number of steps/iterations taken to find the element.



```c
int main()
    while (low <= high)
    {
        int mid = (low + high) / 2;
        steps++;
        if (arr[mid] == a)
        {
            found = 1;
            break;
        }
        else if (arr[mid] < a)
        {
            low = mid + 1;
        }
        else
        {
            high = mid - 1;
        }
    }

    if (found)
    {
        printf("element found.\n", low);
    }
    else
    {
        printf("element not found.\n", a);
    }

    printf("steps taken: %d\n", steps);
    return 0;
}
```

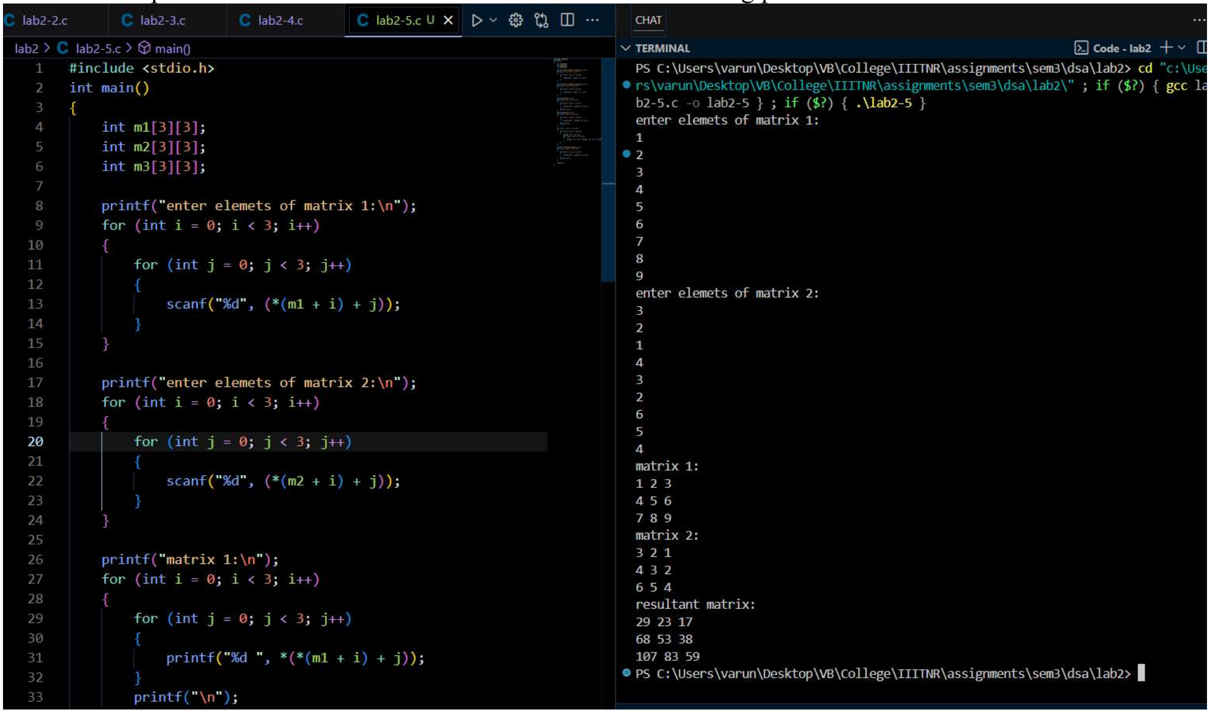
```
PS C:\Users\varun\Desktop\VB\College\IIITNR\assignments\sem3\dsa\lab2> cd "c:\Use
rs\varun\Desktop\VB\College\IIITNR\assignments\sem3\dsa\lab2\" ; if ($?) { gcc la
b2-4.c -o lab2-4 } ; if ($?) { .\lab2-4 }
No of elements in array: 6
Enter the elements of the array:
1
3
2
6
4
9
The elements of the array are:
1, 3, 2, 6, 4, 9,
The sorted array is:
1, 2, 3, 4, 6, 9, enter element to search:2
element found.
steps taken: 3
PS C:\Users\varun\Desktop\VB\College\IIITNR\assignments\sem3\dsa\lab2>
```

Task 5: Matrix Multiplication Using Pointers:

(https://github.com/varunnnb/dsa-sem3-iiitnr/blob/main/lab2/lab2-5.c)

Write a program to perform the following operations related to matrix multiplication:

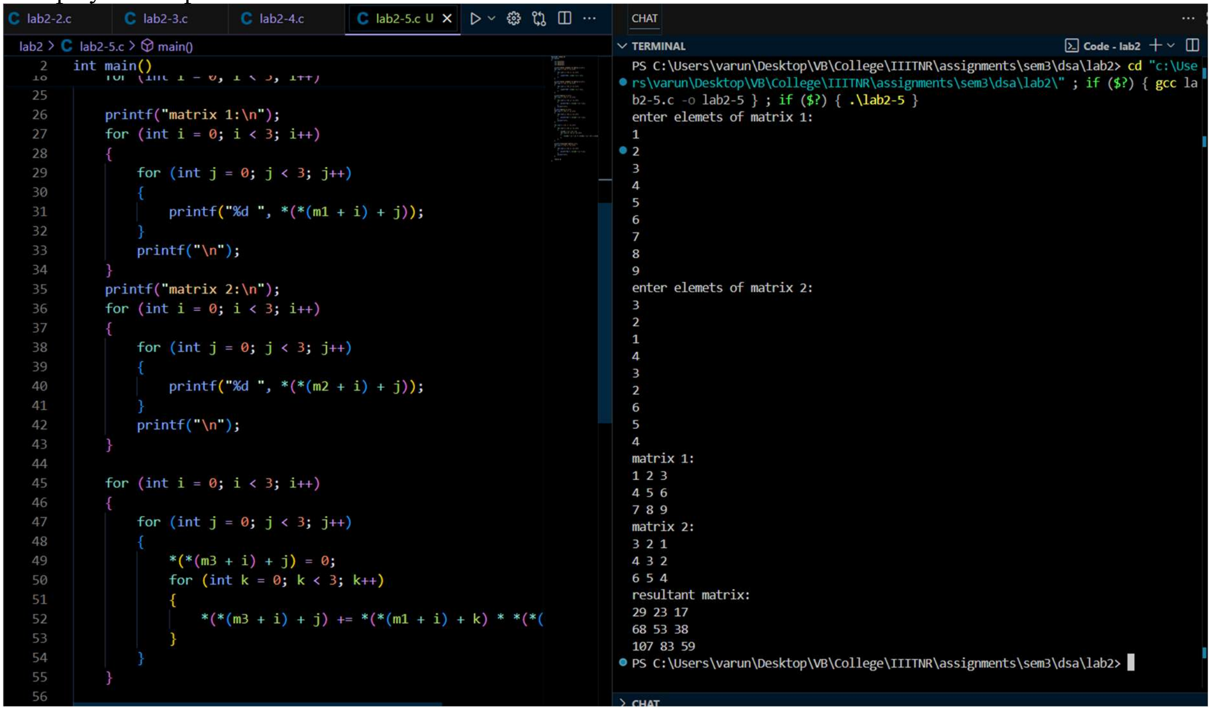
- Declare three 2D matrices A[3][3], B[3][3], and C[3][3].
- Take user input to enter all elements of Matrix A and Matrix B using pointers.



```c
#include <stdio.h>
int main()
{
    int m1[3][3];
    int m2[3][3];
    int m3[3][3];

    printf("enter elemets of matrix 1:\n");
    for (int i = 0; i < 3; i++)
    {
        for (int j = 0; j < 3; j++)
        {
            scanf("%d", (*(m1 + i) + j));
        }
    }

    printf("enter elemets of matrix 2:\n");
    for (int i = 0; i < 3; i++)
    {
        for (int j = 0; j < 3; j++)
        {
            scanf("%d", (*(m2 + i) + j));
        }
    }

    printf("matrix 1:\n");
    for (int i = 0; i < 3; i++)
    {
        for (int j = 0; j < 3; j++)
        {
            printf("%d ", *(*(m1 + i) + j));
        }
        printf("\n");
```

```
PS C:\Users\varun\Desktop\VB\College\IIITNR\assignments\sem3\dsa\lab2> cd "c:\Use
rs\varun\Desktop\VB\College\IIITNR\assignments\sem3\dsa\lab2\" ; if ($?) { gcc la
b2-5.c -o lab2-5 } ; if ($?) { .\lab2-5 }
enter elemets of matrix 1:
1
2
3
4
5
6
7
8
9
enter elemets of matrix 2:
3
2
1
4
3
2
6
5
4
matrix 1:
1 2 3
4 5 6
7 8 9
matrix 2:
3 2 1
4 3 2
6 5 4
resultant matrix:
29 23 17
68 53 38
107 83 59
PS C:\Users\varun\Desktop\VB\College\IIITNR\assignments\sem3\dsa\lab2>
```

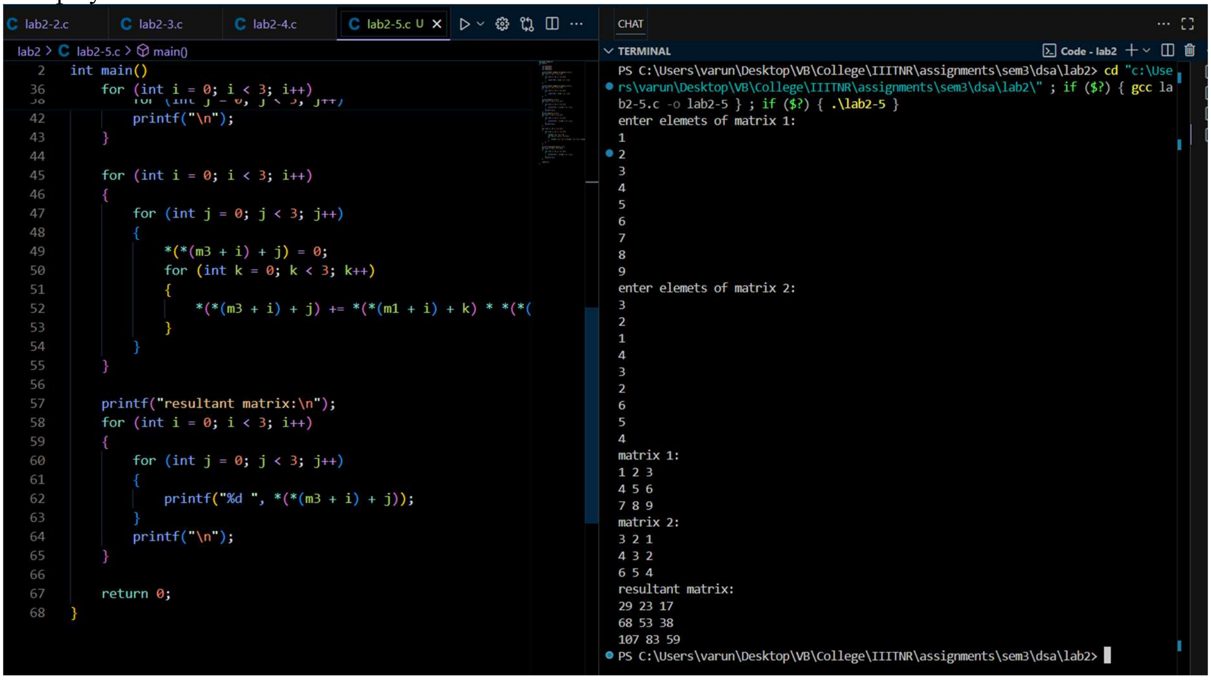
- Display both input matrices in matrix format.



```c
int main()

    printf("matrix 1:\n");
    for (int i = 0; i < 3; i++)
    {
        for (int j = 0; j < 3; j++)
        {
            printf("%d ", *(*(m1 + i) + j));
        }
        printf("\n");
    }
    printf("matrix 2:\n");
    for (int i = 0; i < 3; i++)
    {
        for (int j = 0; j < 3; j++)
        {
            printf("%d ", *(*(m2 + i) + j));
        }
        printf("\n");
    }

    for (int i = 0; i < 3; i++)
    {
        for (int j = 0; j < 3; j++)
        {
            *(*(m3 + i) + j) = 0;
            for (int k = 0; k < 3; k++)
            {
                *(*(m3 + i) + j) += *(*(m1 + i) + k) * *(*(
            }
        }
    }

```

```
PS C:\Users\varun\Desktop\VB\College\IIITNR\assignments\sem3\dsa\lab2> cd "c:\Use
rs\varun\Desktop\VB\College\IIITNR\assignments\sem3\dsa\lab2\" ; if ($?) { gcc la
b2-5.c -o lab2-5 } ; if ($?) { .\lab2-5 }
enter elemets of matrix 1:
1
2
3
4
5
6
7
8
9
enter elemets of matrix 2:
3
2
1
4
3
2
6
5
4
matrix 1:
1 2 3
4 5 6
7 8 9
matrix 2:
3 2 1
4 3 2
6 5 4
resultant matrix:
29 23 17
68 53 38
107 83 59
PS C:\Users\varun\Desktop\VB\College\IIITNR\assignments\sem3\dsa\lab2>
```

- Perform matrix multiplication using pointer arithmetic and store the result in Matrix C.
- Display the resultant matrix C in matrix format.



```c
int main()
    for (int i = 0; i < 3; i++)
        for (int j = 0; j < 3; j++)
        printf("\n");
    }

    for (int i = 0; i < 3; i++)
    {
        for (int j = 0; j < 3; j++)
        {
            *(*(m3 + i) + j) = 0;
            for (int k = 0; k < 3; k++)
            {
                *(*(m3 + i) + j) += *(*(m1 + i) + k) * *(*(
            }
        }
    }

    printf("resultant matrix:\n");
    for (int i = 0; i < 3; i++)
    {
        for (int j = 0; j < 3; j++)
        {
            printf("%d ", *(*(m3 + i) + j));
        }
        printf("\n");
    }

    return 0;
}
```

```
PS C:\Users\varun\Desktop\VB\College\IIITNR\assignments\sem3\dsa\lab2> cd "c:\Use
rs\varun\Desktop\VB\College\IIITNR\assignments\sem3\dsa\lab2\" ; if ($?) { gcc la
b2-5.c -o lab2-5 } ; if ($?) { .\lab2-5 }
enter elemets of matrix 1:
1
2
3
4
5
6
7
8
9
enter elemets of matrix 2:
3
2
1
4
3
2
6
5
4
matrix 1:
1 2 3
4 5 6
7 8 9
matrix 2:
3 2 1
4 3 2
6 5 4
resultant matrix:
29 23 17
68 53 38
107 83 59
PS C:\Users\varun\Desktop\VB\College\IIITNR\assignments\sem3\dsa\lab2>
```