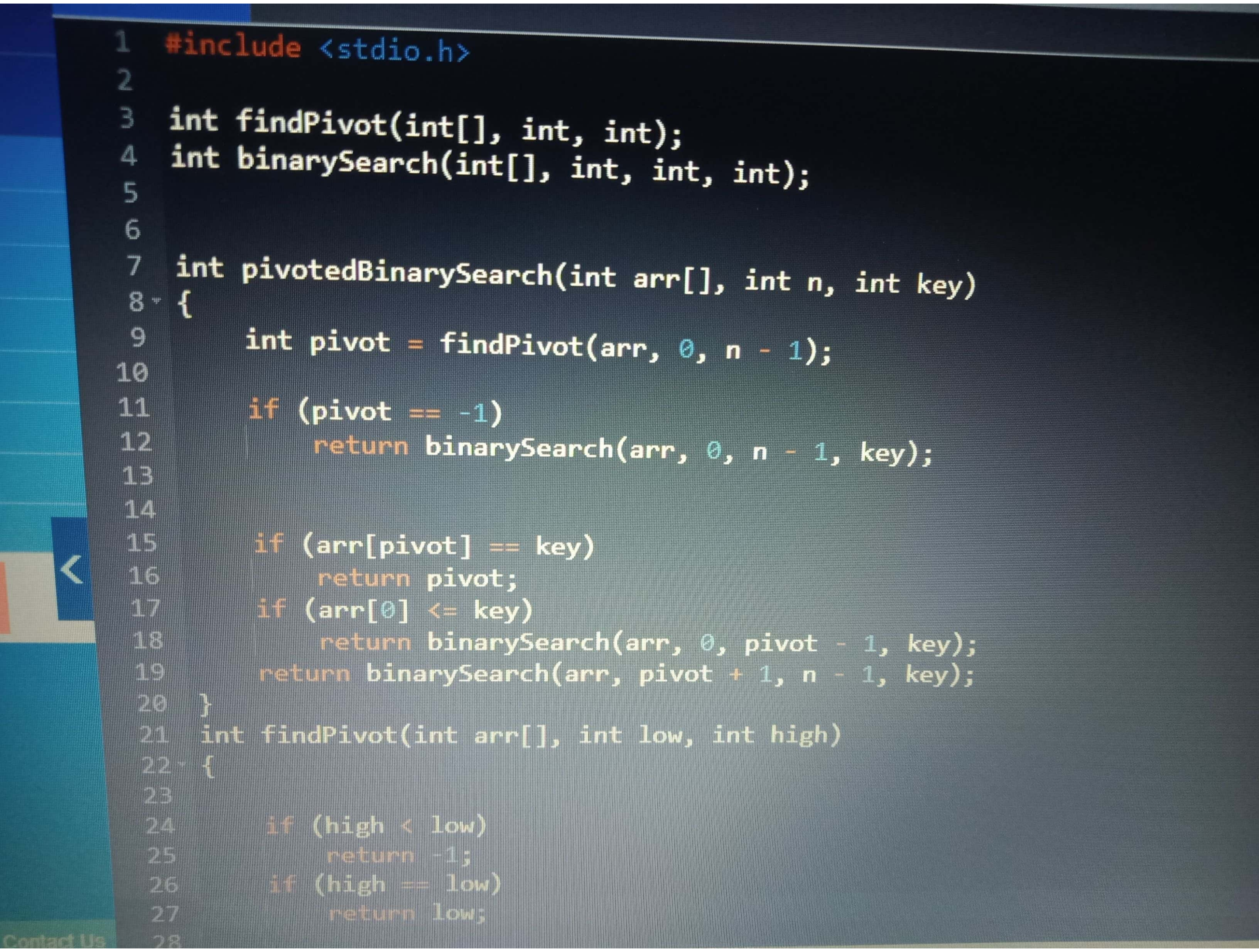
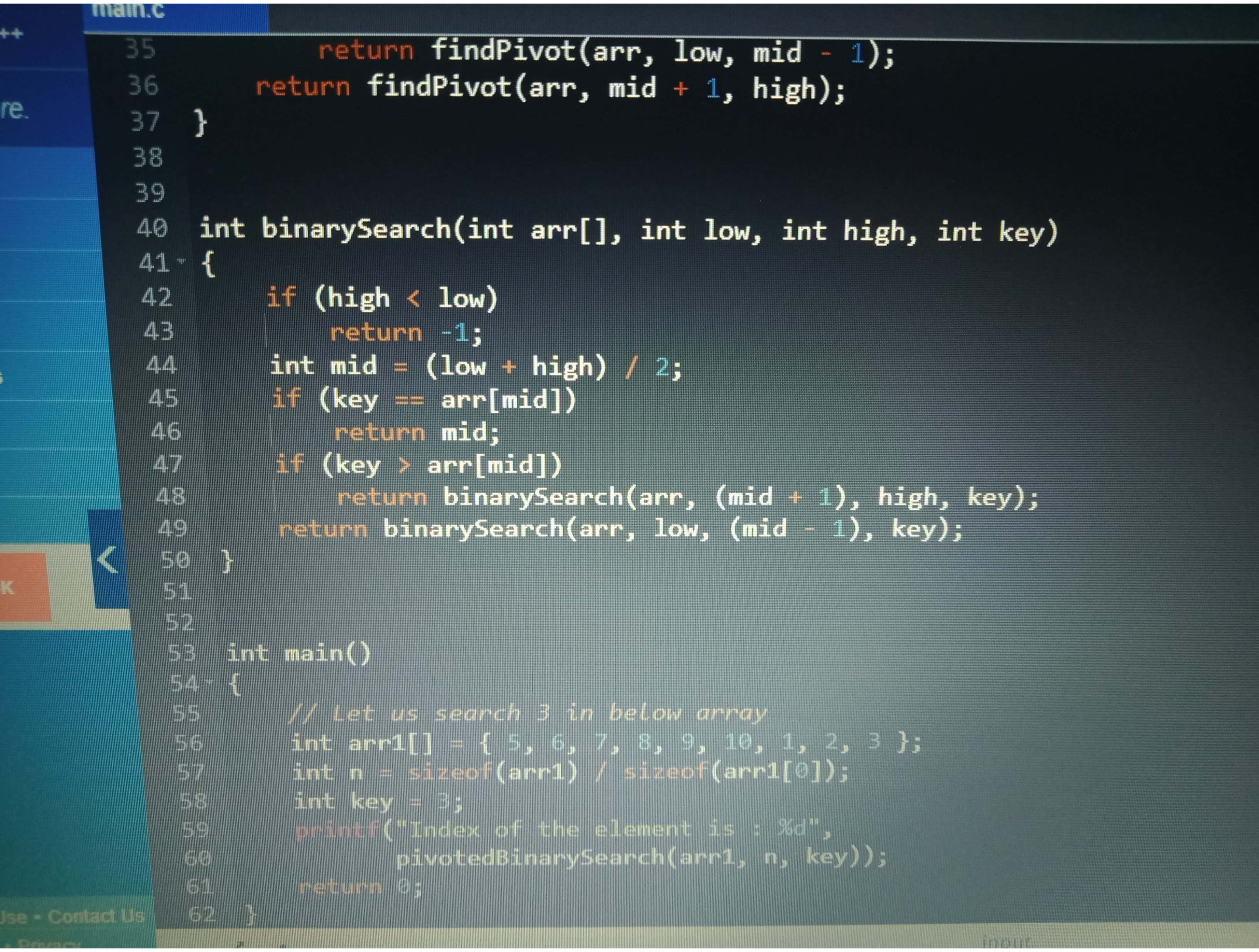
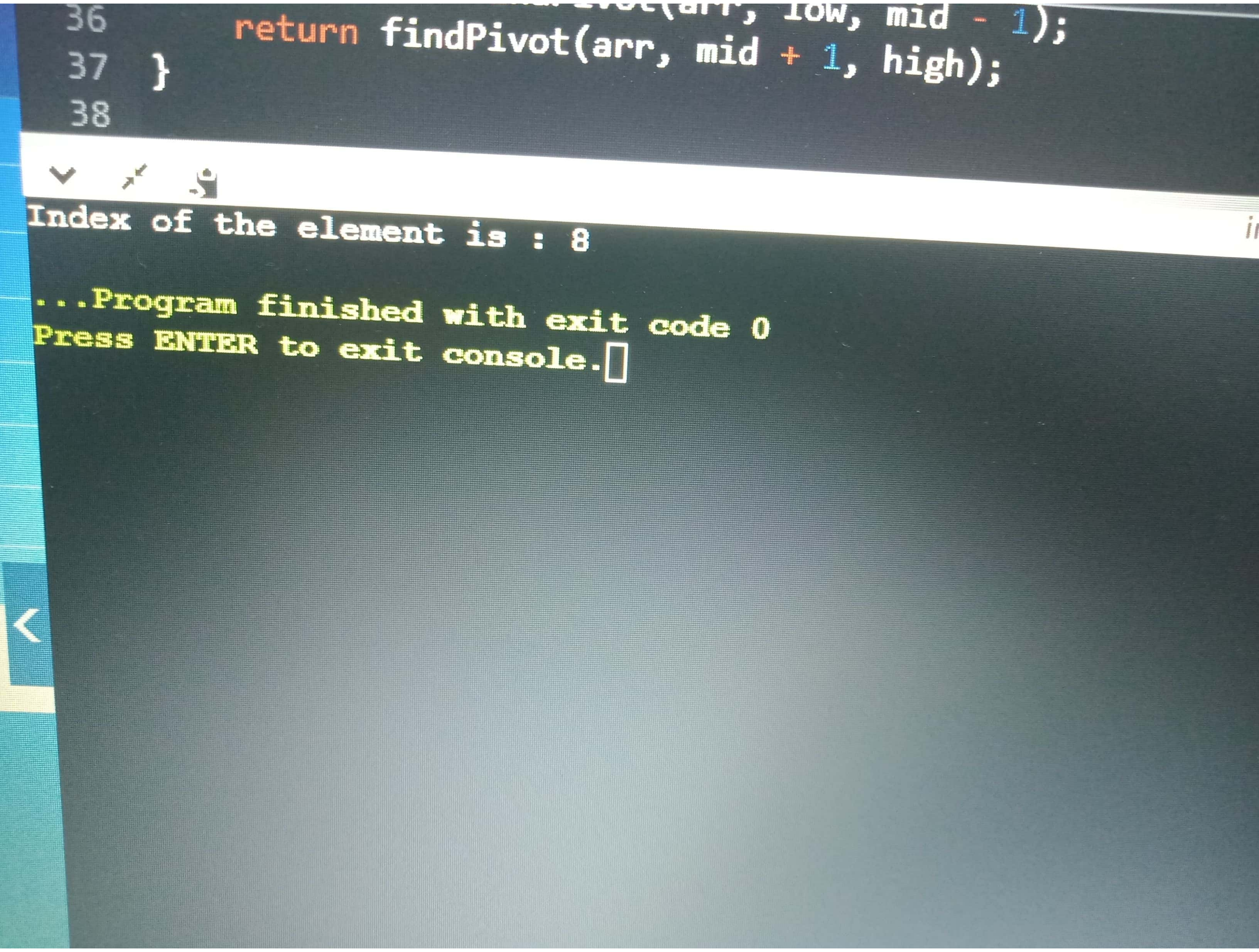


```c
#include <stdio.h>

int findPivot(int[], int, int);
int binarySearch(int[], int, int, int);


int pivotedBinarySearch(int arr[], int n, int key)
{
    int pivot = findPivot(arr, 0, n - 1);

    if (pivot == -1)
        return binarySearch(arr, 0, n - 1, key);


    if (arr[pivot] == key)
        return pivot;
    if (arr[0] <= key)
        return binarySearch(arr, 0, pivot - 1, key);
    return binarySearch(arr, pivot + 1, n - 1, key);
}
int findPivot(int arr[], int low, int high)
{

    if (high < low)
        return -1;
    if (high == low)
        return low;
```



```
        return findPivot(arr, low, mid - 1);
    return findPivot(arr, mid + 1, high);
}


int binarySearch(int arr[], int low, int high, int key)
{
    if (high < low)
        return -1;
    int mid = (low + high) / 2;
    if (key == arr[mid])
        return mid;
    if (key > arr[mid])
        return binarySearch(arr, (mid + 1), high, key);
    return binarySearch(arr, low, (mid - 1), key);
}


int main()
{
    // Let us search 3 in below array
    int arr1[] = { 5, 6, 7, 8, 9, 10, 1, 2, 3 };
    int n = sizeof(arr1) / sizeof(arr1[0]);
    int key = 3;
    printf("Index of the element is : %d",
           pivotedBinarySearch(arr1, n, key));
    return 0;
}
```

Scanned with CamScanner



```
              ivot(arr, low, mid - 1);
        return findPivot(arr, mid + 1, high);
}
```

```
Index of the element is : 8

...Program finished with exit code 0
Press ENTER to exit console.
```