14. Write C programs that demonstrate the mathematical analysis of non-recursive and recursive algorithms.

Code:

Non Recursive algorithm:

```c
#include <stdio.h>
#include <stdbool.h>

bool linearSearch(int arr[], int size, int target) {
    for (int i = 0; i < size; i++) {
        if (arr[i] == target) {
            return true;
        }
    }
    return false;
}

int main() {
    int arr[] = {1, 3, 5, 7, 9, 11, 13, 15, 17, 19};
    int size = sizeof(arr) / sizeof(arr[0]);
    int target = 7;

    bool found = linearSearch(arr, size, target);

    if (found) {
        printf("Element %d found in the array.\n", target);
    } else {
        printf("Element %d not found in the array.\n", target);
    }

    return 0;
}
```

Recursive algorithm:

```c
#include <stdio.h>

int factorial(int n) {
    if (n == 0) {
        return 1;
    } else {
        return n * factorial(n - 1);
    }
}

int main() {
    int number = 5;
    int result = factorial(number);

    printf("Factorial of %d is %d\n", number, result);

    return 0;
}
```

Output:

Non Recursive algorithm:

```
Element 7 found in the array.
```

Recursive algorithm:

```
Factorial of 5 is 120
```

Time Complexity:

- T(n)= O(n)