Practical-2

Bubble Sort

Iterative Approach:

main.cpp Share Run

```cpp
#include <iostream>
#include <vector>
#include <algorithm>
using namespace std;

int main() {
    int n;
    cout << "Enter the size of array: ";
    cin >> n;

    vector<int> in(n);
    cout << "Enter " << n << " elements:\n";
    for(int i = 0; i < n; i++) {
        cin >> in[i];
    }

    int counter = 0;

    counter += 2;
    counter++;

    for (int i = 0; i < n - 1; i++) {
        counter++;
        counter++;

        for (int j = 0; j < n - 1 - i; j++) {
            counter++;
            counter++;
            counter++;

            if (in[j] > in[j + 1]) {
                counter++;
                swap(in[j], in[j + 1]);
            }

            counter++;
        }

        counter++;
        counter++;
    }
```

```
41      }
42
43      counter++;
44      counter++;
45
46      cout << "Counter using Iterative Approach (Worst Case): " <<
            counter;
47      return 0;
48  }
```

INPUT(3)

```
Enter the size of array: 3
Enter 3 elements:
1 6 8
Counter using Iterative Approach (Worst Case): 25

=== Code Execution Successful ===
```

INPUT(5)

```
Enter the size of array: 5
Enter 5 elements:
45 55 15 12 4
Counter using Iterative Approach (Worst Case): 70
```

INPUT(8)

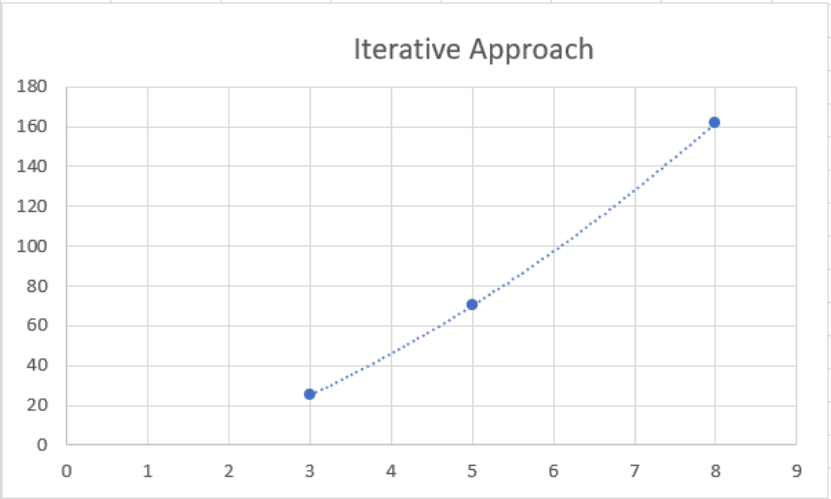
```
Enter the size of array: 8
Enter 8 elements:
14 25 3 1 7 9 8 5
Counter using Iterative Approach (Worst Case): 162

=== Code Execution Successful ===
```

Graph:

Practical-2

3	25
5	70
8	162



Recursive Approach:

main.cpp Share Run

```cpp
#include <iostream>
#include <vector>
#include <algorithm>
using namespace std;

int counter = 0;

void bubble(vector<int> &in, int n) {
    counter++;
    counter++;
    if (n == 1) {
        counter++;
        return;
    }

    counter++; // i = 0
    for (int i = 0; i < n - 1; i++) {
        counter++;
        counter++;
        counter++;
```

Practical-2

main.cpp Share Run

```
        if (in[i] > in[i + 1]) {
            counter++;
            swap(in[i], in[i + 1]);
        }
        counter++;
    }

    counter++;
    counter++;
    counter++;
    bubble(in, n - 1);
}

int main() {
    int n;
    cout << "Enter the size of array: ";
    cin >> n;
```

```
    vector<int> in(n);
    cout << "Enter " << n << " elements:\n";
    for(int i = 0; i < n; i++) {
        cin >> in[i];
    }

    counter += 2;
    counter++;
    bubble(in, n);

    counter++;

    cout << "Input Size: " << n << endl;
    cout << "Counter using Recursive Approach (Worst Case): " <<
        counter << endl;

    return 0;
}
```

INPUT(3)

Practical-2

```
Enter the size of array: 3
Enter 3 elements:
5 7 3
Input Size: 3
Counter using Recursive Approach (Worst Case): 33


=== Code Execution Successful ===
```

INPUT(5)

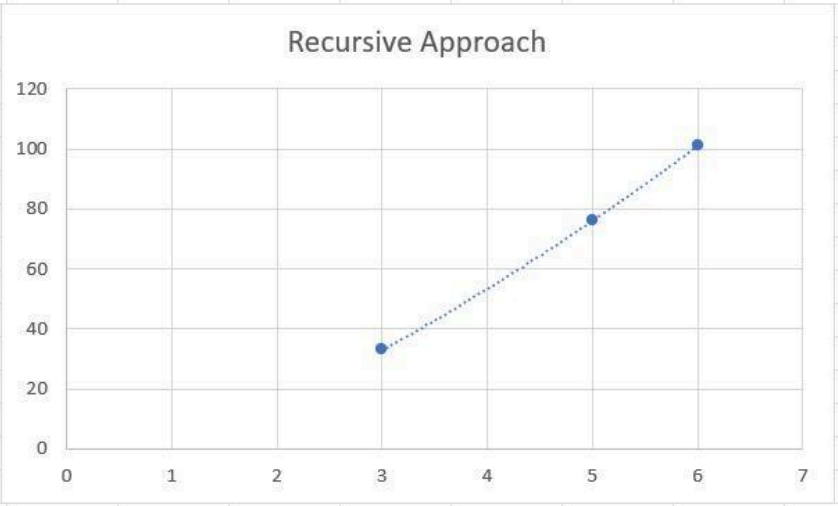
```
Enter the size of array: 5
Enter 5 elements:
4 9 7 5 6
Input Size: 5
Counter using Recursive Approach (Worst Case): 76


=== Code Execution Successful ===
```

INPUT(6)

```
Enter the size of array: 6
Enter 6 elements:
2 3 45 12 32 22
Input Size: 6
Counter using Recursive Approach (Worst Case): 101
```

Graph:

Practical-2

3	33
5	76
6	101