UNIVERSITI TEKNOLOGI PETRONAS

TEB1113
ALGORITHM AND DATA STRUCTURE

No.	Name	ID No	Programme
1	Amir Zikri bin Ahmad Badruddin	22002705	Bachelor of Computer Science
2	Faris Amsyar bin Mat Rodhi	22006426	Bachelor of Computer Science
3	Muhammad Fakruzzaharfan bin Mohd Farydz	22007161	Bachelor of Computer Science
4	Nurin Akma Alya Binti Muhammad Akmal	24006493	Bachelor of Computer Science

Functions

average()

```
1 reference
public float average()
{
    if (num == 0) return 0;

    float sum = 0;
    for (int i = 0; i < num; i++)
    {
        sum += agents[i].ID;
    }
    return sum / num;
}
```

max()

```
1 reference
public int max()
{
    if (num == 0) return 0;

    int maxValue = agents[0].ID;
    for (int i = 1; i < num; i++)
    {
        if (agents[i].ID > maxValue)
        {
            maxValue = agents[i].ID;
        }
    }
    return maxValue;
}
```

min()

```
1 reference
public int min()
{
    if (num == 0) return 0;

    int minValue = agents[0].ID;
    for (int i = 1; i < num; i++)
    {
        if (agents[i].ID < minValue)
        {
            minValue = agents[i].ID;
        }
    }
    return minValue;
}
```

bubblesort()

```
1 reference
public void bubblesort()
{
    for (int i = 0; i < num - 1; i++)
    {
        for (int j = 0; j < num - i - 1; j++)
        {
            if (agents[j].ID > agents[j + 1].ID)
            {
                Drone temp = agents[j];
                agents[j] = agents[j + 1];
                agents[j + 1] = temp;
            }
        }
    }
}
```

Timing for each function

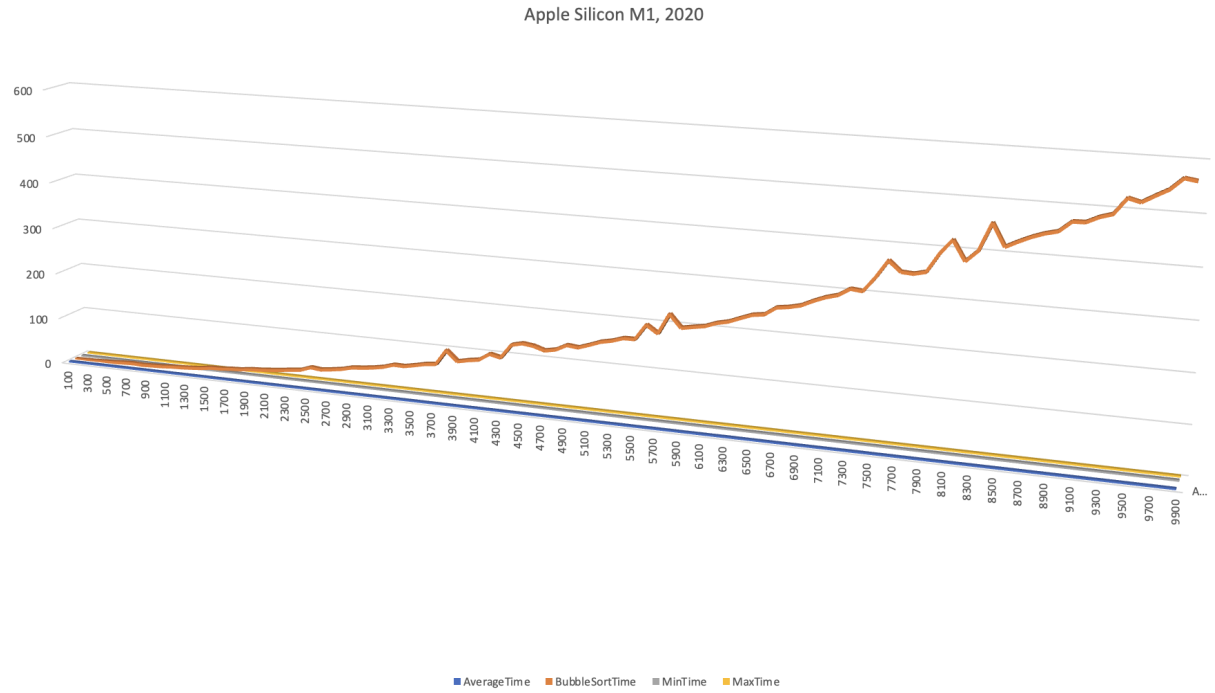
```
for (int i = 0; i < numsteps; i++)
{
    int numdrones = i * stepsize + min;
    Console.WriteLine("Current num drones = " + numdrones);

    Flock flock = new Flock(numdrones);
    flock.Init(numdrones); // Initialize flock with numdrones

    var watch = new System.Diagnostics.Stopwatch();

    // Timing for average()
    watch.Start();
    for (int rep = 0; rep < numRepeat; rep++)
    {
        flock.average();
    }
    watch.Stop();
    timeAverage[i] = watch.ElapsedMilliseconds / (float)numRepeat;

    // Timing for bubble sort
    watch.Restart();
    for (int rep = 0; rep < numRepeat; rep++)
    {
        flock.bubblesort();
    }
    watch.Stop();
    timeBubbleSort[i] = watch.ElapsedMilliseconds / (float)numRepeat;

    // Timing for min()
    watch.Restart();
    for (int rep = 0; rep < numRepeat; rep++)
    {
        flock.min();
    }
    watch.Stop();
    timeMin[i] = watch.ElapsedMilliseconds / (float)numRepeat;

    // Timing for max()
    watch.Restart();
    for (int rep = 0; rep < numRepeat; rep++)
    {
        flock.max();
    }
    watch.Stop();
    timeMax[i] = watch.ElapsedMilliseconds / (float)numRepeat;
}
```

Print Results into CSV

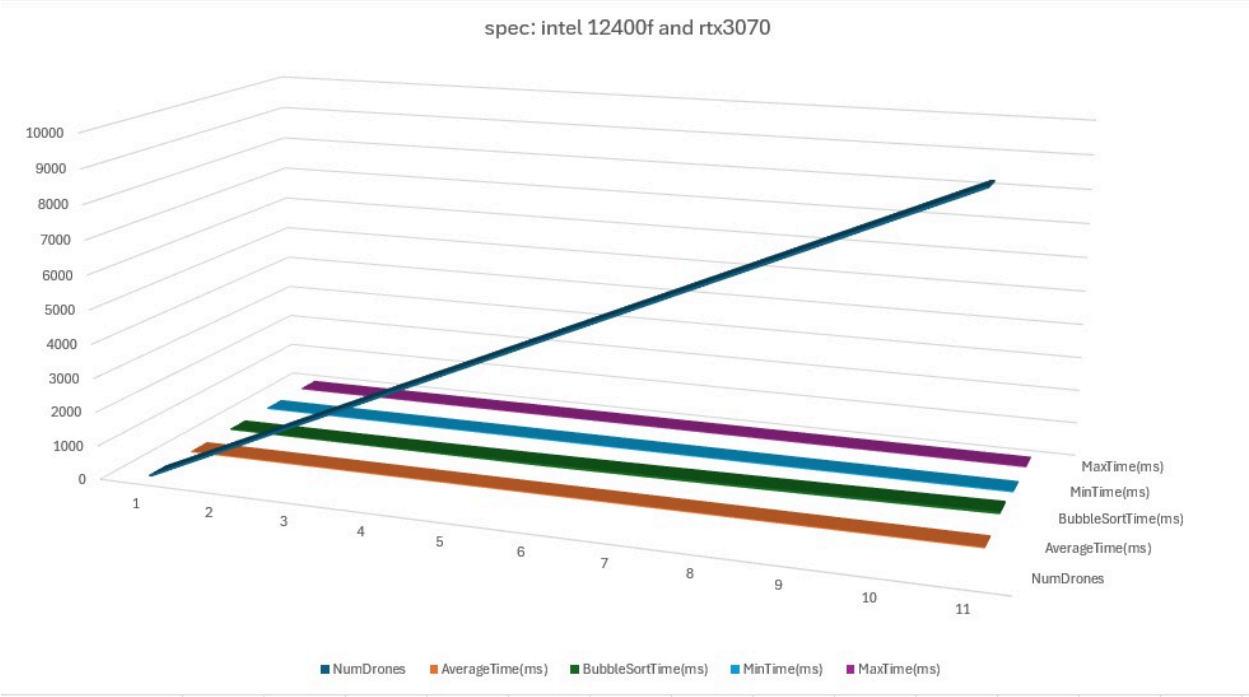
```
// Write results to CSV
using (StreamWriter file = new StreamWriter("results.csv"))
{
    file.WriteLine("NumDrones,AverageTime,BubbleSortTime,MinTime,MaxTime");
    for (int i = 0; i < numsteps; i++)
    {
        int numdrones = i * stepsize + min;
        file.WriteLine($"{numdrones},{timeAverage[i]},{timeBubbleSort[i]},{timeMin[i]},{timeMax[i]}");
    }
}

Console.WriteLine("Results written to results.csv");
```

Graph Results

1. Apple Silicon M1 , 2020



2. Intel 12400f and RTX3070



3. Intel(R) Core(™) i5-7200U