Operating Systems

Assignment 3

Roll no: 200901018

Name: Muhammad Tayyab

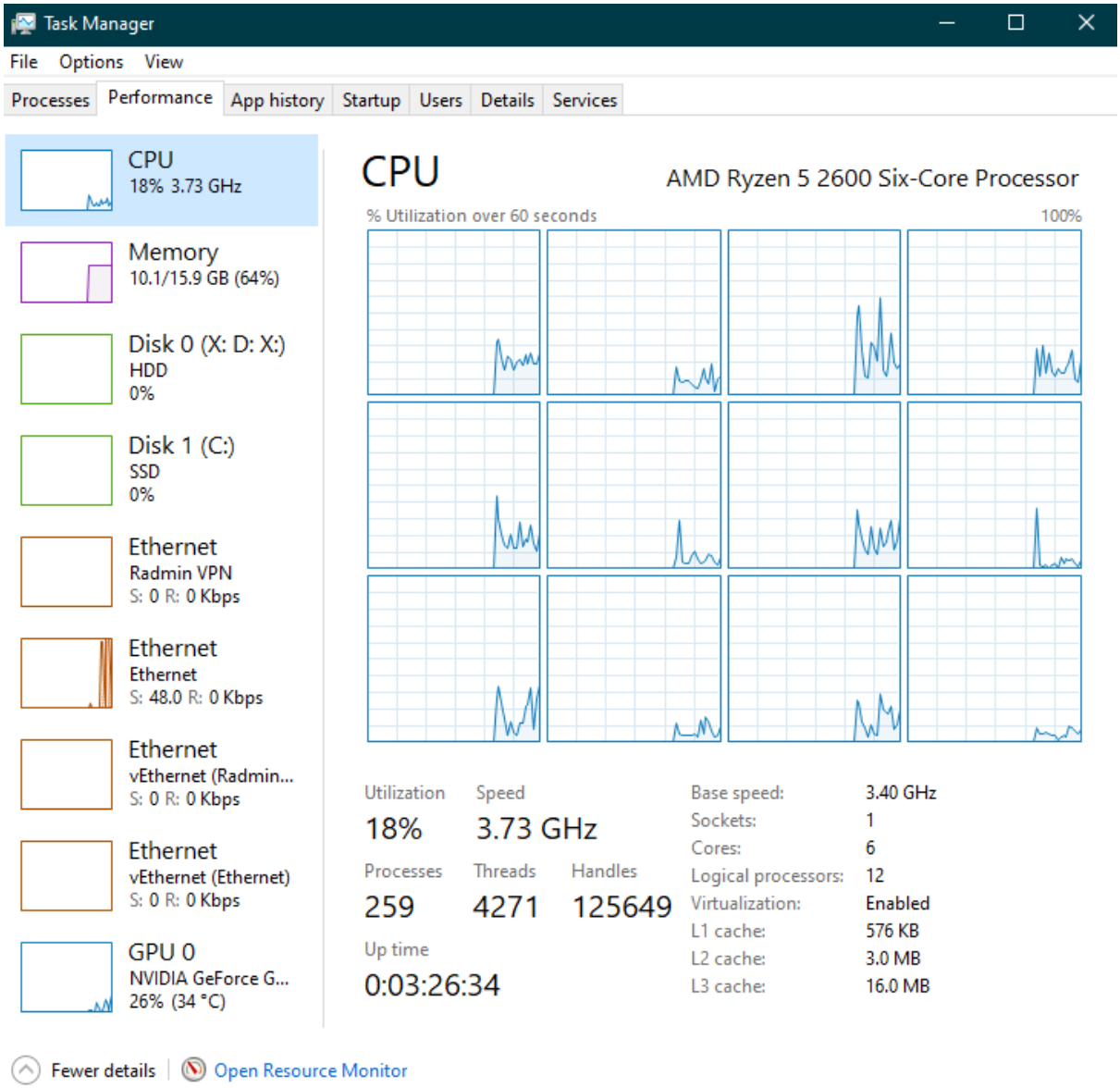
Batch & Section: BS-CS-01 (B)

Date of Submission: 28/12/2022

Merge Sort using Threads

CPU Cores and Threads:

Our processor has 6 cores, and each core is capable of running 2 threads at one time.



Total Threads are 12, we will be utilizing them for our merge sort algorithm.

Merge Sort Code:

```
#include <iostream>
#include <thread>
#include <cstdlib>
#include <conio.h>
```

```
using namespace std;


// merge sort function taken from programmiz
https://www.programiz.com/dsa/merge-sort
void merge(int arr[], int left, int middle, int right)
{
    int l, m, r;    //indexes that we use to denote left right and center
    int n1 = middle - left + 1; //name of sub arrays we use for combining
    int n2 = right - middle;

    int* L = new int[n1];
    int* R = new int[n2];   //declaring the sub arrays

    for (l = 0; l < n1; l++)    //copying values of the array from sub
array to the combined array
        L[l] = arr[left + l];
    for (m = 0; m < n2; m++)
        R[m] = arr[middle + 1 + m];

    l = 0;
    m = 0;
    r = left;
    while (l < n1 && m < n2){        //logic to decide whether to combine
the index with left array or right array
        if (L[l] <= R[m]){
            arr[r] = L[l];
            l++;
        }
        else{
            arr[r] = R[m];
            m++;
        }
        r++;
    }

    while (l < n1){
        arr[r] = L[l];
        l++;
        r++;
    }

    while (m < n2){
        arr[r] = R[m];
        m++;
        r++;
    }
}

void mergeSort(int arr[], int left, int right){
    if (left < right){
        int middle = left + (right - left) / 2;

        //assign arrays to separate threads
```

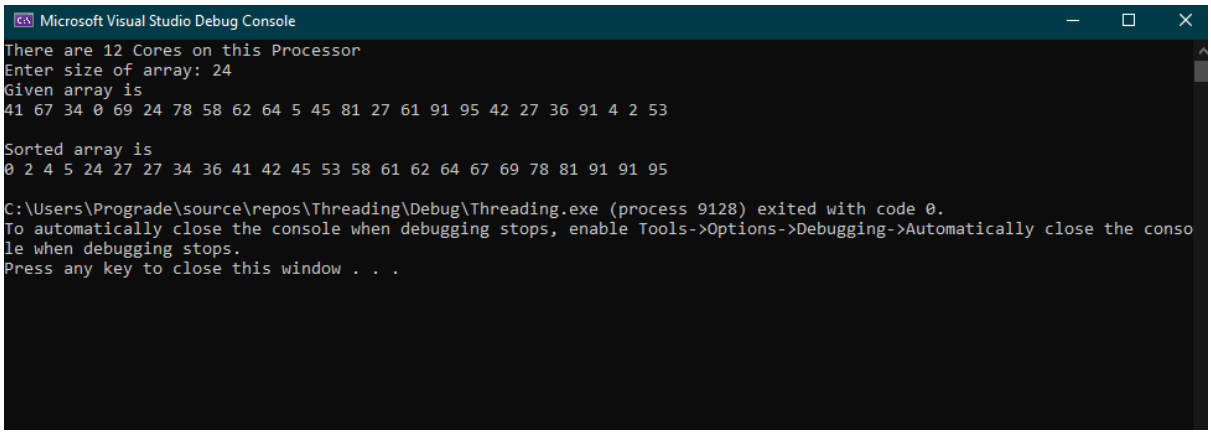
```
        thread leftSorter(mergeSort, arr, left, middle);
        thread rightSorter(mergeSort, arr, middle + 1, right);

        //merge outputs after join
        leftSorter.join();
        rightSorter.join();
        merge(arr, left, middle, right);
    }
}

int main(){
    int* array1;
    int arr_size;
    cout << "There are 12 Cores on this Processor\n";
    cout << "Enter size of array: ";
    cin >> arr_size;
    array1 = new int[arr_size];
    for (int i = 0; i < arr_size; i++){
        array1[i] = (rand() % 100);
    }
    cout << "Given array is \n";
    for (int i = 0; i < arr_size; i++)
        cout << array1[i] << " ";
    cout << endl;

    mergeSort(array1, 0, arr_size - 1);

    cout << "\nSorted array is \n";
    for (int i = 0; i < arr_size; i++)
        cout << array1[i] << " ";
    cout << endl;
    return 0;
}
```



MAC Address for this System: