**Name: Nishant Kumar Giri**

Roll No: AC-1254

Course Name: BSc (Hons) Computer Science

Paper Name: Operating Systems Practical Exam (2021 – 22)

Semester: 3rd

CODE

*/\**

*Write a multhreaded program that calculates*

*Average, Minimum and maximum from a list*

*of numbers.*

*\*/*

#include <stdio.h>

#include <stdlib.h>

#include <pthread.h>

*int* arr[50], i; *// array to store numbers and i is used to iterarion*

*int* n;          *// this variable is being used to store the input*

*/\**

*average, minimum, and maximum*

*values will be stored globally*

*\*/*

*float* average;

*int* minimum;

*int* maximum;

*//thread1 for calculating average*

*void* \*th()

{

*int* sum = 0;

    for (i = 1; i <= n; i++)

    {

        sum = sum + arr[i];

        average = sum / n;

    }

}

*//thread2 for calculating minimum value*

*void* \*th1()

{

    minimum = arr[1];

    for (*int* i = 1; i < n; i++)

    {

        if (minimum > arr[i])

        {

            minimum = arr[i];

        }

    }

}

*//thread3 for calculating maximum value*

*void* \*th2()

{

    maximum = arr[1];

    for (*int* i = 1; i <= n; i++)

    {

        if (maximum < arr[i])

        {

            maximum = arr[i];

        }

    }

}

*int* main(*int* *argc*, *char* \**argv[]*)

{

*int* count = 0;

*/\**

*a series of numbers on the command line is passing*

*\*/*

    for (*int* i = 1; i < *argc*; i++)

    {

        arr[i] = atoi(*argv*[i]);

        count++;

    }

    n = count;

    printf("%d numbers has been entered using command line arguments  \n", count);

    for (*int* i = 1; i <= n; i++)

    {

        printf("%d\t", arr[i]);

    }

    printf("\n\n");

*int* t, i;

*//three objectof worker threads are t1 ,t2 and t3*

*pthread\_t* t1;

*pthread\_t* t2;

*pthread\_t* t3;

*//creating threads*

    t = pthread\_create(&t1, NULL, &th, NULL);

    pthread\_join(t1, NULL);

    t = pthread\_create(&t2, NULL, &th1, NULL);

    pthread\_join(t2, NULL);

    t = pthread\_create(&t3, NULL, &th2, NULL);

    pthread\_join(t3, NULL);

*/\*main is the parent thread.*

*the parent thread will output the values*

*once the workers have exited.*

*\*/*

    printf("Average Value: %f", average);

    printf("\nMinimum Value: %d", minimum);

    printf("\nMaximum Value: %d", maximum);

    return 0;

}

OUTPUT

Text

Description automatically generated