UCS1411 Operating Systems Lab
AY: 2021-22

Name: Krithika Swaminathan
Roll No.: 205001057

Assignment 11 – Implementation of Threads

Date: 23/05/2022

Aim:

To create a multithreaded program that calculates various statistical values for a list of numbers passed as command line arguments.

Algorithm:

1. Start
2. Initialize the global variables sum, min, max and avg to zero.
3. Define functions to compute the minimum, maximum and average values of a given list of numbers.
4. In the main function, define the thread identifiers and initialize the thread attributes.
5. Read a list of elements from the command line arguments.
6. Create threads for each function.
7. Wait for the threads to close using the join function.
8. Print the results of the computations performed by each thread.
9. Stop
10.

Code:

```
//Program to implement threads
#include <pthread.h>
#include <stdio.h>
#include <stdlib.h>

int size;/* this data is shared by the thread(s) */
int arr[50];
float avgresult = 0, minresult = 0, maxresult = 0;

/* threads call this function */
void *avg()
{
  float sum = 0;
  for (int i = 0; i<size; i++)
    sum+=arr[i];
  avgresult = sum/size;
}

void *min()
{
  int minind;
```

```
  minind = 0;
  for (int i = 0 ; i<size; i++)
  {
    if (arr[minind] > arr[i])
      minind = i;
  }
  minresult = arr[minind];
}

void *max()
{
  int maxind;
  maxind = 0;
  for (int i = 0 ; i<size; i++)
  {
    if (arr[maxind] < arr[i])
      maxind = i;
  }
  maxresult = arr[maxind];
}

int main(int argc, char *argv[])
{
  pthread_t tid1; /* the thread identifier */
  pthread_attr_t attr1;
  pthread_t tid2; /* the thread identifier */
  pthread_attr_t attr2;
  pthread_t tid3; /* the thread identifier */
  pthread_attr_t attr3;
  pthread_attr_init(&attr1);
  pthread_attr_init(&attr2);
  pthread_attr_init(&attr3);
  //getting input
  size = argc - 1;
  for (int i = 0 ; i < size; i++)
    arr[i] = atoi(argv[i+1]);
  /* create the thread */
  pthread_create(&tid1,&attr1,avg,NULL);
  pthread_create(&tid2,&attr2,min,NULL);
  pthread_create(&tid3,&attr3,max,NULL);
  /* wait for the thread to exit */
  pthread_join(tid1,NULL);
  pthread_join(tid2,NULL);
  pthread_join(tid3,NULL);
  printf("\n");
  printf("The average value is %.2f\n", avgresult);
  printf("The minimum value is %.2f\n", minresult);
  printf("The maximum value is %.2f\n", maxresult);
}
```

Name: Krithika Swaminathan
Roll No.: 205001057

Output:

```
~/OSL$ ./a11 90 81 78 95 79 72 85

The average value is 82.86
The minimum value is 72.00
The maximum value is 95.00
~/OSL$
```

Learning outcomes:

- Concurrent execution using threads was understood and implemented.
- Manipulation of threads using the pthread library was understood.
