Department of Computer Science and Engineering

S.G.Shivanirudh , 185001146, Semester IV

9 April 2020

UCS1411 - Operating Systems Laboratory

Lab Exercise 13: Threading Applications

Objective:

Develop a C multithreaded program that calculates various statistical values for a list of numbers.

Code:

Q.To Write a multithreaded program that calculates various statistical values for a list of numbers. This program will be passed a series of numbers on the command line and will then create three separate worker threads. One thread will determine the average of the numbers, the second will determine the maximum value, and the third will determine the minimum value.

```c
#include <pthread.h>
#include <stdio.h>
#include<stdlib.h>
#include<string.h>

double avg;
int max;
int min;

void *average(void *val){
    int sum=0;
    int *ival=(int*)val;

    for(int i=1;i<ival[0];i++){
        sum+=ival[i];
    }

    avg=sum/(ival[0]-1);
    pthread_exit(0);
}

void *maximum(void *val){
    int *ival=(int*)val;
    max=ival[1];
    for(int i=1;i<ival[0];i++)
        if(max<ival[i])
            max=ival[i];

    pthread_exit(0);
}

void *minimum(void *val){
    int *ival=(int*)val;
    min=ival[1];
    for(int i=1;i<ival[0];i++)
        if(min>ival[i])
            min=ival[i];

    pthread_exit(0);
}

int main(int argc,char *argv[]){
    pthread_t thread1;
    pthread_t thread2;
    pthread_t thread3;
```

```
46
47      if (argc < 2){
48          fprintf(stderr,"usage: a.out <integer value>\n");
49          return -1;
50      }
51
52      int *val = calloc(100,sizeof(int));
53      val[0]=argc;
54      for(int i=1;i<argc;i++){
55          int x=atoi(argv[i]);
56          val[i]=x;
57      }
58
59      pthread_create(&thread1,NULL,average,val);
60      pthread_create(&thread2,NULL,maximum,val);
61      pthread_create(&thread3,NULL,minimum,val);
62
63      pthread_join(thread1,NULL);
64      pthread_join(thread2,NULL);
65      pthread_join(thread3,NULL);
66
67      printf("\nAverage is: %.2lf\n",avg);
68      printf("\nMaximum value is: %d\n",max);
69      printf("\nMinimum value is: %d\n",min);
70
71  }
```

Output:

```
1
2  ./a 45 35 24 61 85 94 12 10
3
4  Average is: 45.00
5
6  Maximum value is: 94
7
8  Minimum value is: 10
```
