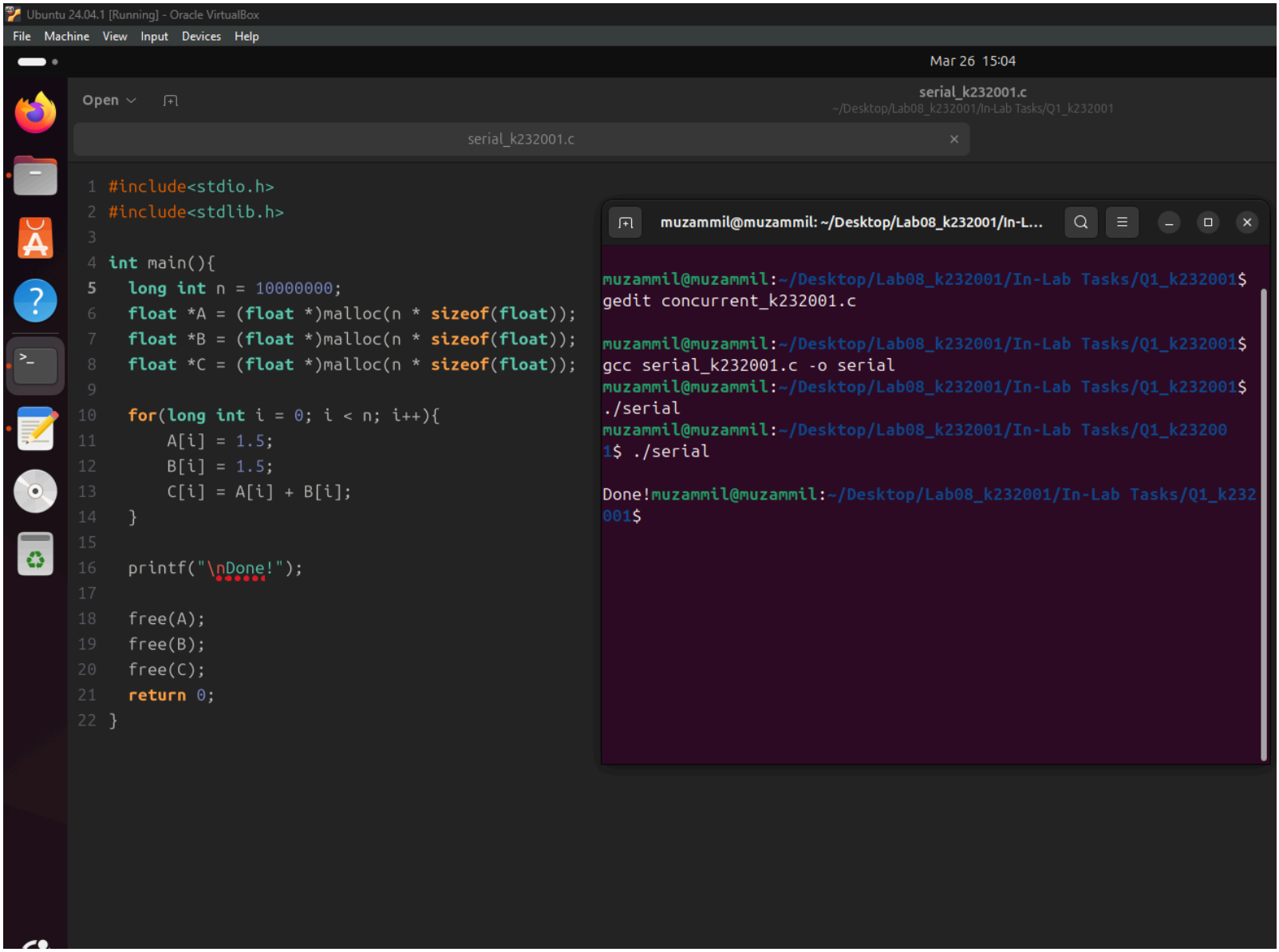
# Operating Systems

## LAB#08 (In-LAB Tasks)


**23K-2001**

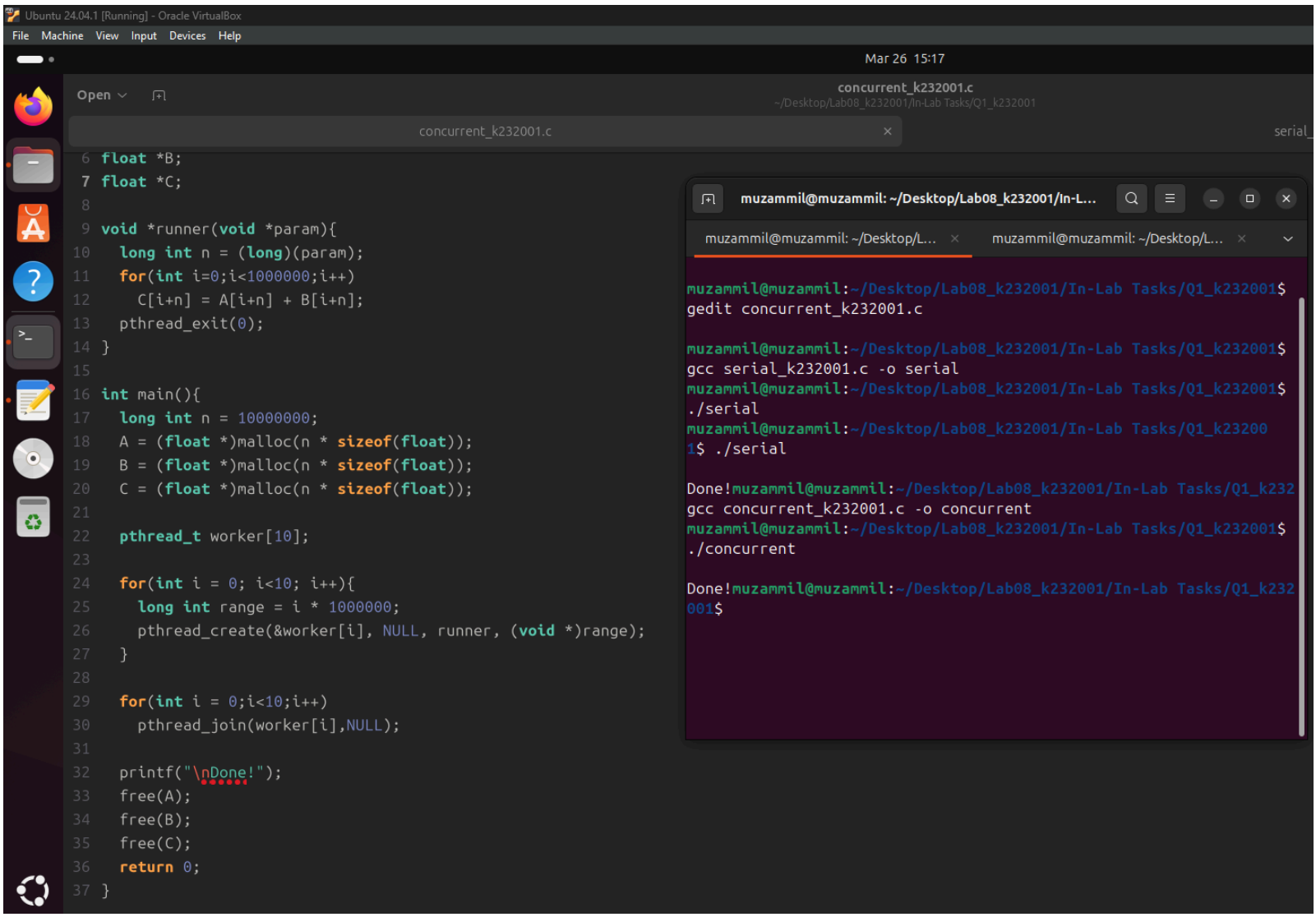
**BCS-4J**

# Q1:


Ubuntu 24.04.1 [Running] - Oracle VirtualBox
File Machine View Input Devices Help
Mar 26 15:04
Open
serial_k232001.c
~/Desktop/Lab08_k232001/In-Lab Tasks/Q1_k232001
serial_k232001.c
1 #include<stdio.h>
2 #include<stdlib.h>
3
4 int main(){
5 long int n = 10000000;
6 float *A = (float *)malloc(n * sizeof(float));
7 float *B = (float *)malloc(n * sizeof(float));
8 float *C = (float *)malloc(n * sizeof(float));
9
10 for(long int i = 0; i < n; i++){
11 A[i] = 1.5;
12 B[i] = 1.5;
13 C[i] = A[i] + B[i];
14 }
15
16 printf("\nDone!");
17
18 free(A);
19 free(B);
20 free(C);
21 return 0;
22 }
muzammil@muzammil: ~/Desktop/Lab08_k232001/In-L...
muzammil@muzammil:~/Desktop/Lab08_k232001/In-Lab Tasks/Q1_k232001$
gedit concurrent_k232001.c
muzammil@muzammil:~/Desktop/Lab08_k232001/In-Lab Tasks/Q1_k232001$
gcc serial_k232001.c -o serial
muzammil@muzammil:~/Desktop/Lab08_k232001/In-Lab Tasks/Q1_k232001$
./serial
muzammil@muzammil:~/Desktop/Lab08_k232001/In-Lab Tasks/Q1_k232001$ ./serial
Done!muzammil@muzammil:~/Desktop/Lab08_k232001/In-Lab Tasks/Q1_k232001$

a) Serial Code:

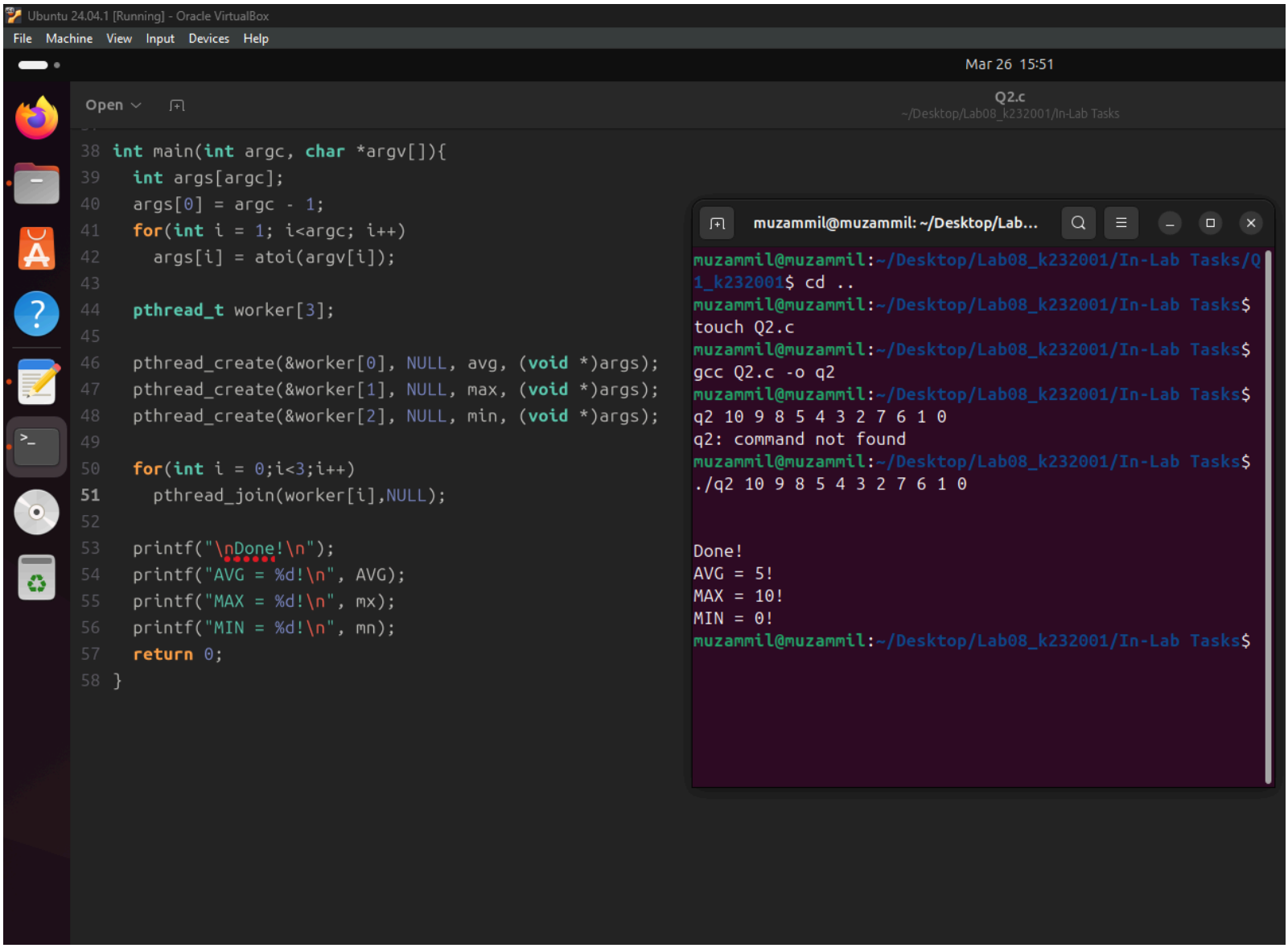
```
#include<stdio.h>
#include<stdlib.h>

int main(){
  long int n = 10000000;
  float *A = (float *)malloc(n * sizeof(float));
  float *B = (float *)malloc(n * sizeof(float));
  float *C = (float *)malloc(n * sizeof(float));

  for(long int i = 0; i < n; i++){
          A[i] = 1.5;
          B[i] = 1.5;
          C[i] = A[i] + B[i];
  }

  printf("\nDone!");

  free(A);
  free(B);
  free(C);
  return 0;
}
```


Ubuntu 24.04.1 [Running] - Oracle VirtualBox
File Machine View Input Devices Help
Mar 26 15:17
Open
concurrent_k232001.c
~/Desktop/Lab08_k232001/In-Lab Tasks/Q1_k232001
concurrent_k232001.c
serial_
float *B;
float *C;
void *runner(void *param){
long int n = (long)(param);
for(int i=0;i<1000000;i++)
C[i+n] = A[i+n] + B[i+n];
pthread_exit(0);
}
int main(){
long int n = 10000000;
A = (float *)malloc(n * sizeof(float));
B = (float *)malloc(n * sizeof(float));
C = (float *)malloc(n * sizeof(float));
pthread_t worker[10];
for(int i = 0; i<10; i++){
long int range = i * 1000000;
pthread_create(&worker[i], NULL, runner, (void *)range);
}
for(int i = 0;i<10;i++)
pthread_join(worker[i],NULL);
printf("\nDone!");
free(A);
free(B);
free(C);
return 0;
}
muzammil@muzammil: ~/Desktop/Lab08_k232001/In-L...
muzammil@muzammil: ~/Desktop/L...
muzammil@muzammil: ~/Desktop/L...
muzammil@muzammil:~/Desktop/Lab08_k232001/In-Lab Tasks/Q1_k232001$
gedit concurrent_k232001.c
muzammil@muzammil:~/Desktop/Lab08_k232001/In-Lab Tasks/Q1_k232001$
gcc serial_k232001.c -o serial
muzammil@muzammil:~/Desktop/Lab08_k232001/In-Lab Tasks/Q1_k232001$
./serial
muzammil@muzammil:~/Desktop/Lab08_k232001/In-Lab Tasks/Q1_k23200
1$ ./serial
Done!muzammil@muzammil:~/Desktop/Lab08_k232001/In-Lab Tasks/Q1_k232
gcc concurrent_k232001.c -o concurrent
muzammil@muzammil:~/Desktop/Lab08_k232001/In-Lab Tasks/Q1_k232001$
./concurrent
Done!muzammil@muzammil:~/Desktop/Lab08_k232001/In-Lab Tasks/Q1_k232
001$

b) Concurrent code:

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

float *A;
float *B;
float *C;

void *runner(void *param){
  long int n = (long)(param);
  for(int i=0;i<1000000;i++)
        C[i+n] = A[i+n] + B[i+n];
  pthread_exit(0);
}

int main(){
  long int n = 10000000;
  A = (float *)malloc(n * sizeof(float));
  B = (float *)malloc(n * sizeof(float));
  C = (float *)malloc(n * sizeof(float));

  pthread_t worker[10];

  for(int i = 0; i<10; i++){
        long int range = i * 1000000;
        pthread_create(&worker[i], NULL, runner, (void *)range);
  }

  for(int i = 0;i<10;i++)
        pthread_join(worker[i],NULL);

  printf("\nDone!");
  free(A);
  free(B);
  free(C);
  return 0;
}
```

# Q2:


Ubuntu 24.04.1 [Running] - Oracle VirtualBox
File Machine View Input Devices Help
Mar 26 15:51
Open
Q2.c
~/Desktop/Lab08_k232001/In-Lab Tasks
38 int main(int argc, char *argv[]){
39 int args[argc];
40 args[0] = argc - 1;
41 for(int i = 1; i<argc; i++)
42 args[i] = atoi(argv[i]);
43
44 pthread_t worker[3];
45
46 pthread_create(&worker[0], NULL, avg, (void *)args);
47 pthread_create(&worker[1], NULL, max, (void *)args);
48 pthread_create(&worker[2], NULL, min, (void *)args);
49
50 for(int i = 0;i<3;i++)
51 pthread_join(worker[i],NULL);
52
53 printf("\nDone!\n");
54 printf("AVG = %d!\n", AVG);
55 printf("MAX = %d!\n", mx);
56 printf("MIN = %d!\n", mn);
57 return 0;
58 }
muzammil@muzammil: ~/Desktop/Lab...
muzammil@muzammil:~/Desktop/Lab08_k232001/In-Lab Tasks/Q1_k232001$ cd ..
muzammil@muzammil:~/Desktop/Lab08_k232001/In-Lab Tasks$ touch Q2.c
muzammil@muzammil:~/Desktop/Lab08_k232001/In-Lab Tasks$ gcc Q2.c -o q2
muzammil@muzammil:~/Desktop/Lab08_k232001/In-Lab Tasks$ q2 10 9 8 5 4 3 2 7 6 1 0
q2: command not found
muzammil@muzammil:~/Desktop/Lab08_k232001/In-Lab Tasks$ ./q2 10 9 8 5 4 3 2 7 6 1 0
Done!
AVG = 5!
MAX = 10!
MIN = 0!
muzammil@muzammil:~/Desktop/Lab08_k232001/In-Lab Tasks$

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

int AVG, mx, mn;

void *avg(void *param){
  int *arr = (int *)param;
  int n = arr[0];
  int sum = 0;
  for(int i = 1; i<=n; i++)
          sum += arr[i];

  AVG = sum/n;
  pthread_exit(0);
}
void *max(void *param){
  int *arr = (int *)param;
  int n = arr[0];
  mx = arr[1];
  for(int i = 2; i<=n; i++){
          if (arr[i] > mx)
          mx= arr[i];
  }
  pthread_exit(0);
}
void *min(void *param){
  int *arr = (int *)param;
  int n = arr[0];
  mn = arr[1];
  for(int i = 2; i<=n; i++){
          if (arr[i] < mn)
          mn = arr[i];
  }
  pthread_exit(0);
}

int main(int argc, char *argv[]){
  int args[argc];
  args[0] = argc - 1;
  for(int i = 1; i<argc; i++)
          args[i] = atoi(argv[i]);

  pthread_t worker[3];
```

```
  pthread_create(&worker[0], NULL, avg, (void *)args);
  pthread_create(&worker[1], NULL, max, (void *)args);
  pthread_create(&worker[2], NULL, min, (void *)args);

  for(int i = 0;i<3;i++)
          pthread_join(worker[i],NULL);

  printf("\nDone!\n");
  printf("AVG = %d!\n", AVG);
  printf("MAX = %d!\n", mx);
  printf("MIN = %d!\n", mn);
  return 0;
}
```