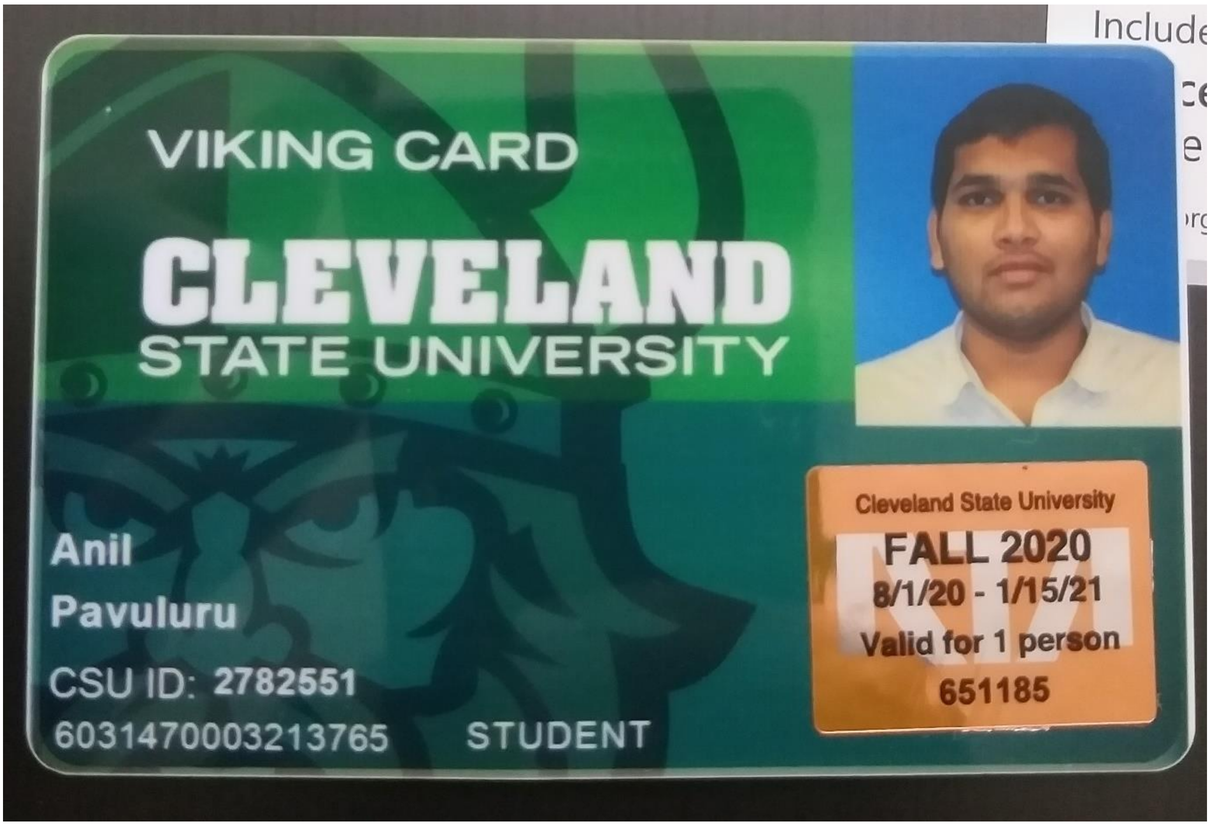
Name: Anil Pavuluru

CSU ID:2782551

Login id: anpavulu



VIKING CARD

CLEVELAND
STATE UNIVERSITY

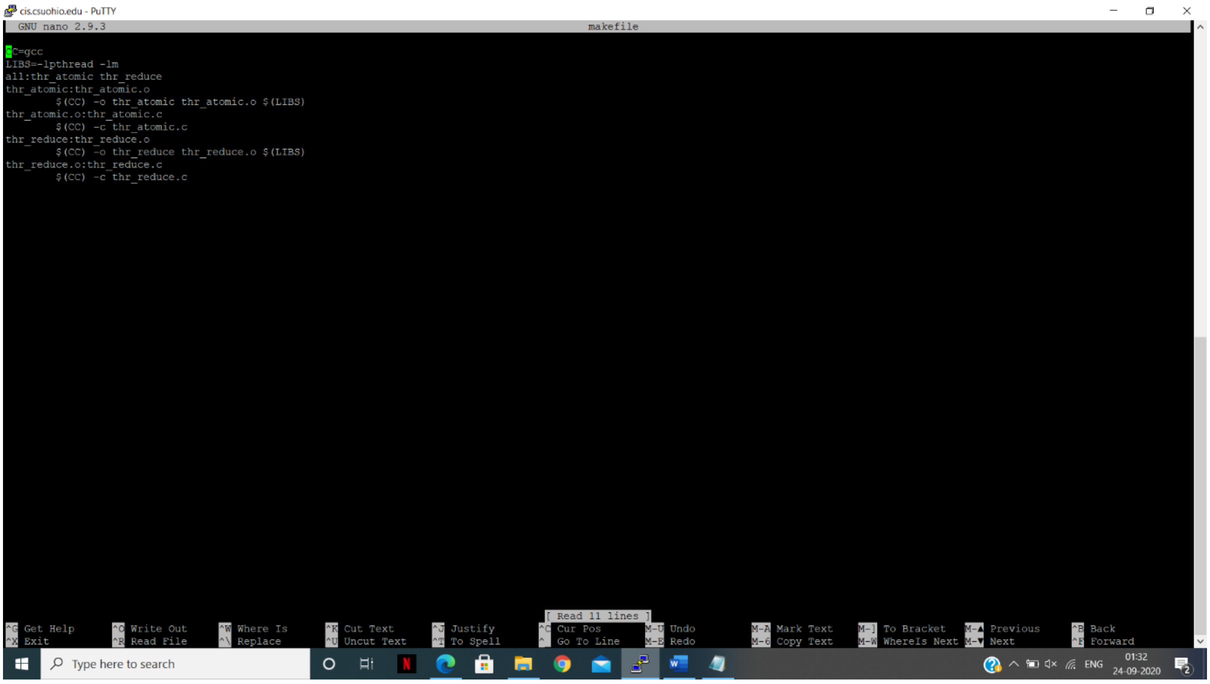
Anil
Pavuluru

CSU ID: 2782551

6031470003213765 STUDENT

Cleveland State University
FALL 2020
8/1/20 - 1/15/21
Valid for 1 person
651185

1)**MAKING FILE**:



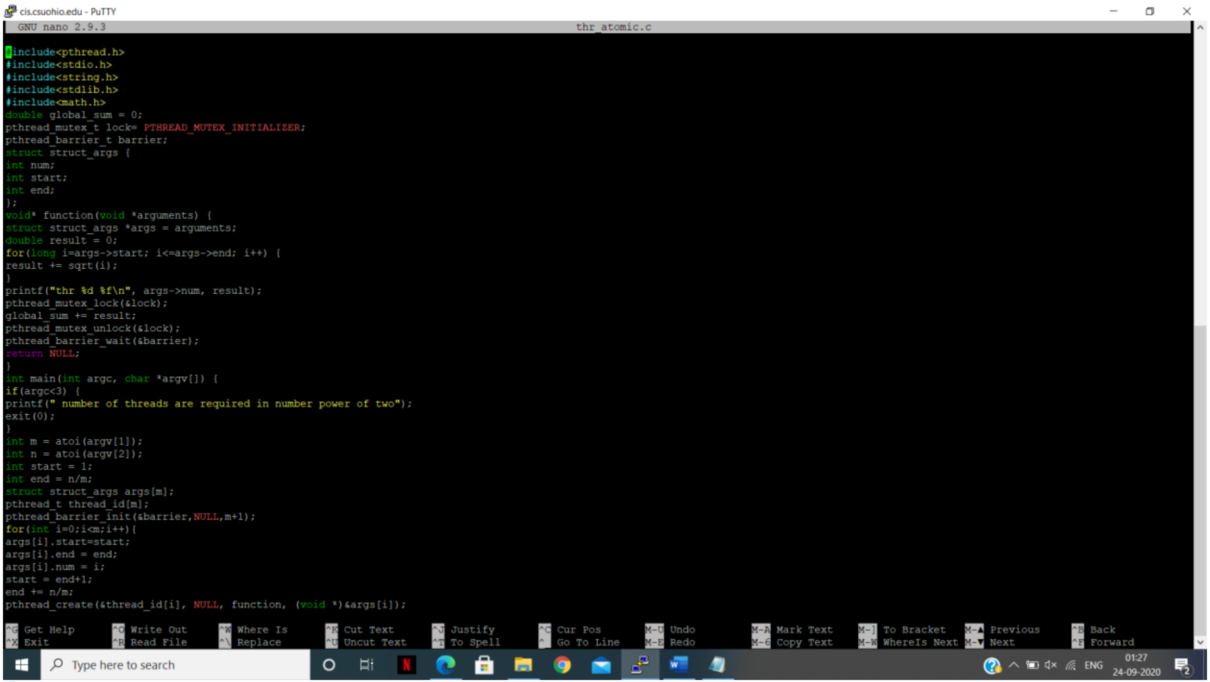
```
CC=gcc
LIBS=-lpthread -lm
all:thr_atomic thr_reduce
thr_atomic:thr_atomic.o
        $(CC) -o thr_atomic thr_atomic.o $(LIBS)
thr_atomic.o:thr_atomic.c
        $(CC) -c thr_atomic.c
thr_reduce:thr_reduce.o
        $(CC) -o thr_reduce thr_reduce.o $(LIBS)
thr_reduce.o:thr_reduce.c
        $(CC) -c thr_reduce.c
```

```
[ Read 11 lines ]
```

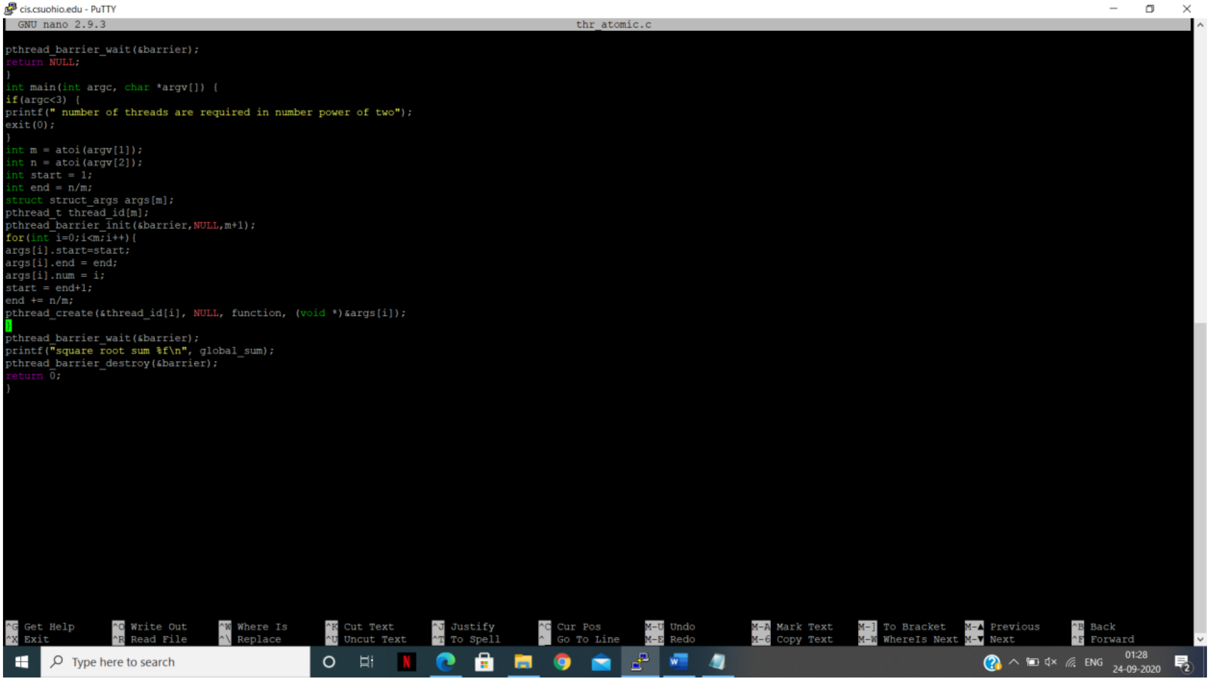
Experience and Explanation:

The most liked part in this proj1 is the makefile.it was pretty nice to write the steps very easy because all the steps you mention in the class lecture of project-1 by following the video I done this part.

2)thr_atomic.c



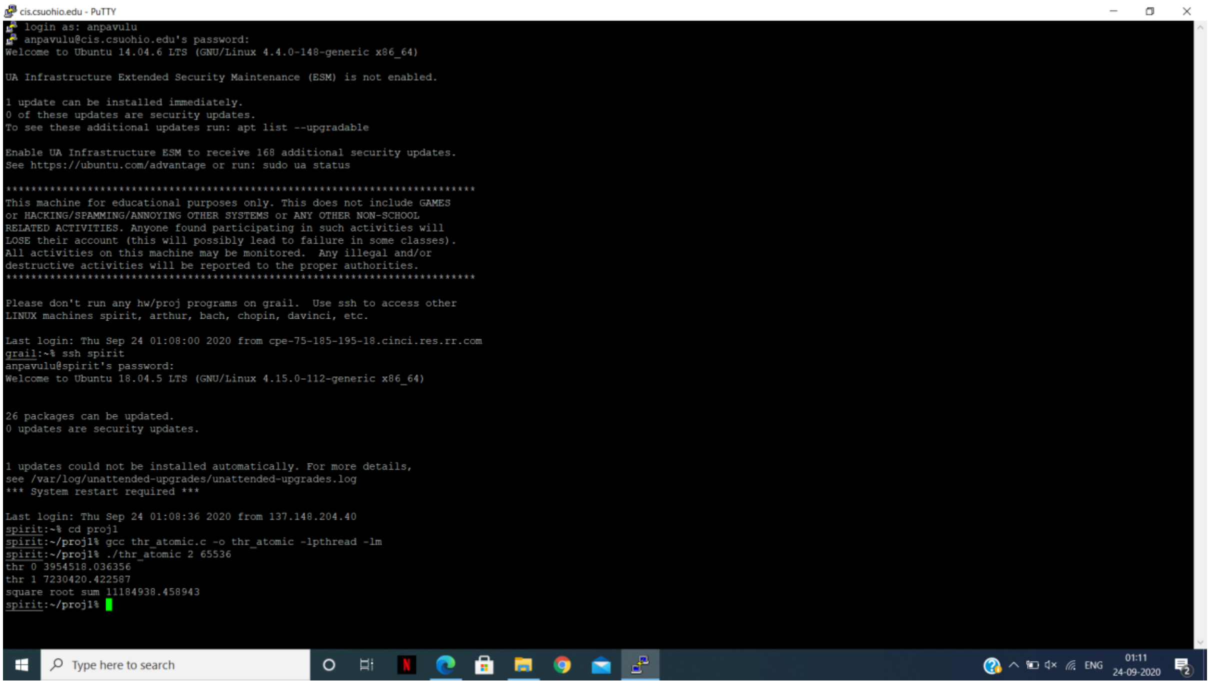
```
#include<pthread.h>
#include<stdio.h>
#include<string.h>
#include<stdlib.h>
#include<math.h>
double global_sum = 0;
pthread_mutex_t lock= PTHREAD_MUTEX_INITIALIZER;
pthread_barrier_t barrier;
struct struct_args {
int num;
int start;
int end;
};
void* function(void *arguments) {
struct struct_args *args = arguments;
double result = 0;
for(long i=args->start; i<=args->end; i++) {
result += sqrt(i);
}
printf("thr %d %f\n", args->num, result);
pthread_mutex_lock(&lock);
global_sum += result;
pthread_mutex_unlock(&lock);
pthread_barrier_wait(&barrier);
return NULL;
}
int main(int argc, char *argv[]) {
if(argc<3) {
printf(" number of threads are required in number power of two");
exit(0);
}
int m = atoi(argv[1]);
int n = atoi(argv[2]);
int start = 1;
int end = n/m;
struct struct_args args[m];
pthread_t thread_id[m];
pthread_barrier_init(&barrier,NULL,m+1);
for(int i=0;i<m;i++){
args[i].start=start;
args[i].end = end;
args[i].num = i;
start = end+1;
end += n/m;
pthread_create(&thread_id[i], NULL, function, (void *)&args[i]);
```



```
pthread_barrier_wait(&barrier);
return NULL;
}
int main(int argc, char *argv[]) {
if(argc<3) {
printf(" number of threads are required in number power of two");
exit(0);
}
int m = atoi(argv[1]);
int n = atoi(argv[2]);
int start = 1;
int end = n/m;
struct struct_args args[m];
pthread_t thread_id[m];
pthread_barrier_init(&barrier,NULL,m+1);
for(int i=0;i<m;i++){
args[i].start=start;
args[i].end = end;
args[i].num = i;
start = end+1;
end += n/m;
pthread_create(&thread_id[i], NULL, function, (void *)&args[i]);
}
pthread_barrier_wait(&barrier);
printf("square root sum %f\n", global_sum);
pthread_barrier_destroy(&barrier);
return 0;
}
```

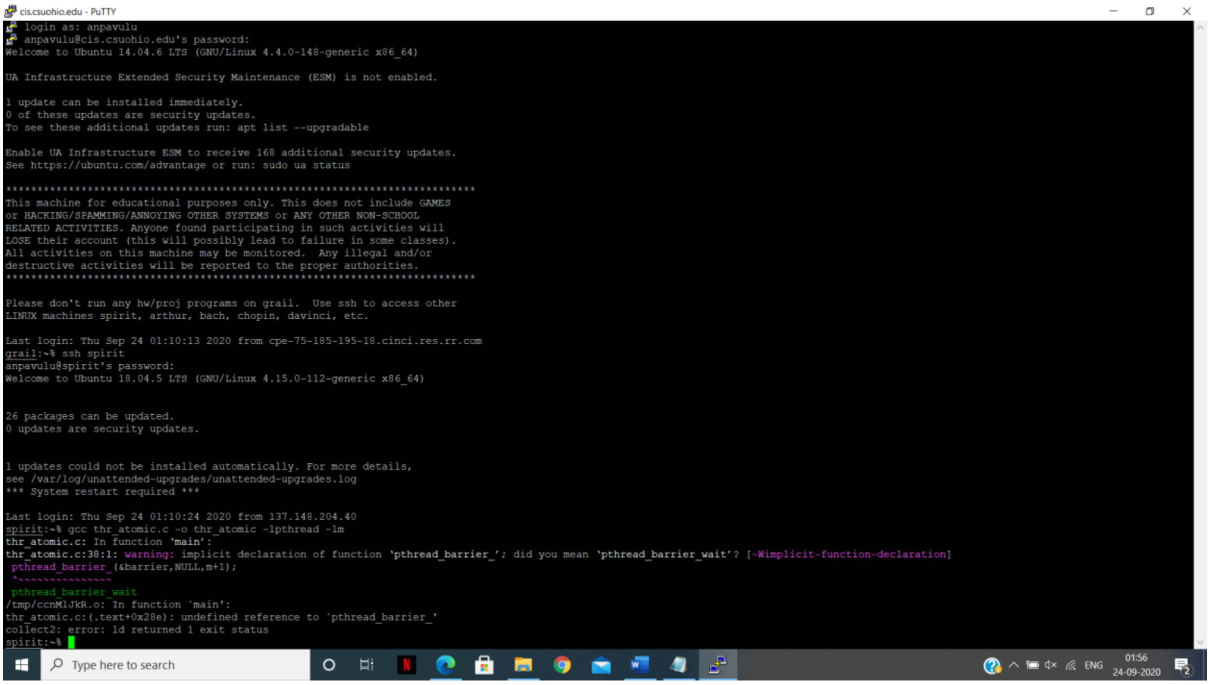
Coding explaination:First I written defined libraries file like thread,standardlibrary,string,input and output and math for the calculation of sqrt.next initialized with global sum it store the partial sum of all the threads and passing command line arguments,struct array,mutex_intializer for locking of thread to complete one processor and then go to the other. I locked at global sum.then after in the main first know the range and threads then divide with threads in the four loop we have to run with logic as mentioned above.

Output:



```
login as: anpavulu
anpavulu@cis.csuohio.edu's password:
Welcome to Ubuntu 14.04.6 LTS (GNU/Linux 4.4.0-148-generic x86_64)

UA Infrastructure Extended Security Maintenance (ESM) is not enabled.

1 update can be installed immediately.
0 of these updates are security updates.
To see these additional updates run: apt list --upgradable

Enable UA Infrastructure ESM to receive 168 additional security updates.
See https://ubuntu.com/advantage or run: sudo ua status

***************************************************************************
This machine for educational purposes only. This does not include GAMES
or HACKING/SPAMMING/ANNOYING OTHER SYSTEMS or ANY OTHER NON-SCHOOL
RELATED ACTIVITIES. Anyone found participating in such activities will
LOSE their account (this will possibly lead to failure in some classes).
All activities on this machine may be monitored.  Any illegal and/or
destructive activities will be reported to the proper authorities.
***************************************************************************

Please don't run any hw/proj programs on grail.  Use ssh to access other
LINUX machines spirit, arthur, bach, chopin, davinci, etc.

Last login: Thu Sep 24 01:08:00 2020 from cpe-75-185-195-18.cinci.res.rr.com
grail:~% ssh spirit
anpavulu@spirit's password:
Welcome to Ubuntu 18.04.5 LTS (GNU/Linux 4.15.0-112-generic x86_64)


26 packages can be updated.
0 updates are security updates.


1 updates could not be installed automatically. For more details,
see /var/log/unattended-upgrades/unattended-upgrades.log
*** System restart required ***

Last login: Thu Sep 24 01:08:36 2020 from 137.148.204.40
spirit:~% cd proj1
spirit:~/proj1% gcc thr_atomic.c -o thr_atomic -lpthread -lm
spirit:~/proj1% ./thr_atomic 2 65536
thr 0 3954518.036356
thr 1 7230420.422587
square root sum 11184938.458943
spirit:~/proj1%
```

Experience:



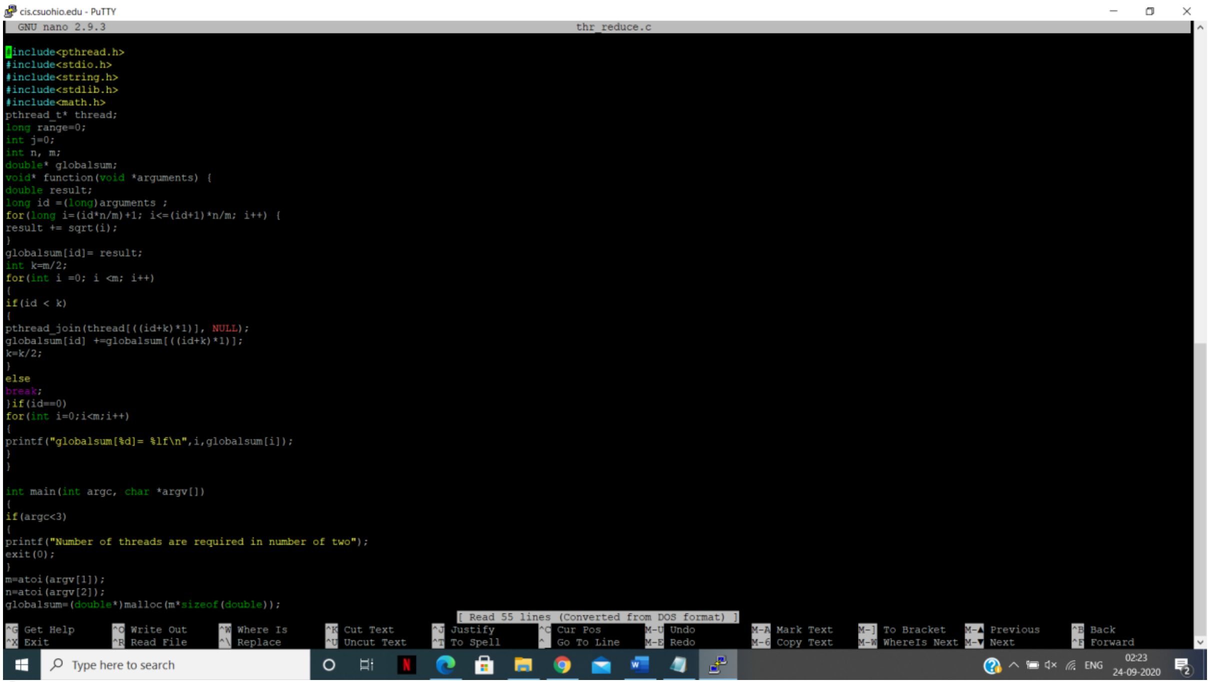
```
login as: anpavulu
anpavulu@cis.csuohio.edu's password:
Welcome to Ubuntu 14.04.6 LTS (GNU/Linux 4.4.0-148-generic x86_64)

UA Infrastructure Extended Security Maintenance (ESM) is not enabled.

1 update can be installed immediately.
0 of these updates are security updates.
To see these additional updates run: apt list --upgradable

Enable UA Infrastructure ESM to receive 168 additional security updates.
See https://ubuntu.com/advantage or run: sudo ua status

***************************************************************************
This machine for educational purposes only. This does not include GAMES
or HACKING/SPAMMING/ANNOYING OTHER SYSTEMS or ANY OTHER NON-SCHOOL
RELATED ACTIVITIES. Anyone found participating in such activities will
LOSE their account (this will possibly lead to failure in some classes).
All activities on this machine may be monitored.  Any illegal and/or
destructive activities will be reported to the proper authorities.
***************************************************************************

Please don't run any hw/proj programs on grail.  Use ssh to access other
LINUX machines spirit, arthur, bach, chopin, davinci, etc.

Last login: Thu Sep 24 01:10:13 2020 from cpe-75-185-195-18.cinci.res.rr.com
grail:~% ssh spirit
anpavulu@spirit's password:
Welcome to Ubuntu 18.04.5 LTS (GNU/Linux 4.15.0-112-generic x86_64)


26 packages can be updated.
0 updates are security updates.


1 updates could not be installed automatically. For more details,
see /var/log/unattended-upgrades/unattended-upgrades.log
*** System restart required ***

Last login: Thu Sep 24 01:10:24 2020 from 137.148.204.40
spirit:~% gcc thr_atomic.c -o thr_atomic -lpthread -lm
thr_atomic.c: In function ‘main’:
thr_atomic.c:38:1: warning: implicit declaration of function ‘pthread_barrier_’; did you mean ‘pthread_barrier_wait’? [-Wimplicit-function-declaration]
 pthread_barrier_(&barrier,NULL,m+1);
 ^~~~~~~~~~~~~~~
 pthread_barrier_wait
/tmp/ccnMlJkR.o: In function `main':
thr_atomic.c:(.text+0x28e): undefined reference to `pthread_barrier_'
collect2: error: ld returned 1 exit status
spirit:~%
```

pthread_barrier_init(&barrier,NULL,m+1);

this init trouble me for some time later search in the google and fix it.

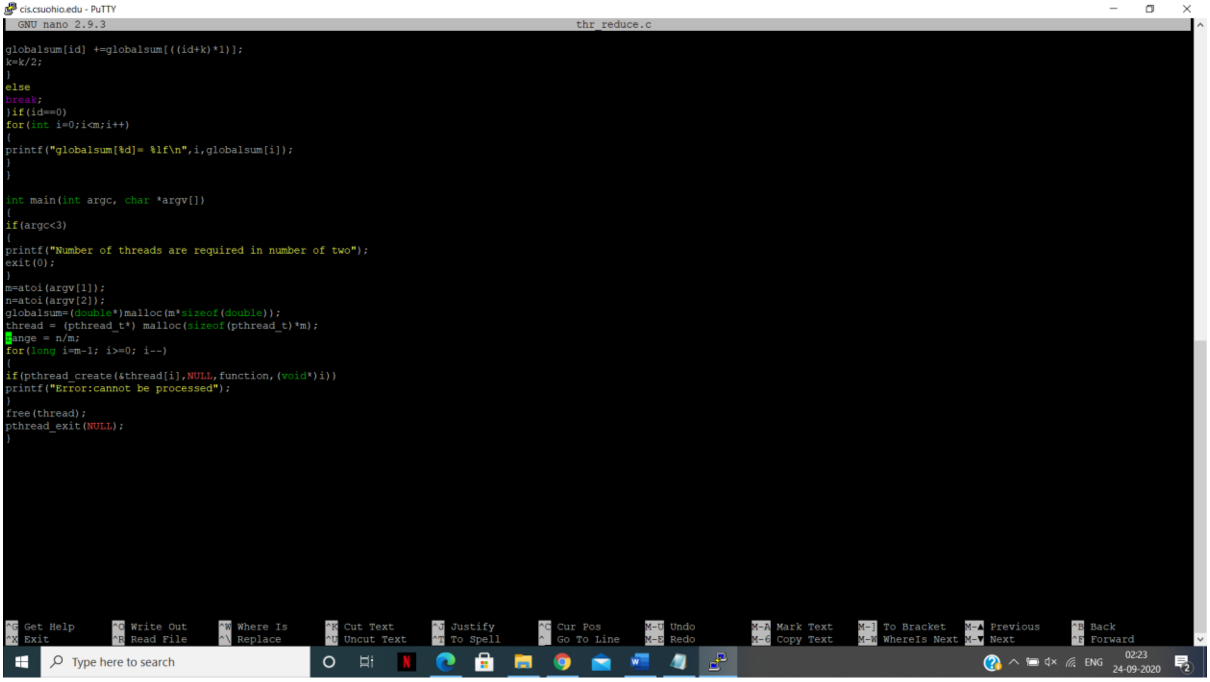
By listening the lecture remaining all small syntaxes and some logical errors figure it out and get output overall some trouble but done with first part nicely.

3)thr_reduce.c:



```
#include<pthread.h>
#include<stdio.h>
#include<string.h>
#include<stdlib.h>
#include<math.h>
pthread_t* thread;
long range=0;
int j=0;
int n, m;
double* globalsum;
void* function(void *arguments) {
double result;
long id =(long)arguments ;
for(long i=(id*n/m)+1; i<=(id+1)*n/m; i++) {
result += sqrt(i);
}
globalsum[id]= result;
int k=m/2;
for(int i =0; i <m; i++)
{
if(id < k)
{
pthread_join(thread[((id+k)*1)], NULL);
globalsum[id] +=globalsum[((id+k)*1)];
k=k/2;
}
else
break;
}if(id==0)
for(int i=0;i<m;i++)
{
printf("globalsum[%d]= %lf\n",i,globalsum[i]);
}
}

int main(int argc, char *argv[])
{
if(argc<3)
{
printf("Number of threads are required in number of two");
exit(0);
}
m=atoi(argv[1]);
n=atoi(argv[2]);
globalsum=(double*)malloc(m*sizeof(double));
```

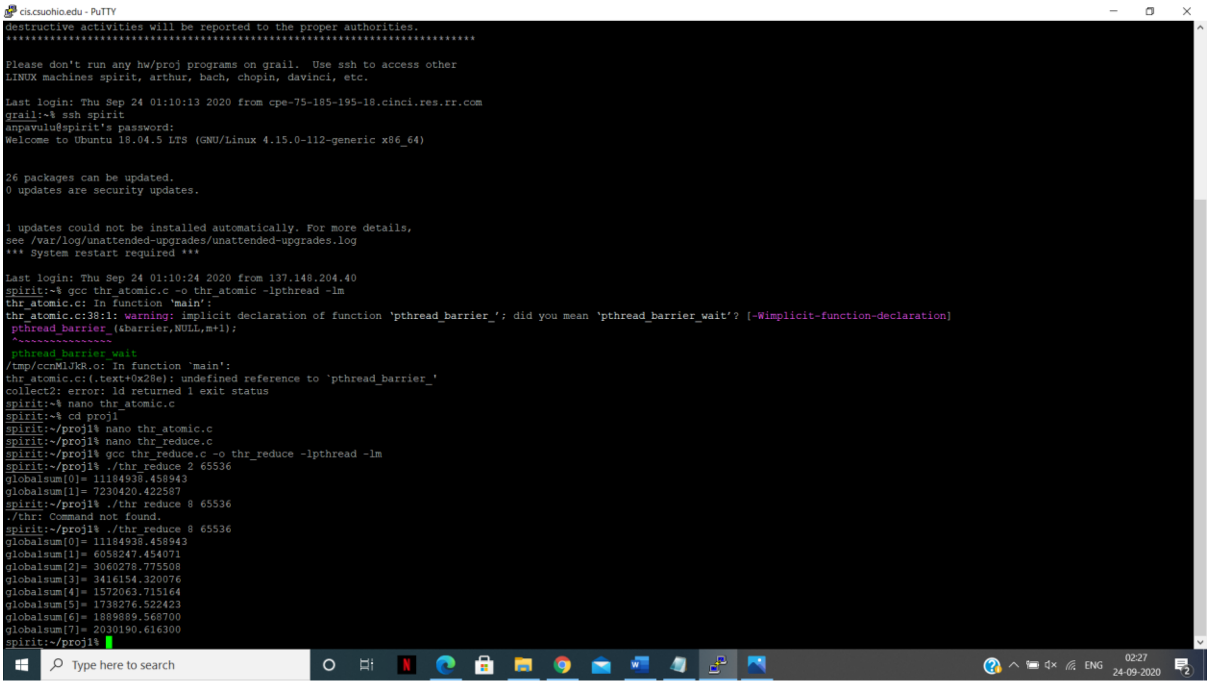
```
[ Read 55 lines (Converted from DOS format) ]
```



```
globalsum[id] +=globalsum[((id+k)*1)];
k=k/2;
}
else
break;
}if(id==0)
for(int i=0;i<m;i++)
{
printf("globalsum[%d]= %lf\n",i,globalsum[i]);
}
}

int main(int argc, char *argv[])
{
if(argc<3)
{
printf("Number of threads are required in number of two");
exit(0);
}
m=atoi(argv[1]);
n=atoi(argv[2]);
globalsum=(double*)malloc(m*sizeof(double));
thread = (pthread_t*) malloc(sizeof(pthread_t)*m);
range = n/m;
for(long i=m-1; i>=0; i--)
{
if(pthread_create(&thread[i],NULL,function,(void*)i))
printf("Error:cannot be processed");
}
free(thread);
pthread_exit(NULL);
}
```

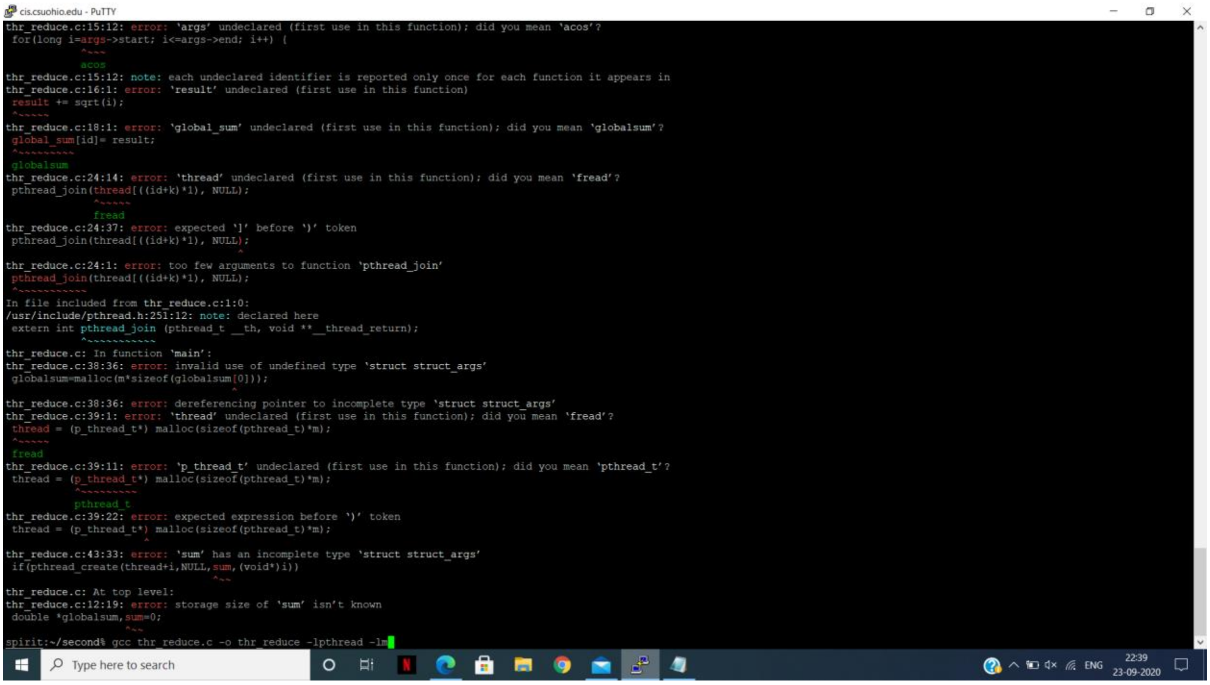
coding explanation: defined the libraries standard,,thread,string,and math. After that declare global variables and global sum and then divided with number of threads .after that setting up for loop with(id+k)*1 to wait for thr partner id and then again slices until for zero thread like that we will print all the values.the remaining flow will be same mentioned in the thr_atomic.c same logic is taken here I just modified for partner thread id only to join.

Output:



```
destructive activities will be reported to the proper authorities.
****************************************************************************

Please don't run any hw/proj programs on grail.  Use ssh to access other
LINUX machines spirit, arthur, bach, chopin, davinci, etc.

Last login: Thu Sep 24 01:10:13 2020 from cpe-75-185-195-18.cinci.res.rr.com
grail:~% ssh spirit
anpavulu@spirit's password:
Welcome to Ubuntu 18.04.5 LTS (GNU/Linux 4.15.0-112-generic x86_64)


26 packages can be updated.
0 updates are security updates.


1 updates could not be installed automatically. For more details,
see /var/log/unattended-upgrades/unattended-upgrades.log
*** System restart required ***

Last login: Thu Sep 24 01:10:24 2020 from 137.148.204.40
spirit:~% gcc thr_atomic.c -o thr_atomic -lpthread -lm
thr_atomic.c: In function ‘main’:
thr_atomic.c:38:1: warning: implicit declaration of function ‘pthread_barrier_’; did you mean ‘pthread_barrier_wait’? [-Wimplicit-function-declaration]
 pthread_barrier_(&barrier,NULL,m+1);
 ^~~~~~~~~~~~~~~~
 pthread_barrier_wait
/tmp/ccnMlJkR.o: In function `main':
thr_atomic.c:(.text+0x28e): undefined reference to `pthread_barrier_'
collect2: error: ld returned 1 exit status
spirit:~% nano thr_atomic.c
spirit:~% cd proj1
spirit:~/proj1% nano thr_atomic.c
spirit:~/proj1% nano thr_reduce.c
spirit:~/proj1% gcc thr_reduce.c -o thr_reduce -lpthread -lm
spirit:~/proj1% ./thr_reduce 2 65536
globalsum[0]= 11184938.458943
globalsum[1]= 7230420.422587
spirit:~/proj1% ./thr reduce 8 65536
./thr: Command not found.
spirit:~/proj1% ./thr_reduce 8 65536
globalsum[0]= 11184938.458943
globalsum[1]= 6058247.454071
globalsum[2]= 3060278.775508
globalsum[3]= 3416154.320076
globalsum[4]= 1572063.715164
globalsum[5]= 1738276.522423
globalsum[6]= 1889889.568700
globalsum[7]= 2030190.616300
spirit:~/proj1%
```

Experience:



```
thr_reduce.c:15:12: error: ‘args’ undeclared (first use in this function); did you mean ‘acos’?
 for(long i=args->start; i<=args->end; i++) {
            ^~~~
            acos
thr_reduce.c:15:12: note: each undeclared identifier is reported only once for each function it appears in
thr_reduce.c:16:1: error: ‘result’ undeclared (first use in this function)
 result += sqrt(i);
 ^~~~~~
thr_reduce.c:18:1: error: ‘global_sum’ undeclared (first use in this function); did you mean ‘globalsum’?
 global_sum[id]= result;
 ^~~~~~~~~~
 globalsum
thr_reduce.c:24:14: error: ‘thread’ undeclared (first use in this function); did you mean ‘fread’?
 pthread_join(thread[((id+k)*1), NULL);
              ^~~~~~
              fread
thr_reduce.c:24:37: error: expected ‘]’ before ‘)’ token
 pthread_join(thread[((id+k)*1), NULL);
                                     ^
thr_reduce.c:24:1: error: too few arguments to function ‘pthread_join’
 pthread_join(thread[((id+k)*1), NULL);
 ^~~~~~~~~~~~
In file included from thr_reduce.c:1:0:
/usr/include/pthread.h:251:12: note: declared here
 extern int pthread_join (pthread_t __th, void **__thread_return);
            ^~~~~~~~~~~~
thr_reduce.c: In function ‘main’:
thr_reduce.c:38:36: error: invalid use of undefined type ‘struct struct_args’
 globalsum=malloc(m*sizeof(globalsum[0]));
                                    ^
thr_reduce.c:38:36: error: dereferencing pointer to incomplete type ‘struct struct_args’
thr_reduce.c:39:1: error: ‘thread’ undeclared (first use in this function); did you mean ‘fread’?
 thread = (p_thread_t*) malloc(sizeof(pthread_t)*m);
 ^~~~~~
 fread
thr_reduce.c:39:11: error: ‘p_thread_t’ undeclared (first use in this function); did you mean ‘pthread_t’?
 thread = (p_thread_t*) malloc(sizeof(pthread_t)*m);
           ^~~~~~~~~~
           pthread_t
thr_reduce.c:39:22: error: expected expression before ‘)’ token
 thread = (p_thread_t*) malloc(sizeof(pthread_t)*m);
                      ^
thr_reduce.c:43:33: error: ‘sum’ has an incomplete type ‘struct struct_args’
 if(pthread_create(thread+i,NULL,sum,(void*)i))
                                 ^~~
thr_reduce.c: At top level:
thr_reduce.c:12:19: error: storage size of ‘sum’ isn’t known
 double *globalsum,sum=0;
                   ^~~
spirit:~/second% gcc thr_reduce.c -o thr_reduce -lpthread -lm
```

double* globalsum; firstly this declaration I messed with double globalsum[m]; as I already used an int “m”.here solved first error.

Second result is declared in function but not declared starting then after declared double result;

Third after if condition pthread_join(thread[((id+k)*1))],NULL); in these line closing the brackets I did mistake after that solved again.

Fourth allocating memory to threads using malloc that steps also got error

Fifth one declaring m,n as global and used as int m,int nin the main function how you can use global as local that errors.

Sixth declaring p_thread*t only I forgot to store in the pointer of thread later I declare . p_thread*t pthread.

I just mentioning the some errors which are irritated me because It become really hard and get panic by seeing that much error and warnings by seeing the errors knowing that error type and solving it become challenging a quite bit over all I was done finally.The lecture video helped me to get logical id especially when I stucked.how to write?later I got simply while listening twice and slowly understanding things happening in that .