**ARYAMAN MISHRA**

**19BCE1027**

**LAB EXERCISE 1**

**Q.1.Write a Program to implement a “Hello World” Program using OpenMP.**

#include<omp.h>

#include <stdio.h>

#include <stdlib.h>

int main()

{

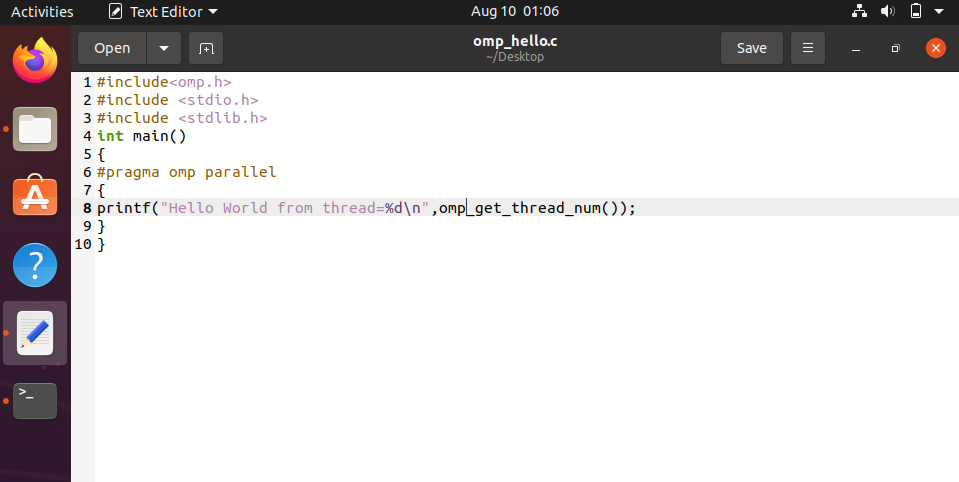
#pragma omp parallel

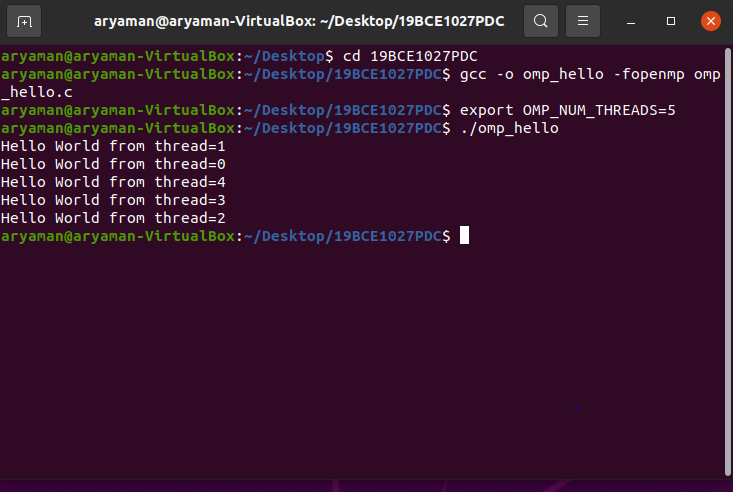
{

printf("Hello World from thread=%d\n",omp\_get\_thread\_num());

}

}





**Q.2.Write a Program to implement a “Hello World” with thread id and allow master thread to print the total number of threads.**

#include<omp.h>

#include <stdio.h>

#include <stdlib.h>

int main()

{

int nthreads,tid;

/\*Fork a team of threads woth each thread having a private tid variable\*/

#pragma omp parallel private(tid)

{

/\*Obtain and print thread id\*/

tid=omp\_get\_thread\_num();

printf("Hello World from thread=%d\n",tid);

/\*Only master thread does this\*/

if(tid==0)

{

nthreads=omp\_get\_num\_threads();

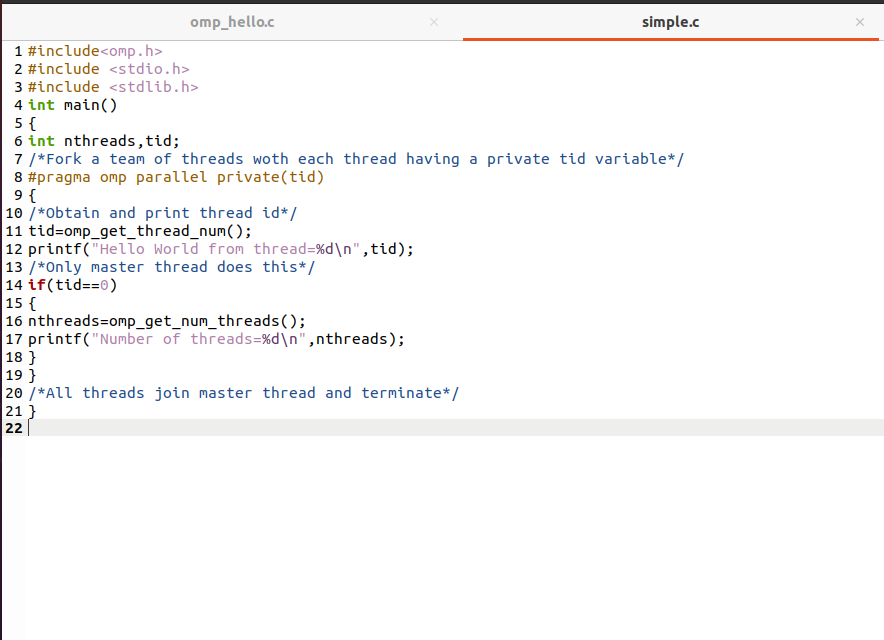
printf("Number of threads=%d\n",nthreads);

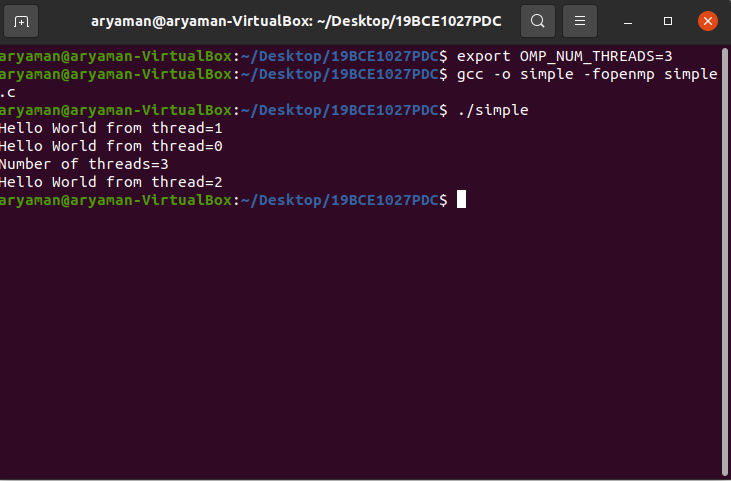
}

}

/\*All threads join master thread and terminate\*/

}





**Q.3.Write a Program to print name and registration number.**

#include<omp.h>

#include <stdio.h>

#include <stdlib.h>

int main()

{

int nthreads,tid;

/\*Fork a team of threads woth each thread having a private tid variable\*/

#pragma omp parallel private(tid)

{

/\*Obtain and print thread id\*/

tid=omp\_get\_thread\_num();

printf("Aryaman Mishra\n");

/\*Only master thread does this\*/

if(tid==0)

{

nthreads=omp\_get\_num\_threads();

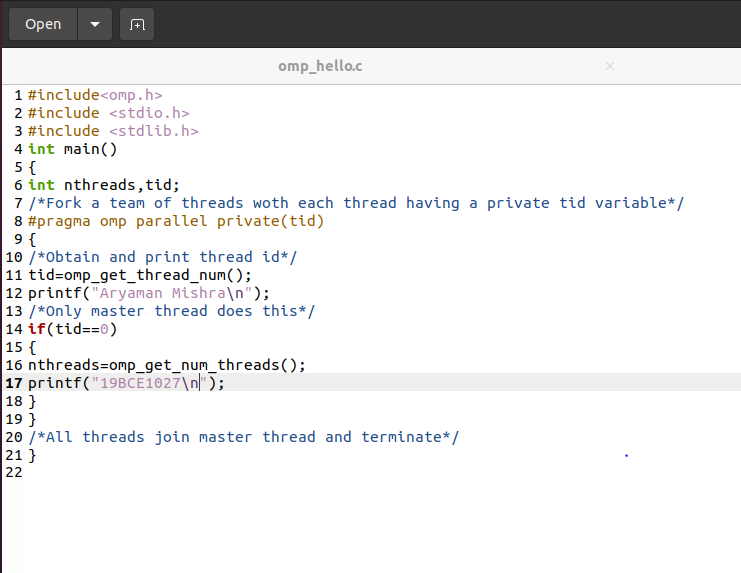
printf("19BCE1027\n");

}

}

/\*All threads join master thread and terminate\*/

}





**RESULT:ALL 3 PROGRAMS HAVE BEEN SUCCESFULLY EXECUTED.**