**12.2 OpenMP: (C library for Threading on multicore) shared memory programming.**

**Objectives:**

1. To learn about openMP for better use multicore system.

2. Implement the program OpenMP threads and print prime number task, odd number and Fibonacci series using three thread on core. Comment on performance CPU.

**Theory:**

OpenMP Is:

An Application Program Interface (API) that may be used to explicitly direct multi-threaded, shared memory parallelism.

Comprised of three primary API components:

Compiler Directives

Runtime Library Routines

Environment Variables

An abbreviation for: Open Multi-Processing

Shared Memory Model:

OpenMP is designed for multi-processor/core, shared memory machines. The underlying architecture can be shared memory UMA or NUMA.

**Program:**

#include<stdio.h>

#include<stdlib.h>

#include<omp.h>

Int main()

{

int i, n, t1 = 0, t2 = 1, nextTerm=0,j=0,counter,i,a,count;

#pragma openmp parallel private(i,counter,a,count)

In ID = omp\_get\_thread\_num();

if(ID==1)

{

for (i = 1; i <= n; ++i)

{

printf("%dth Fib no: =%d \n", nextTerm);

nextTerm = t1 + t2;

t1 = t2;

t2 = nextTerm;

}

}

if(ID==2)

{

for(counter = 1; counter <= 100; counter++) {

if(counter%2 == 1) {

printf("%dth Odd no: =%d \n", j,counter);

}

}

if(ID==3)

{

for (i=100;i<500;i++)

{

count=0;

for (a=1;a<=i;a++)

{

if (i%a==0)

count++;

}

if (count==2)

printf("%dth Prime no: =%d \n", j,i);

j++;

}

}

}

**Output:**

1th Fib no: = 0

2th Fib no: = 1

3th Fib no: = 1

4th Fib no: = 2

5th Fib no: = 3

1th Prime no: = 2

6th Fib no: = 5

1th Odd no: = 1

2th Odd no: = 3

2th Prime no: = 3

7th Fib no: = 8

3th Odd no: = 5

3th Prime no: = 5

8th Fib no: = 13

4th Prime no: = 7

9th Fib no: = 21

4th Odd no: = 7

5th Prime no: = 11

10th Fib no: = 34

6th Prime no: = 13

**Conclusion:**

1. If compared with serial code, OpenMP works faster as work is performed by multiple threads simultaneously.

2. CPU is being more utilized.

**References:**

[1] http://mpitutorial.com/tutorials/mpi-introduction/