**DEPARTMENT OF INFORMATION TECHNOLOGY, NITK SURATHKAL**

**Parallel Programming**

**LAB 1 – 27th July 2021**

**Note: Observe the results of each program, take the screenshot of the result and upload it in the Moodle. Total Marks [10]**

**Note:**

**parallel**

Forms a team of threads and starts parallel execution.

**#pragma omp parallel** *[clause[ [, ]clause] ...]*

*structured-block*

*clause*:

**if**(*scalar-expression*)

**num\_threads**(*integer-expression*)

**default**(**shared** | **none**)

**private**(*list*)

**firstprivate**(*list*)

**shared**(*list*)

**copyin**(*list*)

**reduction**(*reduction-identifier***:** *list*)

I. Finding number of CPU s in system **[3 Marks]**

a) lscpu command

$ lscpu  
$ lscpu | egrep 'Model name|Socket|Thread|NUMA|CPU\(s\)'  
$ lscpu -p

**b) Run top or htop command to obtain the number of CPUs/cores in linux**

$top

**c) Execute nproc and print the number of CPUs available on Linux**

$ nproc --all  
 $ echo "Threads/core: $(nproc --all)"

**2.Write a C/C++ simple parallel program to display the *thread\_id* and total number of threads. [3 Marks]**

/\*simpleomp.c\*/

#include<omp.h>

int main(){

int nthreads,tid;

**#pragma omp parallel private(tid)**

{

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

printf(“Hello world from thread=%d\n”,tid);

if(tid==0)

{

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

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

}

}

}

**Execute the program as follows:**

$gcc –o simple –fopenmp simpleomp.c

$export OMP\_NUM\_THREADS=2

$./simple

**Note down the output in your observation book.**

Number of threads in a parallel region is determined by the *if* clause, *num\_threads(), omp\_set\_num\_threads(), OMP\_NUM\_THREADS.*

**Use these various methods to set number of threads and mention the method of setting the same.**

**2. Check the output of following program and Note down the output in your observation book. [2 Marks]**

/\*ifparallel.c\*/

#include<omp.h>

int main(){

int val;

printf(“Enter 0: for serial 1: for parallel\n”);

scanf(“%d”,&val);

#pragma omp parallel if(val)

{

**if(omp\_in\_parallel())**

printf(“Parallel val=%d id= %d\n”,val, omp\_get\_thread\_num());

else

printf(“Serial val=%d id= %d\n”,val, omp\_get\_thread\_num());

}

}

**3.Observe and record the output of following program [2 Marks]**

/\*num\_threads.c\*/

#include<omp.h>

int main(){

#pragma omp parallel num\_threads(4)

{

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

printf(“Hello world from thread=%d\n”,tid);

}

}

**Change the num\_threads and observe the result.**