Parallel Programming

LAB 4

27th August 2018

Note: Write all programs in your observation book and record the results. Get the signature of faculty /teaching assistance.

1. **Execute following code and observe the tasks created and executed by the threads. Also draw the recursion tree and note down the nodes executed by each thread.**

```
#include<stdio.h>
#include<omp.h>
int fibo(int n);
void main()
{
int n,fib;
double t1,t2;
printf("Enter the value of n:\n");
scanf("%d",&n);
t1=omp_get_wtime();
#pragma omp parallel shared(n)
        {
        #pragma omp single
                {
                fib=fibo(n);
                }
        }
t2=omp_get_wtime();
printf("Fib is %d\n",fib);
printf("Time taken is %f s \n",t2-t1);
}

int fibo(int n)
{
int a,b;
if(n<2)
return n;
else
{
#pragma omp task shared(a)
        {
        printf("Task Created by Thread %d\n",omp_get_thread_num());
        a=fibo(n-1);
        printf("Task Executed by Thread %d \ta=%d\n",omp_get_thread_num(),a);
        }
#pragma omp task shared(b)
```

```
        {
        printf("Task Created by Thread %d\n",omp_get_thread_num());
        b=fibo(n-2);
        printf("Task Executed by Thread %d \tb=%d\n",omp_get_thread_num(),b);
}
#pragma omp taskwait
return a+b;
}
}
```

2. **Execute following code and observe the working of threadprivate directive and copyin clause:**

```
#include<stdio.h>
#include<omp.h>
int tid,x;
#pragma omp threadprivate(x,tid)
void main()
{
x=10;
#pragma omp parallel num_threads(4) copyin(x)
        {
        tid=omp_get_thread_num();
                #pragma omp master
                {
                printf("Parallel Region 1 \n");
                x=x+1;
                }
                #pragma omp barrier
        if(tid==1)
        x=x+2;
        printf("Thread % d Value of x is %d\n",tid,x);
        }//#pragma omp barrier
#pragma omp parallel num_threads(4)
        {
                #pragma omp master
                {
                printf("Parallel Region 2 \n");
                }
        #pragma omp barrier
        printf("Thread %d Value of x is %d\n",tid,x);
}
printf("Value of x in Main Region is %d\n",x);
}
```

DO the following:

1. Remove copyin clause and check the output.

2. Remove copyin clause and initialize x globally.
Note the observation about threadprivate directive and copyin clause.

3. **Write a C/C++ OpenMP program to find ROWSUM and COLUMNSUM of a matrix a[n][n].**
4. **Write a C/C++ OpenMP program to perform matrix multiplication.**
5. **The details of an employ with employ id and employ salary is stored in a two dimensional array. The company would like to raise the salary of all its employees by 6%. If the increase in salary is more than 5,000 Rs, the company would like to put tax of 2% on the increased amount more than 5,000 Rs. Calculate the total extra amount the company need spend by increasing the salary 6%.**

Note: Write the program, results and analysis in your observation book and get signature from TAs.