Laboratory for Software Analytics and Pervasive Parallelism
Prof. Ali Jannesari

jannesar@iastate.edu

Concurrent Systems (Coms 527, Spring 23)

Exercise 03

Please submit your solution by Feb 26, 2023 11:59 pm to Canvas. **Solutions which are delivered late, do not earn points.**

Task 1

(4 Points) Answer the following questions:

a) How do threads differ from processes?

b) What does thread-safe mean?

Task 2

(3 Points) What are the advantages and disadvantages of OpenMP?

Task 3

(5 points) Illustrate the fork-join execution model through a figure and explain how this model works.

Task 4

(10 Points)
Given is the following program:

Listing 1: Data sharing semantics

```
1  int count = 0;
2  double val = 0.0;
3
4  double* foo( int c, double* g, double h )
5  {
6    if ( h < c ) return g;
7    else return &val;
8  }
9
10 void bar(double* a, double *z, int n)
11 {
12    static int cnt = 0;
13    int i, j;
14
15    #pragma omp parallel for firstprivate (cnt, val)
16    for( j=0; j<n; j++)
17    {
18        int* res = foo( count, &a[j], z[j] );
19        for (i=0; i<j; i++)
20        {
21            cnt++;
22        }
23        count = count + *res + cnt;
24    }
25 }
```

a) **(2 Points)** Please determine whether the following variables are private or shared in foo:

- count
- val
- g
- *g

b) **(4 Points)** Please determine whether the following variables are private or shared in bar:

- count
- val
- cnt
- res
- *res
- i
- j
- a

c) **(2 Point)** Which variable(s) have a loop carried dependence?

d) **(2 Point)** What type of dependence does appear?

Task 5

(6 Points)

a) **(3 Points)** Assume all variables and arrays used in following loop are defined before this loop. Insert the missing OpenMP directives to parallelize this loop.

Listing 2: Example loop

```
1 for (i=0; i<N; i++)
2 {
3     y = sqrt(A[i]);
4     D[i] = y + A[i] / (x*x);
5 }
6
7 y = y+2;
```

b) **(3 Points)** What is the difference between static scheduling and dynamic scheduling in OpenMP? What are the advantages and disadvantages of dynamic scheduling compared to static scheduling?

Task 6

(5 points)

a) **(3 Points)** Briefly explain the three components (directives, runtime library, and environment variables) of the OpenMP API.

b) **(1 Point)** How can a programmer enable conditional compilation with a preprocessor macro for OpenMP?

c) **(1 Point)** If an OpenMP program is compiled with a compiler which doesn't support OpenMP, what will happen?

Task 7

(7 points) The OpenMP specification is freely available on the internet. Obtain the latest OpenMP specification.

a) **(1 Point)** What is an ICV in the OpenMP specification?

b) **(1 Point)** Lookup `omp_get_num_threads`. Is it a directive, a library function or an environment variable? What is its C-syntax? What does it do?

c) **(1 Point)** Lookup `omp_get_thread_num`. Is it a directive, a library function or an environment variable? What is its C-syntax? What does it do?

d) **(1 Point)** Lookup the `barrier construct`. Is it a directive, a library function or an environment variable? What is its C-syntax? What does it do?

e) **(1 Point)** What is an `inactive parallel region`?

f) **(1 Point)** Lookup `OMP_NUM_THREADS`. Is it a directive, a library function or an environment variable? What does it do?

g) **(1 Point)** Lookup `OMP_THREAD_LIMIT`. Is it a directive, a library function or an environment variable? What does it do?

Task 8

(10 points) Write a program to be implemented in OpenMP to add 2 arrays each of size 1000. Fill the arrays with random values. Also calculate the time taken by your parallel implementation. Compare the time taken with the sequential implementation and record the speedup. Please screenshot the time taken for both implementations.