Parallel Computing
Test 2
11:00 pm - 12:15 pm, Thursday November 15th, 2012
With some solutions.

Name: .....

This is a closed book test. Do not refer to any materials except those supplied for the test.

Supplied: *"Summary of OpenMP 3.0 C/C++ Syntax."*

Answer questions in space provided below questions. Use additional paper if necessary but make sure your name is on additional sheets.

Total /40

Qu. 1 Answer each of the following briefly:

(a) Explain the term *thread-safe routine*. 2

(b) Explain the term *sequential consistency*. 2

(c) Explain the term *cost-optimal*. 2

(d) How might a compiler, or the processor, reorder the following code prior to execution and why might this be done. There may be multiple reasonable answers but no credit will be given unless you clear explain. 2

```
x = y – z;
x++;
a = b * c;
x = a + z;
```

(e) Use Bernstein's conditions to determine whether the two code sequences: 4

```
forall (i = 0; i < 2; i++)
   a[i] += a[i+2];

for (i = 0;i < 2; i++)
   a[i] += a[i+2];
```

always produce the same results. Clearly show how you got your answer. (No credit given for just answering "yes" or "no"!)

(f) Explain the term false sharing in caches. Suppose three integer variables, x, y, and z, are declared in a C program. Under what circumstances could false sharing occur? Explain clearly. 2

(g) In Assignment 4, you had to compute the force applied to a particular body (a) by another body (b). The formula for the x dimension was $F_x = \frac{Gm_a m_b}{r^2}(\frac{x_b - x_a}{r})$. What is *r* in this equation? How is it calculated (what is the equation for *r*)? 4

(h) What do the OpenMP *section* pragma and *sections* pragma do? How is the code executed? What happens if there is no "nowait" at the end of a for pragma in OpenMP? 2

(i) Suppose you have P processors to use on parallel block matrix multiplication where the arrays are N x N and N = 4P. Describe how you would achieve parallel matrix multiplication, giving full details of what each processor does. 2

Using actual block multiplication with submatrices;

Need one processor for each submatrix in result array. So with n x n matrix and p = n/4, and s x s submatrices, $p = n^2/s^2$ which leads to s = 2 x srqt(n) Only works for certain powers of 2. Example n = 64, s = 16, p = 16

If just divide into rows:

Divide up the matrix in rows of size N/4.

(j) Why is the Gauss-Seidel relaxation iteration unsuitable for parallelization without modification? 2

(k) What is meant by hybrid parallel programming? 2

(l) Demonstrate how to sort the sequence: 4

4 5 2 8

using Batchers' Bitonic Mergesort algorithm. Clearly show the positions of each number after each step.

Supplied: "*Summary of OpenMP 3.0 C/C++ Syntax.*"

Qu. 2 Write an OpenMP program that transposes an N x N matrix using N^2 threads. It is incorrect to nest OpenMP for pragmas. The final result needs to be in the original array. If you use a temporary holding array, then you will need to copy the results back to the original.

Provide comments in your code to help the grader! Briefly describe your method. ***If I do not understand the code, I will assume it is incorrect.*** 10

Transposition of a matrix means switching the rows are columns, or swapping values across the diagonal. For example:

If the original matrix is:

1	2	3
4	5	6
7	8	9

The transposed matrix is:

1	4	7
2	5	8
3	6	9

The answer:

```
#pragma omp for
  for (i = 0; i < N; i++)
      for (j = 0; j < N; j++)
          B[j][i] = A[i][j];
```

doesn't make use of N^2 processors. Only N will participate. The others will be idle.

The following makes use of N^2 threads:

```
#pragma omp parallel num_threads(N*N) private (tid, i, j)
{
      #pragma omp for
      for (k = 0; k < N*N; k++) {
          i = k / N;
          j = k % N;
          B[j][i] = A[i][j];
      }

      #pragma omp for
      for (k = 0; k < N*N; k++) {
          i = k / N;
          j = k % N;
          A[i][j] = B[i][j];
      }
}
```