2016.1 Human Media Multicore Computing
Midterm Exam (11am, 4/25)

supervisor signature	

StudentID# : (   ) , Name : (   )

※ You may answer either in English or Korean language unless instructed to answer in English.

1. (30 points) Fill out the blanks (a)~(r) with the most appropriate English words. x______ means the word starts with alphabet 'x'.

- CC-NUMA abbreviation stands for
  (a. C__________) (b. C__________) (c. N__________) (d. U__________) (e. M__________) (f. A__________)

- (g. ________________) uses the collection of computer resources from multiple locations to compute and solve a complex problem. (same as (g)) tends to be more loosely coupled, (h. h________________) and geographically dispersed compared to cluster comupters.

- (i. ____________________) is the observation that, over the history of computing hardware, the number of transistors in a dense integrated circuit has doubled approximately every two years.

- In parallel computing, granularity is the ratio of the amount of (j. ______________) to the amount of (k. ______________). (same as (j)) stages are typically separated from periods of (same as (k)) by (l. ________________) events.

- Fine-grained parallelism means individual tasks (work units) are relatively (m. __________) in terms of code size and execution time. In the fine grained parallelism, we may obtain high performance because of better (n.   ), but, at the same time, we may obtain low performance because of high (o. o________________) of (same as (k) and (same as (l)).

- (p. ______________________) is a type of flaw in an electronic or software system where the output is dependent on the sequence or timing of other uncontrollable events.

- Comparison between concurrency and parallelism: Concurrency means (q.   ) simultaneous processing. On the other hand, parallelism means (r.   ) simultaneous processing.

2. (1) Describe two important advantages of a parallel computer with shared memory (i.e. shared-memory multiprocessor) architecture.
   (i) ((your answer should contain <u>less than 6 English words</u>)   )
   (ii) ((your answer should contain <u>less than 6 English words</u>)   )

(2) Describe two important advantages of a parallel computer with distributed memory architecture.
   (i) ((your answer should contain <u>less than 6 English words</u>)   )
   (ii) ((your answer should contain <u>less than 6 English words</u>)   )

3. Herb Sutter wrote an article "The Free Lunch is Over: A Fundamental Turn Toward Concurrency in Software".
(1) What did he mean by "Free Lunch"? Explain with sufficient details.
(   )

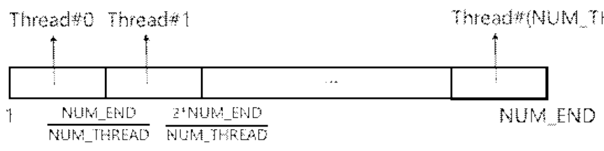
(2) Explain why the "Free Lunch" is over according to his article.
(   )

4. Amdahl's law can be formulated as following:

$$speedup = \frac{\text{execution time using one serial processor}}{\text{execution time using parallel processors}} \simeq \frac{1}{(1-a)+\frac{a}{b}}$$

(1) In above formulation, what is the variable 'a'? (   )
what is the variable 'b'? (   )
(2) The above formulation of Amdahl's law indicates WHAT? Explain. (   )

5.(40 points) Following multi-threaded JAVA code computes the number of prime numbers between 1 and NUM_END using NUM_THREAD threads. (Assume that NUM_END can be divided by NUM_THREAD) Following code uses static load balancing approach where we divide entire number range (1~NUM_END) into NUM_THREAD chunks : (1 ~ NUM_END/NUM_THREAD), (NUM_END/NUM_THREAD+1 ~ 2*NUM_END/NUM_THREAD), ..., ((NUM_THREAD-1)*NUM_END/NUM_THREAD+1 ~ NUM_END), and assign *i*-th chunk to *i*-th thread. Each thread calculates the number of prime numbers from its assigned chunk number range. The main thread finally collects the results from each thread and sum them up to print the final result.



(1) Implement above static load balancing approach by filling out empty boxes with appropriate JAVA code in the code below.

(2) Does above approach result in good load balancing or bad load balancing? Mark: (good load balancing / bad load balancing)
Explain why you think so : (____)

--< source code : ex4.java >--

```
class PrimeThread extends Thread {
  int min_val, max_val, counter;

  PrimeThread(int x,int y) {
    min_val = x;
    max_val = y;
    counter=0;
  }

  public int getCounter() {
    return counter;
  }

  private boolean isPrime(int x){
    int i;
    if (x<=1) return false;
    for (i=2;i<x;i++) {
      if ((x%i == 0) && (i!=x)) return false;
    }
    return true;
  }



}
```

```
public class ex4 {
  private static final int NUM_THREAD=4;
  private static final int NUM_END=200000;

  public static void main(String[] args) {
    int i,sum=0;
    int Width;
    PrimeThread[] t = new PrimeThread[NUM_THREAD];



    System.out.println("number of prime numbers: "+sum);
  }
}
```
