2014.1 Multicore Computing
Midterm Exam (April 22th 11am-12pm)

supervisor signature	

StudentID# : ( ) , Name : ( )

1. (32points) Fill out the blanks (a)~(p) with the most appropriate English words.

- Amdahl's law states that potential program (a.) is limited by the fraction of code that can be (b.).

- In (c.) system, processors have their own memory. Memory addresses in one processor do not map to another processor, so there is no concept of global address space across all processors..

- One of the first steps in designing a parallel program is to break the problem into discrete "chunks" of work that can be distributed to multiple tasks. This process is called as (d.).

- (e.) refers to the practice of distributing approximately equal amounts of work among tasks so that all tasks are kept (f.) all of the time.

- (g.) parallelism means relatively small amounts of computational work are done between communication events.
- (h.) parallelism means relatively large amounts of computational work are done between communication/synchronization events

- (i.) is Intel's technology that allows each physical processor core to act like (j.) to the operating system to improve the performance of multi-threaded program

- Threads belonging to the same process share
(k.), (l.), and (m.)

- (n.) are synchronization primitives that enable threads to wait until a particular condition occurs. Without (`same as the word in blank [n]`), the programmer would need to have threads continually (o.) to check if the condition is met.

- In JAVA language, we can control access to an object by using the (p.) keyword.
Using the (`same as the word in blank [p]`) keyword will force the lock on the object to be used.

2. (15 points) Fill out following blanks (a)~(e) with the most appropriate word(s).

Comparison between process and thread	
process	thread
Child process gets a (a.) of parent's variables.	Child thread (b.) parent's variables.
relatively (c.) to start..	relatively (d.) to start.
Don't have to worry about (e.) access to variables.	(`same as the word in blank [e]`) access to variables needs to be limited (controlled).

3. (16 points) Fill out following blanks (a)~(d) with your explanation.

	Comparison	
	Shared Memory Architecture	Distributed Memory Architecture
Advantage	(a.)	(b.)
Disadvantage	(c.)	(d.)

4.(37points) Following multi-threaded java code efficiently computes and displays the sum of each element in an array with size NUM_END (**int_arr**) that was initialized as {1,2,3,⋯,NUM_END}. Assume the number of threads the program creates is NUM_THREAD, and NUM_END is divisible by NUM_THREAD. Fill out empty boxes below with appropriate java codes..

```
---------------------------------------< source code : ex2.java >-------------------------------------------
class SumThread extends Thread {
  int lo; // fields for communicating inputs
  int hi;
  int[] arr;
  int ans = 0; // for communicating result
  SumThread(int[] a, int l, int h) {
    lo=l; hi=h; arr=a;
  }
  public void run() {



  }
}
class ex2 {

  private static final int NUM_END = 1000;  // assume NUM_END is divisible by NUM_THREAD
  private static final int NUM_THREAD = 4;

  public static void main(String[] args) {
    int[] int_arr = new int [NUM_END];
    int i,s;
    for (i=0;i<NUM_END;i++) int_arr[i]=i+1; // initialization of array : int_arr = {1,2,3,...,NUM_END}
    s=sum(int_arr);
    System.out.println("sum=" + s);
  }

  static int sum(int[] arr) {
    int len = arr.length;
    int ans = 0;
    SumThread[] ts = new SumThread[NUM_THREAD];
    for (int i=0;i<NUM_THREAD;i++) {




    }
    try {
      for (int i=0;i<NUM_THREAD;i++) {




      }
    } catch (InterruptedException IntExp) {}
    return ans;
  }
}
---------------------------------------------------------------------------------------------------------
```

Output result :

```
sum=500500
```