# CSCI-UA.0480-051: Parallel Computing

## Midterm Exam (Practice Exam)

### Total: 100 points

## Important Notes- READ BEFORE SOLVING THE EXAM

• If you perceive any ambiguity in any of the questions, state your assumptions clearly and solve the problem based on your assumptions. We will grade both your solutions and your assumptions.

• This exam is practice.

• You may use any resources you deem necessary to answer these questions. This is a practice exam; the goal is to familiarize yourself with the style and content that might appear on the actual exam.

## Problem 1

a. [8 points] Suppose we have a core with pipelining but lacks superscalar or hyperthreading capabilities. Explain whether branch prediction would still offer a performance benefit and justify your answer concisely.

b. [8 points] An eight-core processor is initially designed using only pipelining. Then it's upgraded to have four execution units per core and finally uses four-way hyperthreading with branch prediction. Detail the maximum number of processes and threads that could run concurrently in each scenario. Explain the reason for differences in the results. Fill in the table below:

If each one of the 8 cores is: | The maximum number of processes that can execute on the whole processor is: | The maximum number of threads that can execute on the whole processor is:

------- | -------- | --------

Only pipeline | |

Superscalar with four execution units per core | |

Four-way hyperthreading with branch prediction | |

## Problem 2

[15 points] Is the statement "Increasing shared variables among processes always increases the chance of false sharing and reduces performance due to coherence overhead" true or false? Provide a detailed explanation and justification.

## Problem 3

[12 points] Referring to the following task flow graph, identify all the spans present in the DAG (directed acyclic graph). Justify why there are multiple spans or why there can be only one span (as the case may be).

```
A
/ \
B   C
/ \ / \
D  E F  G
```

## Problem 4

a. [12 points] For the task flow graph above, determine the minimum number of cores needed to achieve maximum speedup. Show your calculations and specify the task assignment to cores to achieve this speedup, explaining your choices. Assume the following task times: A=10, B=5, C=5, D=10, E=5, F=10, G=10 (nanoseconds).

b. [10 points] Based on your core assignment in the previous question, is there load imbalance? Justify your answer using specific examples.

## Problem 5

[8 points] Calculate the speedup and efficiency for the optimal core assignment in Problem 4. Clearly show your steps and formulas.

## Problem 6

[8 points] In Problem 4, assuming sequential execution on a single core with a 4GHz clock and 100 machine language instructions per task, determine the Cycles Per Instruction (CPI). Show your work.

## Problem 7

[15 points] In the following MPI code snippet, if the `break;` statement in `case 1:` is removed, describe the consequences for each process (0, 1, and 2). Detail how the values of x, y, and z would change in each process and explain the reasons.

```c++
int x, y, z;
switch(my_rank) {
case 0:
x=9; y=8; z=7;
// ... MPI calls ...
break;
case 1:
x=6; y=5; z=4;
// ... MPI calls ...
//break; //Removed break statement
case 2:
x=3; y=2; z=1;
// ... MPI calls ...
break;
}
```

## Problem 8

[4 points] In the MPI code above (with the break; statement in case 1), is it possible for processes 0 and 1 to execute simultaneously on the same core? Justify your answer considering the nature of MPI and parallel execution.

"I understand the ground rules and agree to abide by them. I will not share answers or assist another student during this exam, nor will I seek assistance from another student or attempt to view their answers."