Homework 1

Mincheol Sung
ECE/CS 5510 Multiprocessor Programming

September 6, 2018

1.
(a) *Safety*: both processes are never in the critical section at a same time.
(b) *Liveness*: clients may retain resources eventually.
(c) *Safety*: a situation where it is not cloudy before it rains should never happen.
(d) *Safety*: more than one person never sits on the drivers seat.
(e) *Liveness*: the green light will eventually be turned on.
(f) *Safety*: the car never enters the roundabout when there is a car in the roundabout.
(g) *Liveness*: the car will leave the roundabout eventually.
(h) *Safety*: more than one direction/side is never green at a same time.

2.
Time for single instruction on the first machine:

$$\frac{1}{N * x}$$

Time for single instruction on the second machine:

$$\{(1-p) + \frac{p}{2N}\} * \frac{1}{x} \ (where\ p\ is\ partial\ of\ parallelizable)$$

So that the second system is advantageous,

$$\frac{\frac{1}{N*x}}{\{(1-p) + \frac{p}{2N}\} * \frac{1}{x}} > 1$$

$$\therefore P > \frac{2N-2}{2N-1}$$

3.
We can get two equations,

$$\frac{T_s}{T_p} = P + \frac{1-P}{N} \tag{1}$$

$$4 * \frac{T_s}{T_p} = \frac{P}{10} + \frac{1-\frac{P}{10}}{N} \tag{2}$$

$$where\ T_s\ is\ time\ of\ sequential\ part,$$

$$T_p\ is\ time\ of\ parallel\ part, and\ P\ is\ partial\ of\ sequential$$

$$\therefore P = \frac{5}{N-1}$$

The invariant is

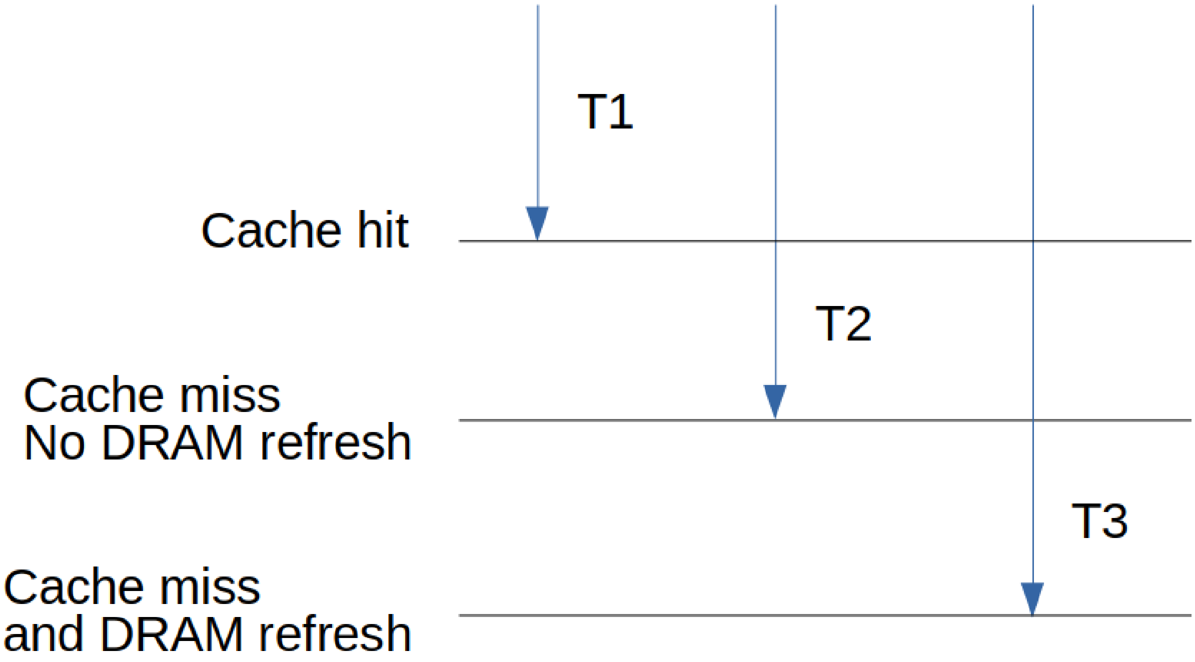
$$0 < P < 1$$

Therefore,

$$0 < \frac{5}{N-1} < 1$$

$$\therefore N > 6$$

4.



Let N is the number of accesses, T_1 is access time of cache hit, T_2 is access time of cache miss with no DRAM refresh, and T_3 is access time of cache miss with DRAM refresh.

Elapsed time is

$$E_1 = NT_1 + 0.35NT_2 + 0.15NT_3$$

If the penalty incurred by DRAM refresh is eliminated,
then

$$T_3 = T_2$$

So, elapsed time in those case is

$$\begin{aligned} E_2 &= NT_1 + 0.35NT_2 + 0.15NT_2 \\ &= NT_1 + 0.5NT_2 \end{aligned}$$

Performance benefit (speed up) is

$$\frac{E_2}{E_1} = \frac{NT_1 + 0.5NT_2}{NT_1 + 0.35NT_2 + 0.15NT_3}$$

5.
There are two cases.

Case 1. low-major language
There is dependency among columns, but no dependency among rows.
The inner loop can be parallelized because the inner loop determines row.
The outer loop can not be parallelized because the outer loop determines column.

Case 2. column-major language
There is dependency among rows, but no dependency among columns.
The inner loop can not be parallelized because the inner loop determines row.
The outer loop can be parallelized because the outer loop determines column.

6.
One student is assigned to a captain. The caption is the only one who can turn on/off the first table lamp. The first table lamp is for flagging that all students have once entered the common room. Only the caption can turn off the second table lamp. Others can only turn on the second table lamp. They can turn on the second table lamp only if it is off. In addition, once the students turn on the lamp, he will never turn on the lamp anymore.

A student enters the common room and checks the second lamp. If he sees the second table lamp is off and he has never turned on it, he would turn the lamp on. Otherwise, he does nothing and goes back to his room. In case of the captain enters the common room and sees the second lamp is on, he add one to his counter in his memory and turns the lamp down. He turns the first lamp on when his counter reaches to 11. If the the first lamp is on, students can state that all the students have once entered the common room.