(1) HW Week 7

1.

	Latency
32kB of Cache	10ns
512MB of DRAM	100ns

Processor 1 signal/step

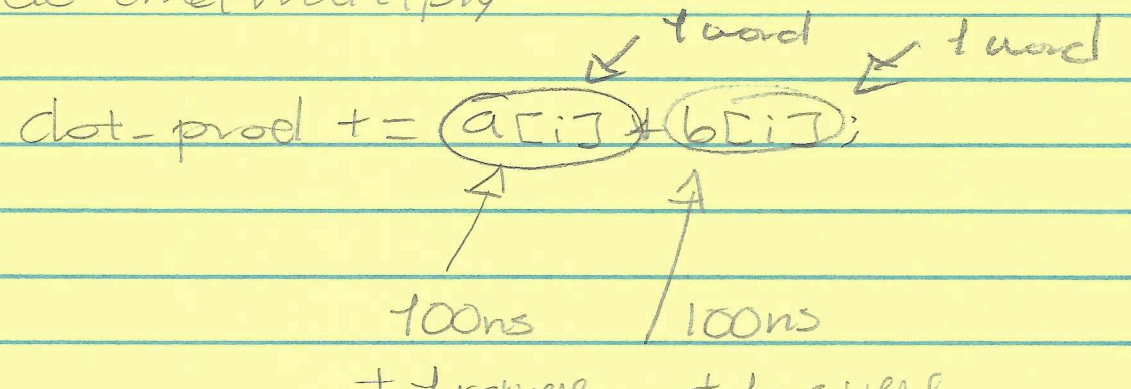
ns (nanosecond)

1 cycle => 4 words from cache

Question: How long an instruction takes to be done?

In the code we have two operations

add and multiply



* stored in cache although never retrieved

hence

$$\frac{1}{10\ ns} =$$

2. We assume conditions of problem 1

	Latency
512MB RAM	100ns
32KB Cache	1ns

Processor gives 1 signal per nanosecond

Data = a 4k*4k Matrix ≡ 16k entries

b 4k*1 Vector ≡ 4k entries

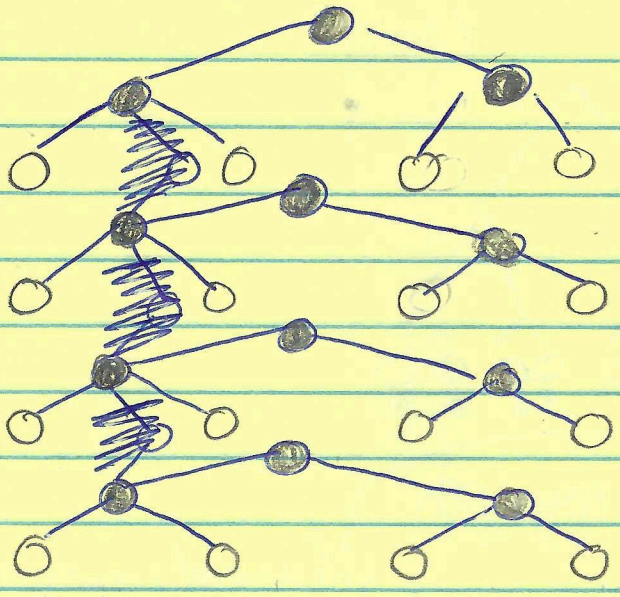
Every cycle the processor takes 4 words from the cache

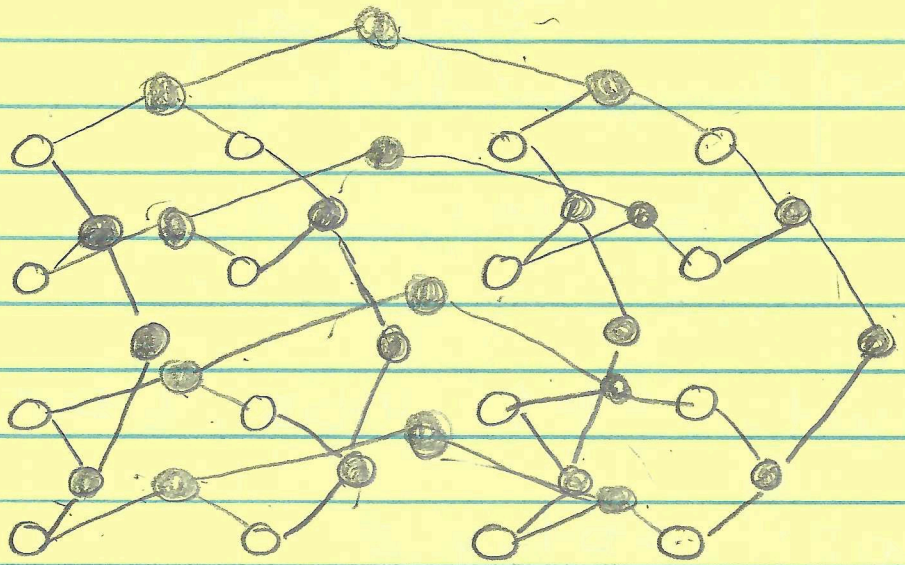
(1) Time to put b in Cache

$$\frac{4000 \text{ words} * 100\text{ns/word}}{16{,}000 \text{ accesses}} = \boxed{400{,}000 \text{ ns}} \Rightarrow 25\text{ns}$$

Each row takes 16KB of storage, when reads every word in the row takes 2 words = 32 bits

(3)



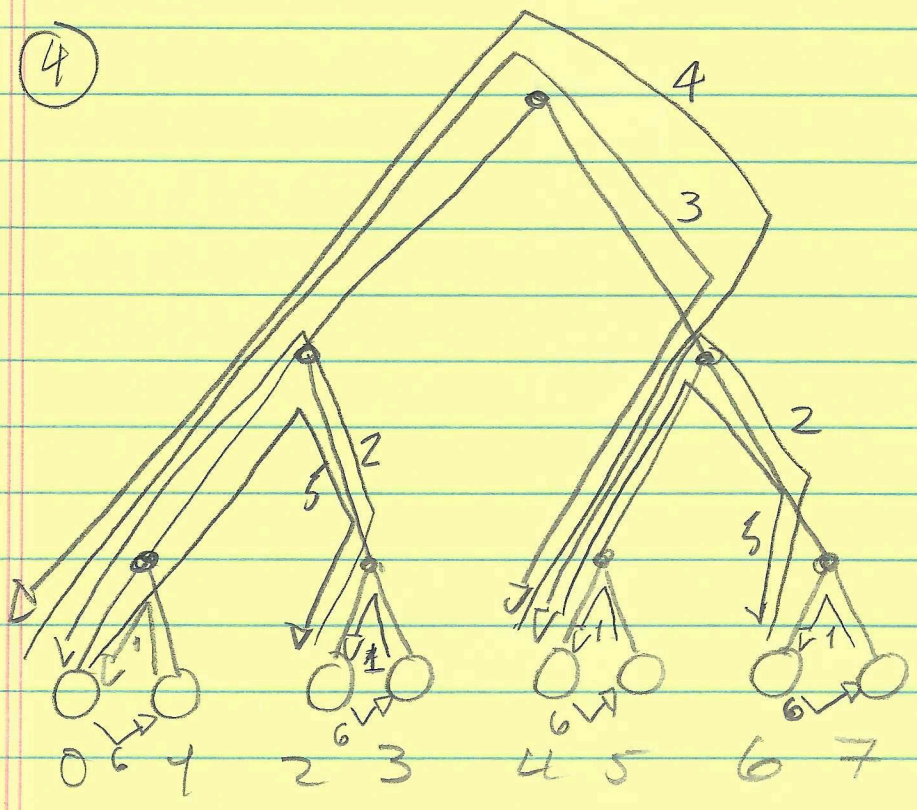


a. Bisection Width = 8

Diameter = 7

Switching nodes = 24

(4)

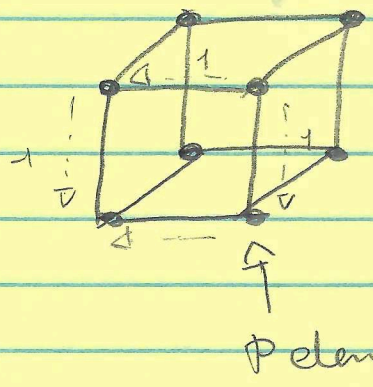


4.

Algorithm

- Transmit the data of the neighbor on the right to the neighbor on the left
- Then transmit the data on successive, lower depth nodes from right to left
- When all depths are covered from depth 0 to depth $\log(p)-1$ the distance from left to right at every depth on the subtrees

5.



* This problem is equal to the prefix sum operation problem but on p elements

Hence the time will be p * time
sum operation

So the algorithm is described as:

For every element in P
do
All-reduce on every element of "P"

Here the cost of the algorithm is

$P(t_s + t_w m) \log(P)$