Date:
Day: M T W T F S

Parallel & Distributed Computing

↳ Basic Communication Operators / Collective Communication Operators

↳ One to All Brodcast / All to one Reduction

Communication Operators

↳ Interconnection (Ring, Ring, Hypercube, linear array)

↓

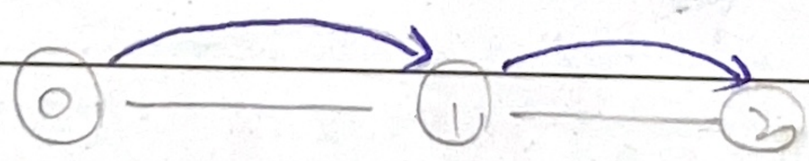
Communication cost

↓

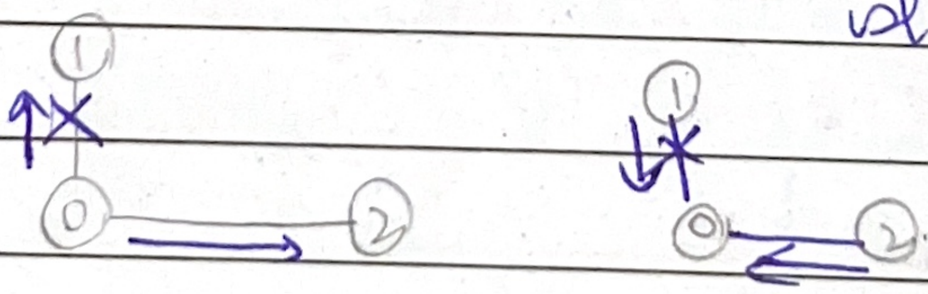
Algorithm Design

Assumptions of Operations:-

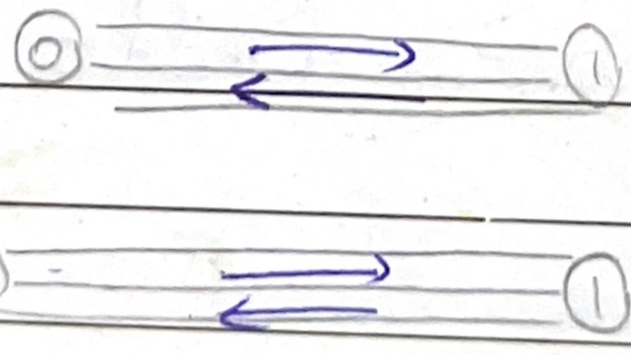
↳ Interconnection should support cut through routing (flits)

↳ 

travel time of msg should be same.

↳ 

Only one send or receive at a time multiple not allowed.

↳ 

communication is bi-directional

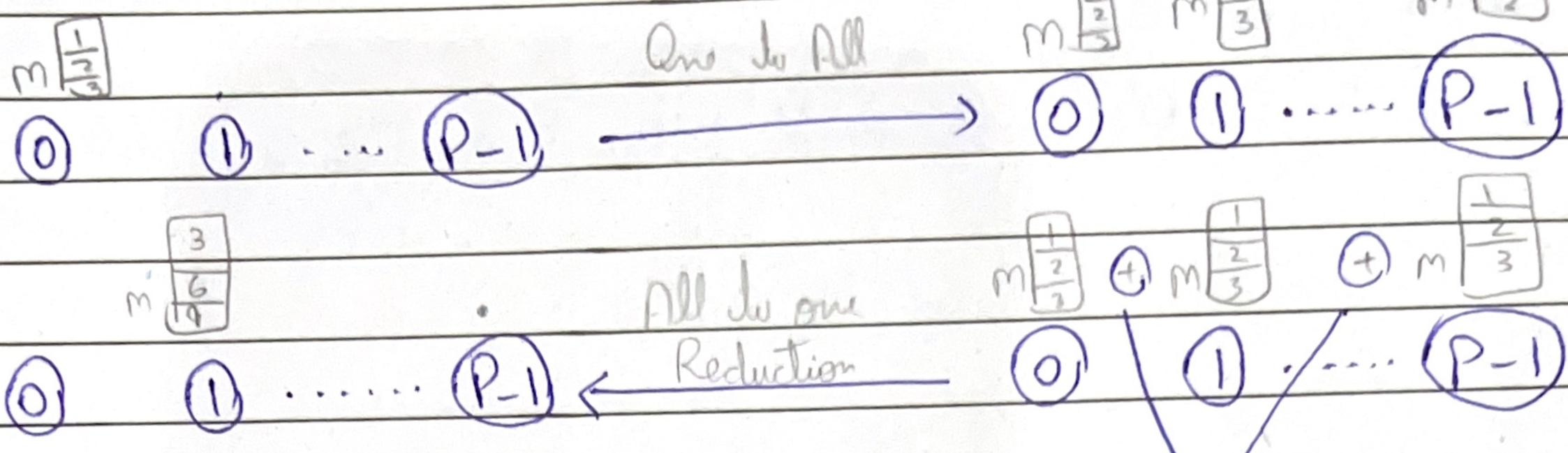
↳

while sending from ① — ① receiving on the same from ① — ① can also be used.

Date:

Day: M T W T F S

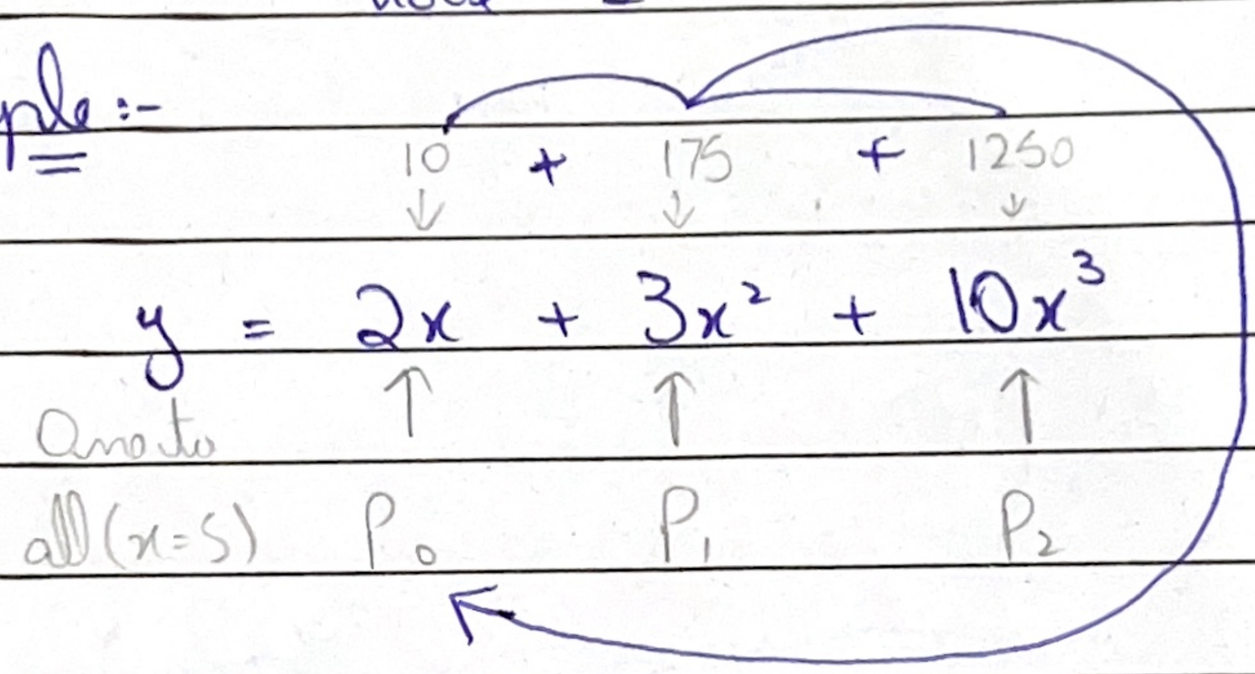
- One - to - All Broadcast :-



* Reducing values from right to the left node 1

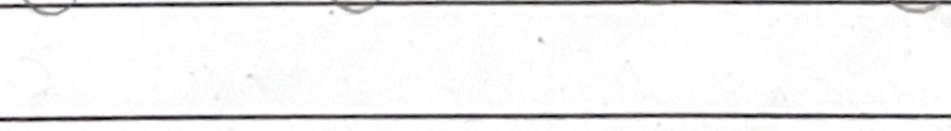
Can be any { XOR, NOR, XNOR, OR, AND }

Another example :-



User did one to all broadcast on processors with $x = 5$
After computation the result of all processors will reduce at P_0 at stored in y.

Recursive Doubling : Send message to another P_i and then in S_2 both P's can send it further reducing the cost This is called as recursive doubling.



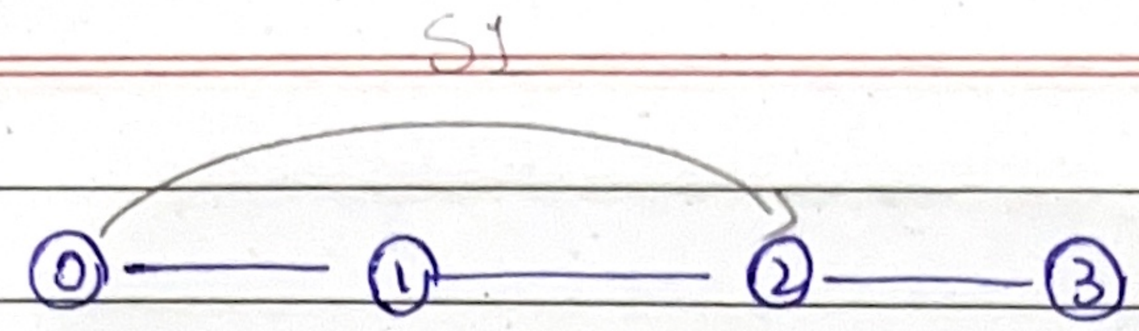
Time for $0 \to 1 = t_s + m t_w$

" $1 \to 2$ = "

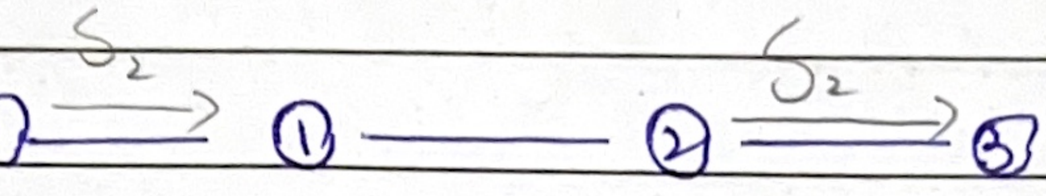
" $2 \to 3$ = "

Total = $(P-1)(t_s + m t_w)$

Date:
Day: M T W T F S

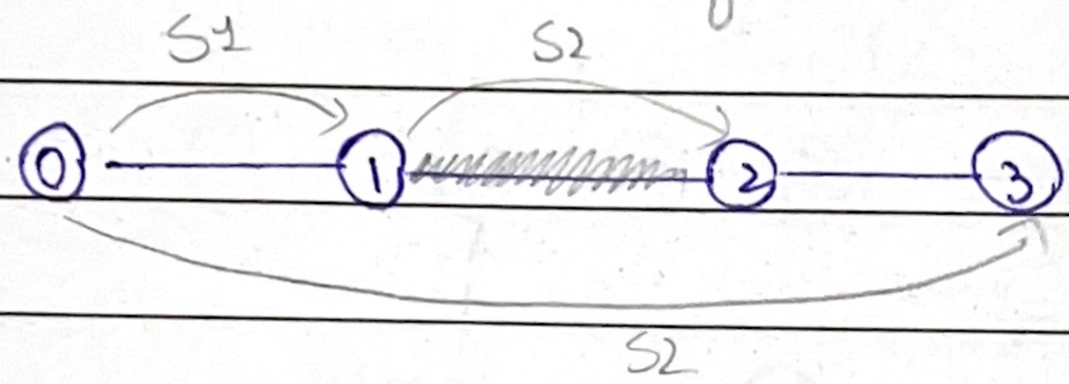


Phase 1 $0 \rightarrow P/2 = 2$



Phase 2

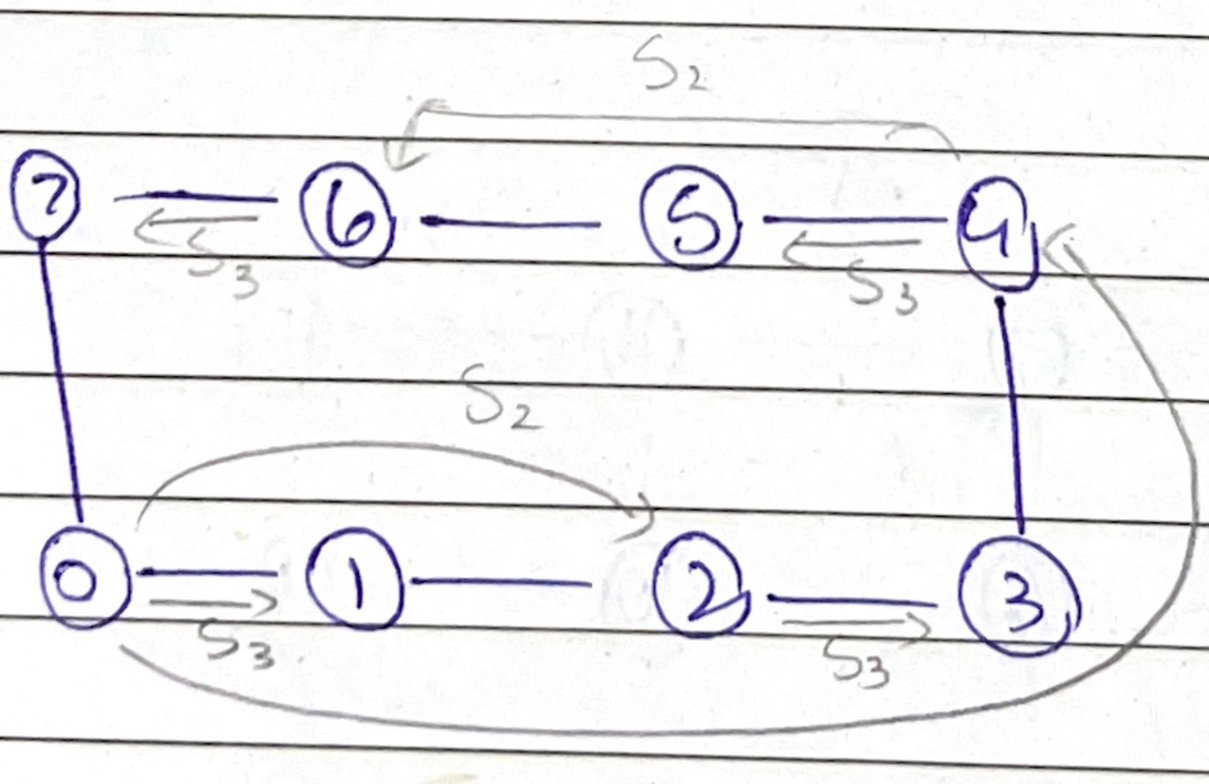
No congestion at ① → ②, it will occur if we do it directly from ⓪ → ① → ② → ③



for ⓪ → ② ① → ② link is used
& for ⓪ → ③ still ① → ② link is used causing congestion.

Steps
$2^i = P$
$2^2 = 4$
i = Steps
Steps = 2

Ring :-



Steps
$2^i = P$
$2^2 = 8$
$i = 3$
Steps = 3

Phase 1 :-

$0 \rightarrow P/2 = 4$ (Half)

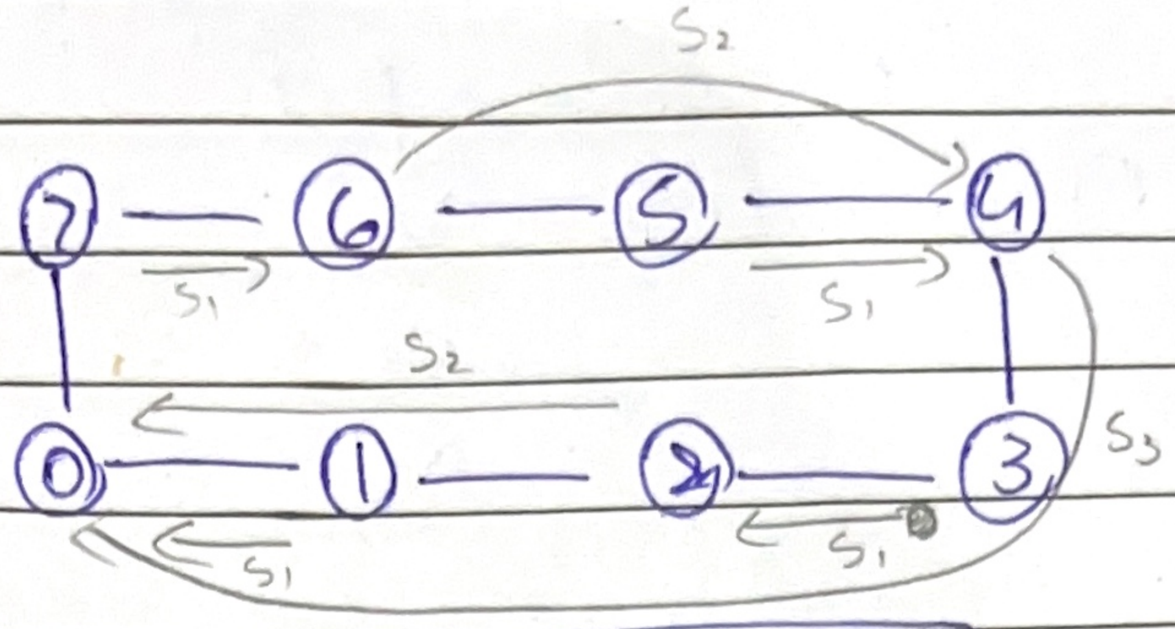
Phase 2 :-

$0 \rightarrow P/4 = 2$ (Quater) First
$4 \rightarrow P/2 + P/4 = 6$ (Quater) Last

Phase 3 :-

$0 \rightarrow 1$
$2 \rightarrow 3$
$4 \rightarrow 5$
$6 \rightarrow 7$

For All to one Reduction :-



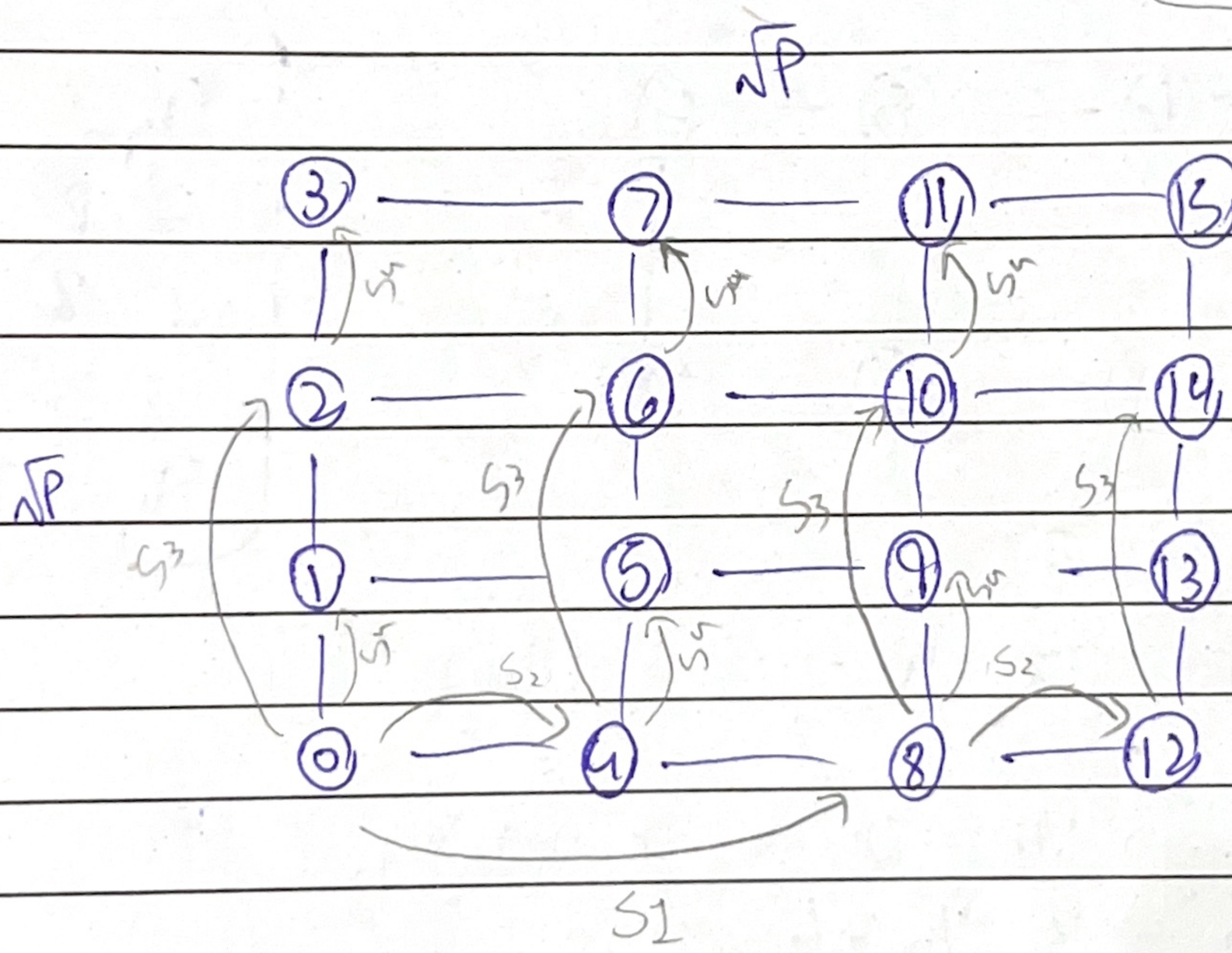
$$\boxed{\text{Cost} = \log P\ (t_s + m t_w)}$$

$\log P$ → Number of steps in which msg is passing to all Pi

Reverse the arrows and change step ordering $\frac{1}{2} = \frac{3}{2}$

$S_1 \rightarrow S_3$
$S_2 \rightarrow S_2$
$S_3 \rightarrow S_1$

Mesh :-



$P = 2^2$

$16 = 2^4$ → Steps

$$\text{Cost} = \log \sqrt{P}\ (t_s + m t_w) + \log \sqrt{P}\ (t_s + m t_w)$$

$$= 2 \log \sqrt{P}\ (t_s + m t_w)$$

$$= 2 \log P^{1/2}\ (t_s + m t_w) = 2 \times \tfrac{1}{2} \log P\ (t_s + m t_w)$$

$$\boxed{\text{Cost} = \log P\ (t_s + m t_w)}$$

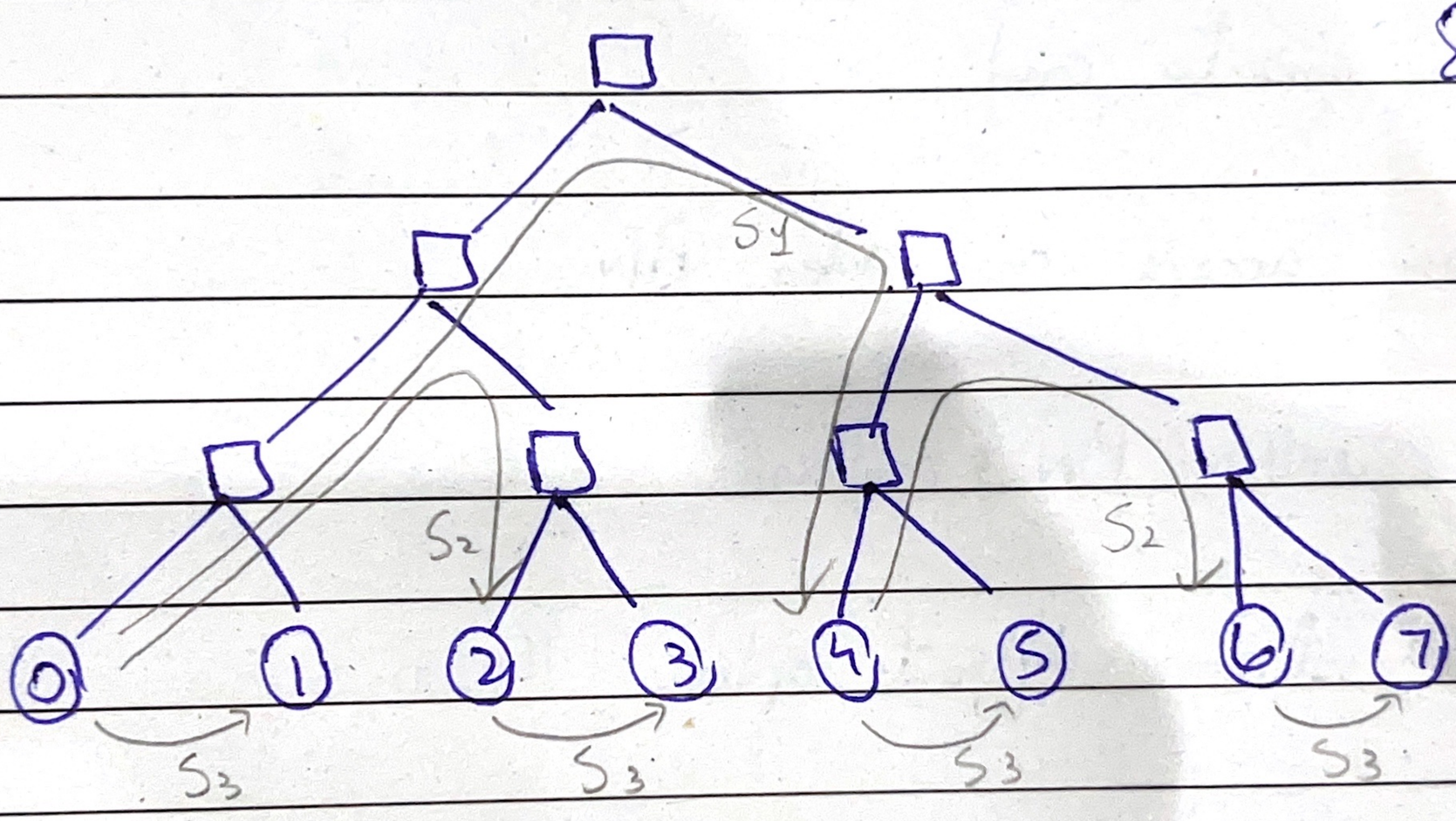
Date: _____

Day: M T W T F S

Tree :-

$P = 2^i$

$8 = 2^{(3)}$ → steps



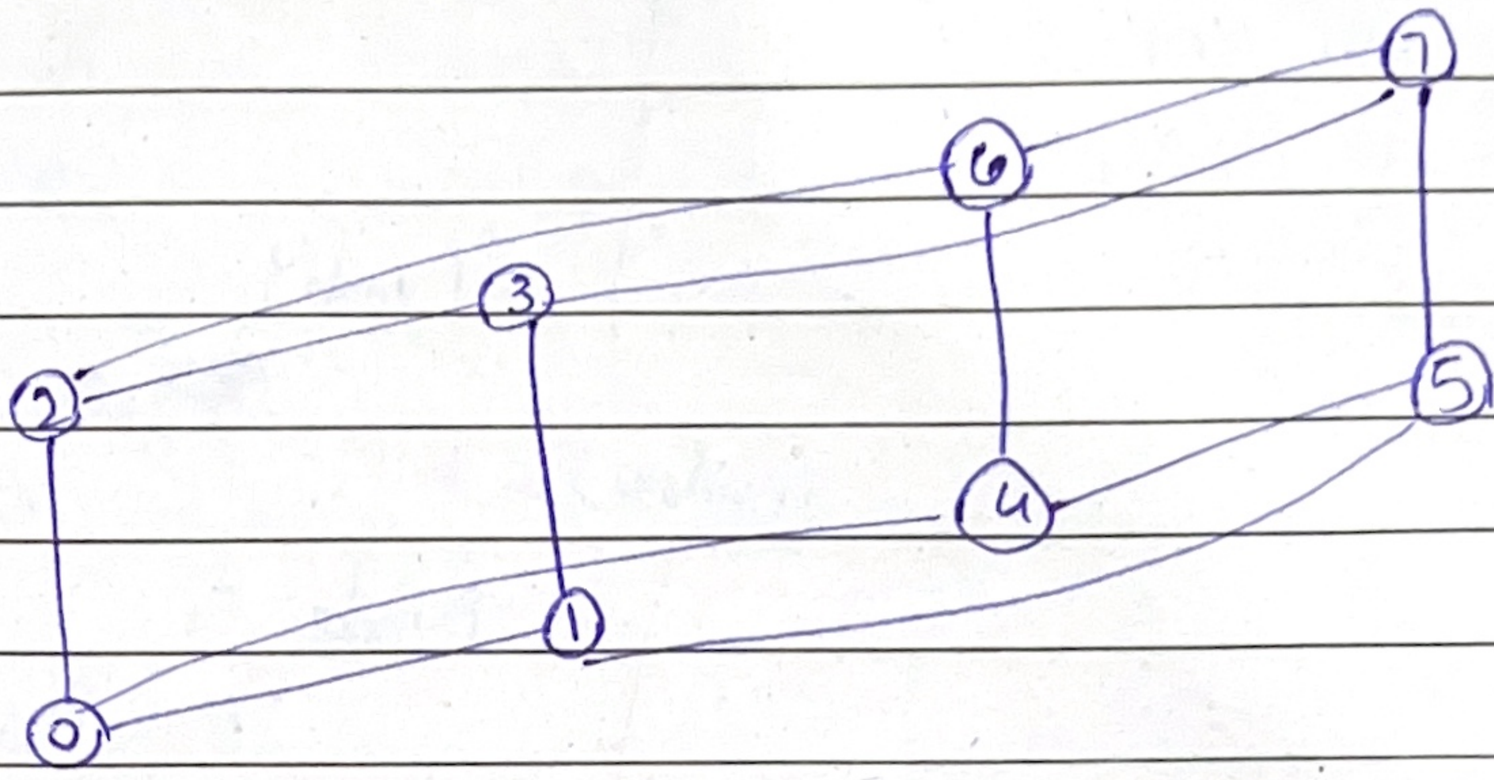
$$\text{Cost} = \log P (ts + mtw)$$

Date: _____

Day: M T W T F S

Basic Communication Operations :-

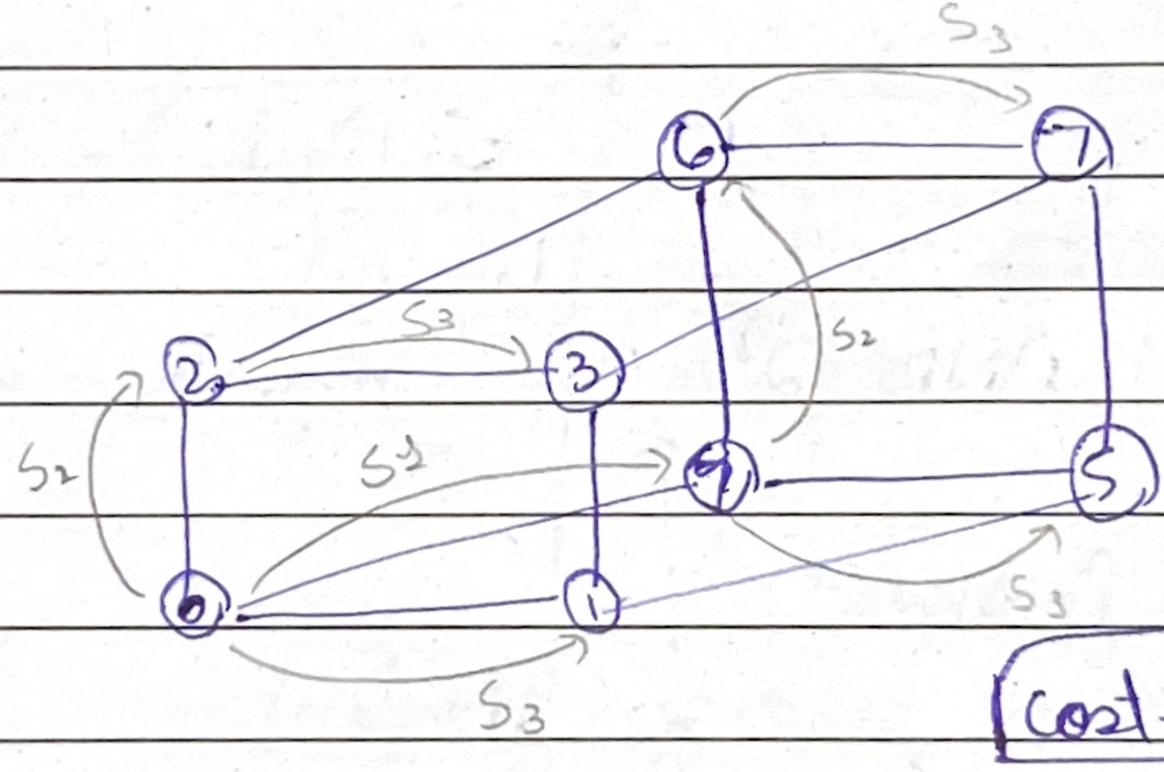
↳ One to All Brodcast / All to one Reduction



$2^i = 8$

$2^{(3)} = 8$

↳ Steps



$\boxed{\text{cost} = \log P(t_s + mt_w)}$

Algorithm 4.4

d = dimensions = 3

my-id = 0/000

X = message

1.) How many number of nodes participating in communication ?

2.) Who is sender/receiver?

3.) Communication partner ?

mask = $2^d - 1$

$= 2^3 - 1 \quad = 8 - 1 = 7$

$\boxed{\text{mask} = 111}$

Date:
Day: M T W T F S

6.) Participating nodes.

i=2

mask = mask XOR 2^i

= 111 XOR 2^2

= 111 XOR 4

= 111 XOR 100

mask = 011

if (my-id AND mask) = 0

Node 0

000 AND 011 ✓

000

Node 1

001 AND 011 X

001

Node 4

100 AND 011 ✓

000

Node 0 & Node 4

7.) Sender ~~of Packet~~ of msg

if (my-id AND 2^i) = 0

Node 0

000 AND 100 ✓

000

Node 4

100 AND 100 ✓

100

8.) Destination & Source

Node 0

msg-des = my-id XOR 2^i

= 000 XOR 100

= 100 (Nod 4)

Node 4

msg-src = my-id AND

~~100 AND 011~~

= 100 AND 011

= 000 (Nod 0)

Code for One to All broadcast

Code for general One to All any node can be root

Code for All to one reduction.

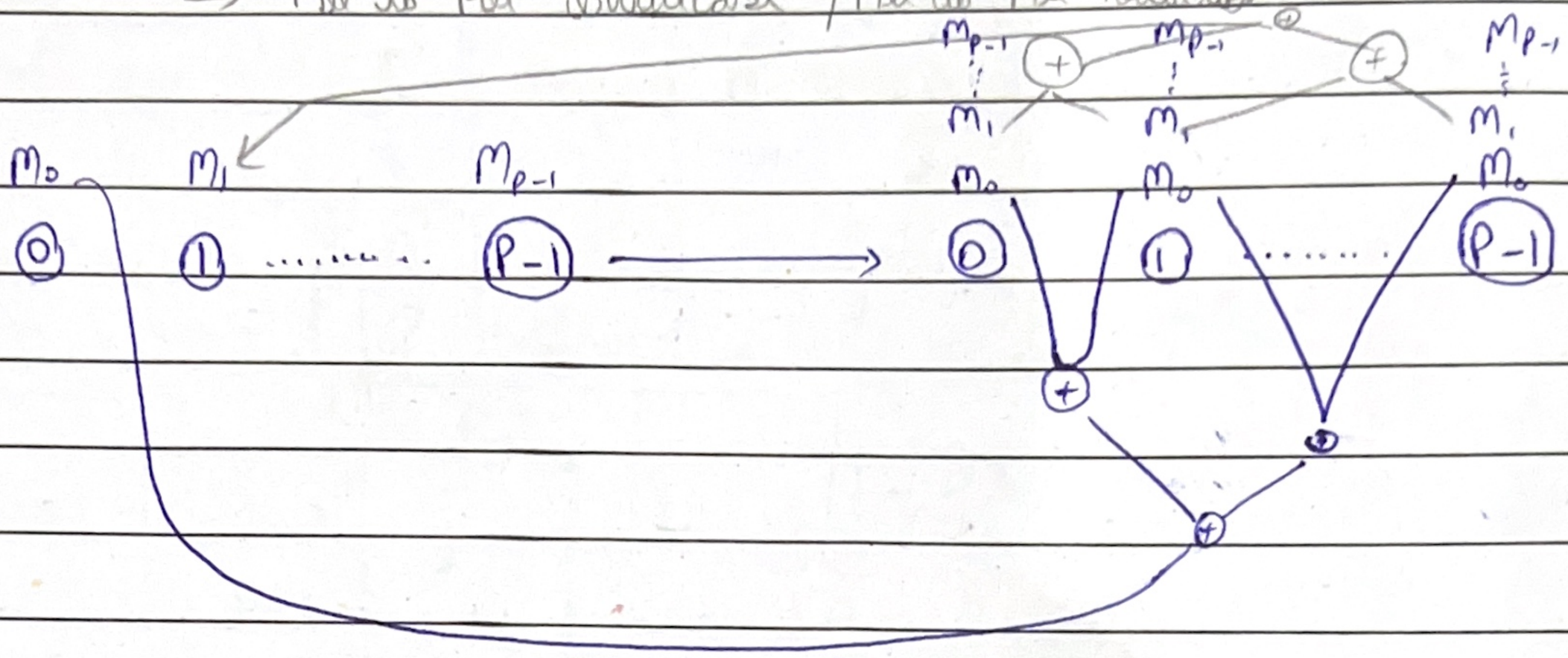
Date:

Day: M T W T F S

Parallel & Distributed Computing

↳ Basic Communication Operations

↳ All to All Broadcast / All to All reduction



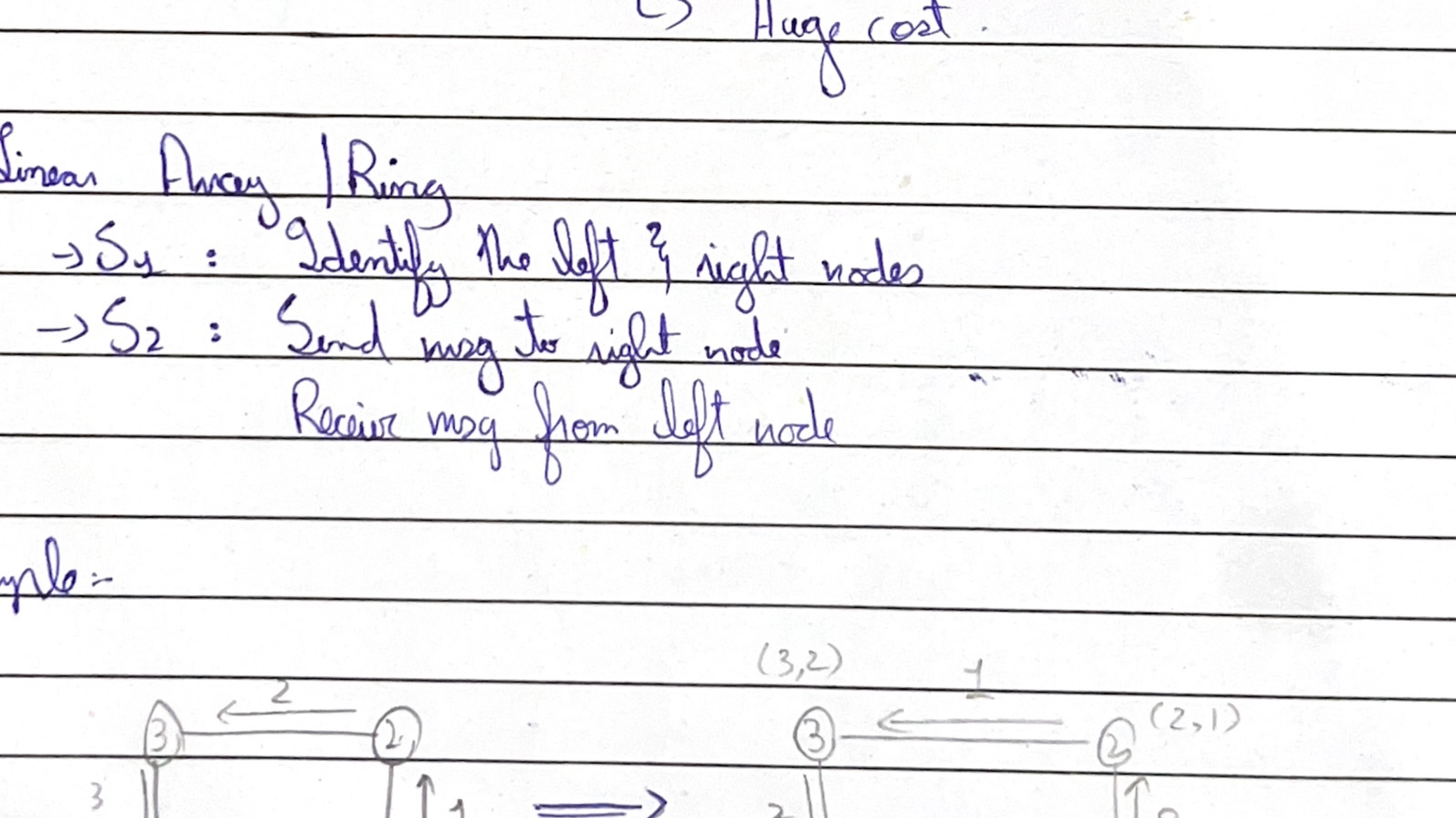
$P * (\text{One to all Broadcast})$

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

↳ Huge cost.

- Linear Array / Ring
  - → S_1 : Identify the left & right nodes
  - → S_2 : Send msg to right node
    Receive msg from left node

Example :-



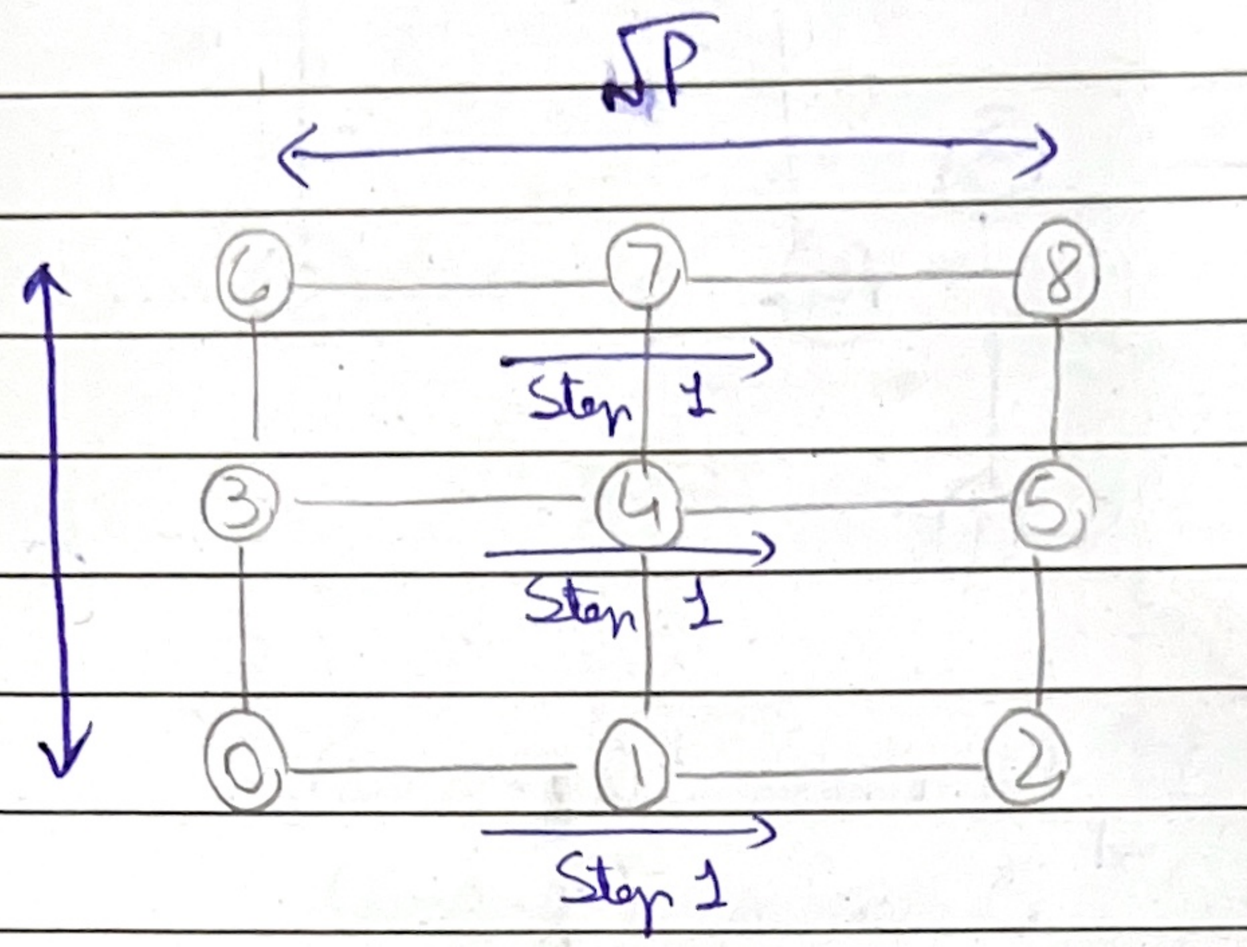
Date: _____

Day: M T W T F S

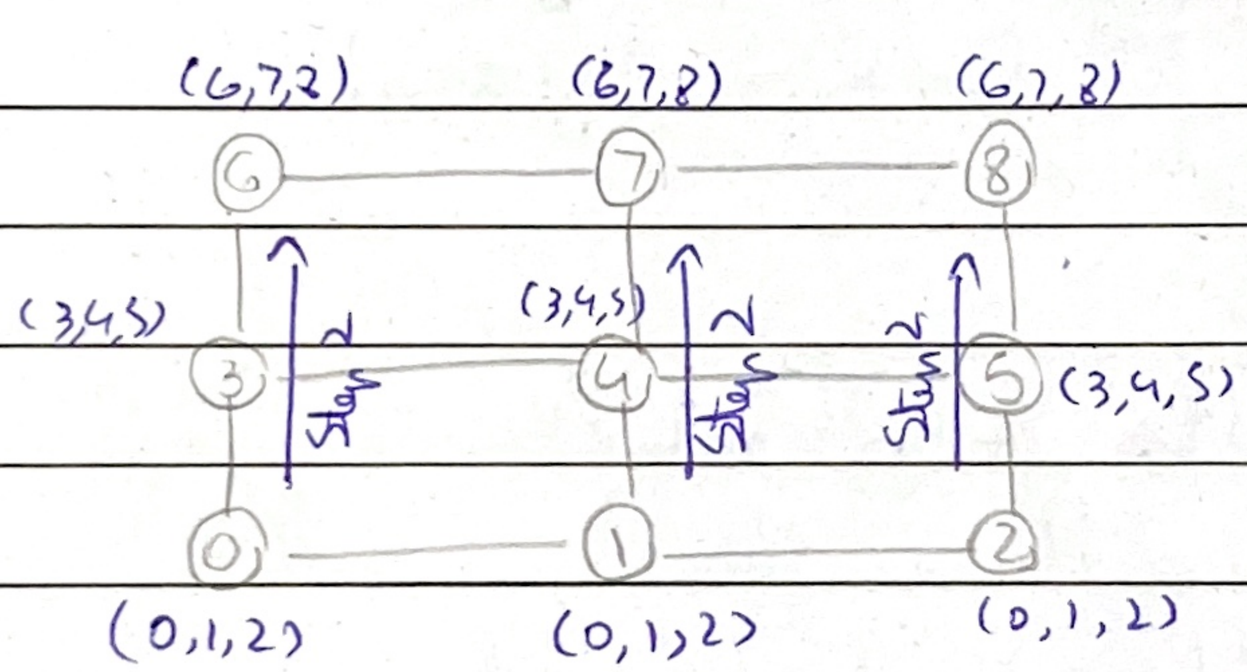
→ Number of Steps

Cost: $(P-1)(t_s + m t_w)$

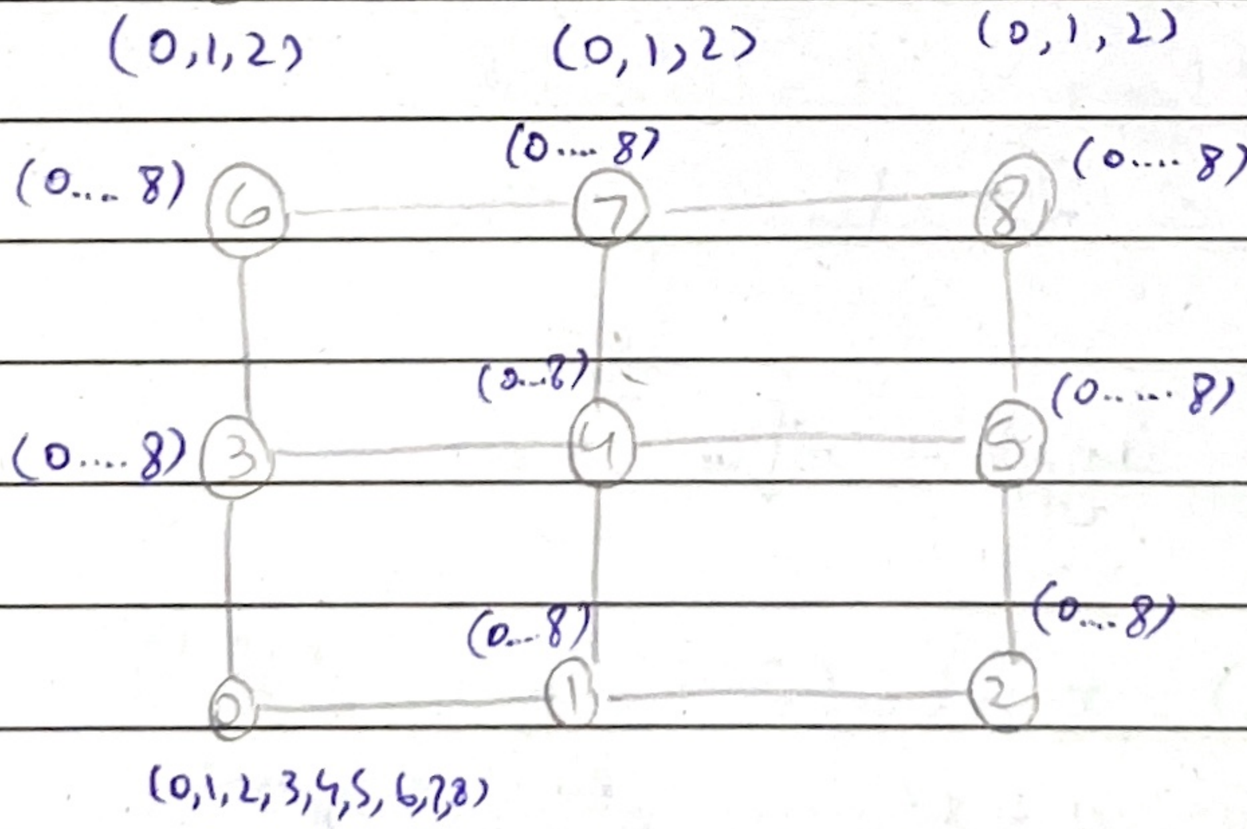
Mesh :-



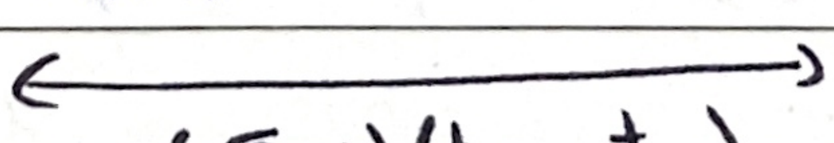
⇒ S_1 → Row wise all to all broadcast (0 — 1 — 2) apply on each row



⇒ S_2 → Column wise all to all broadcast



⇒ S_3 → The 0 will pass all its values (0,1,2) to the column & vice versa



Cost ⇒ $(\sqrt{P}-1)(t_s + m t_w) + (\sqrt{P}-1)(t_s + \sqrt{P} t_w m)$

$= t_s(\sqrt{P}-1) + m t_w(\sqrt{P}-1) + t_s(\sqrt{P}-1) + \sqrt{P} m t_w(\sqrt{P}-1)$

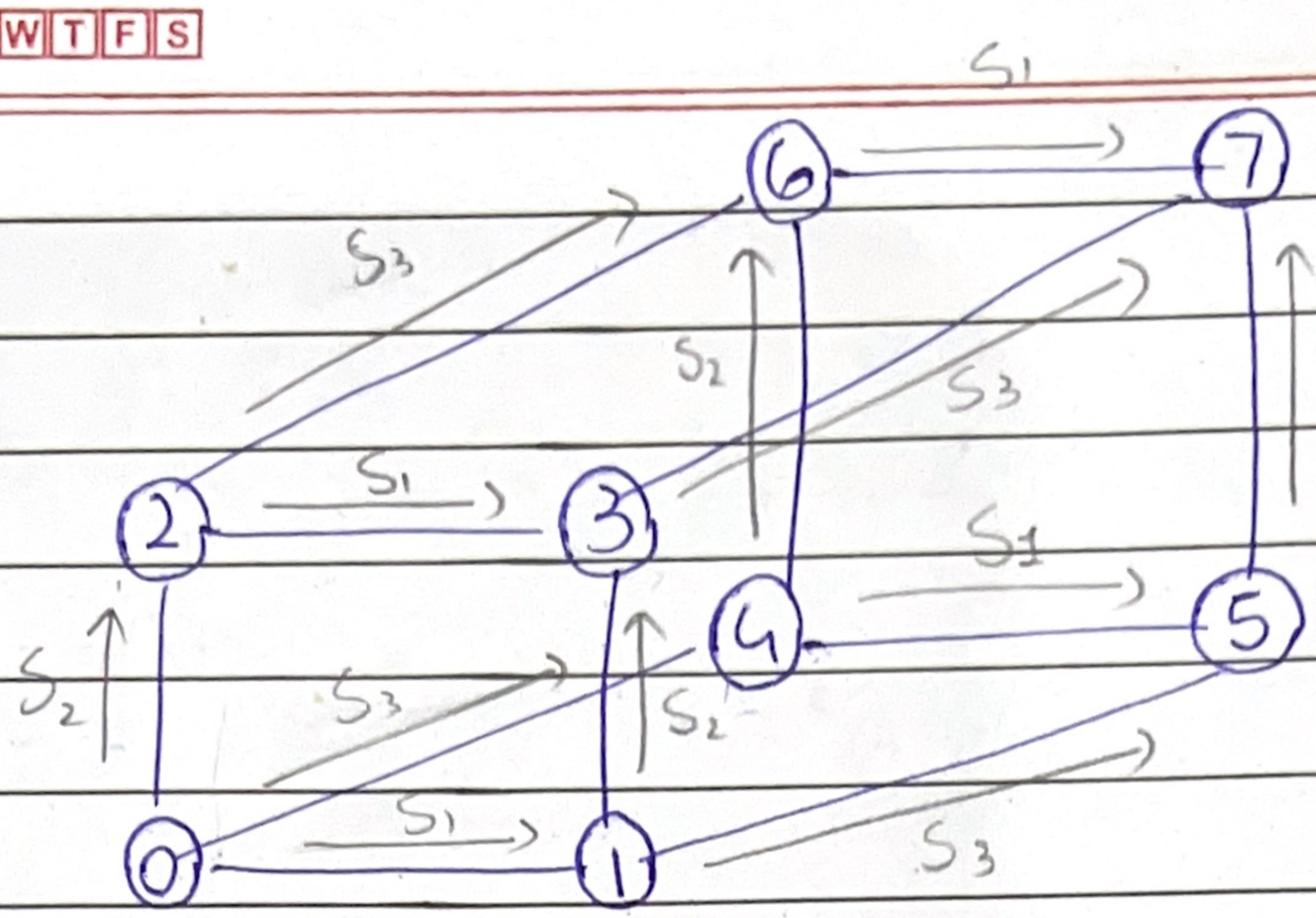
$= 2t_s(\sqrt{P}-1) + \sqrt{P} m t_w - m t_w + P m t_w - \sqrt{P} m t_w$

$$\boxed{\text{Cost} = 2t_s(\sqrt{P}-1) + m t_w (P-1)}$$

Date:

Day: M T W T F S

Mesh



S_1 :- Send in 1st Dimension (x-axis)
S_2 :- Send in 2nd Dimension (y-axis)
S_3 :- Send in 3rd Dimension (z-axis)

Cost => $\sum_{i=1}^{\log p} (t_s + 2^{i-1} m t_w)$ → The size of message is increasing with every step $S_1 = 1$, $S_2 = 2$, $S_3 = 4$

$$= \sum_{i=1}^{\log p} t_s + \sum_{i=1}^{\log p} 2^{i-1} m t_w$$

$$= (t_s + t_s + \dots \log p) + m t_w \sum_{i=1}^{\log p} 2^{i-1}$$

$$= \log p (t_s) + m t_w (2^0 + 2^1 + \dots 2^{\log p - 1})$$

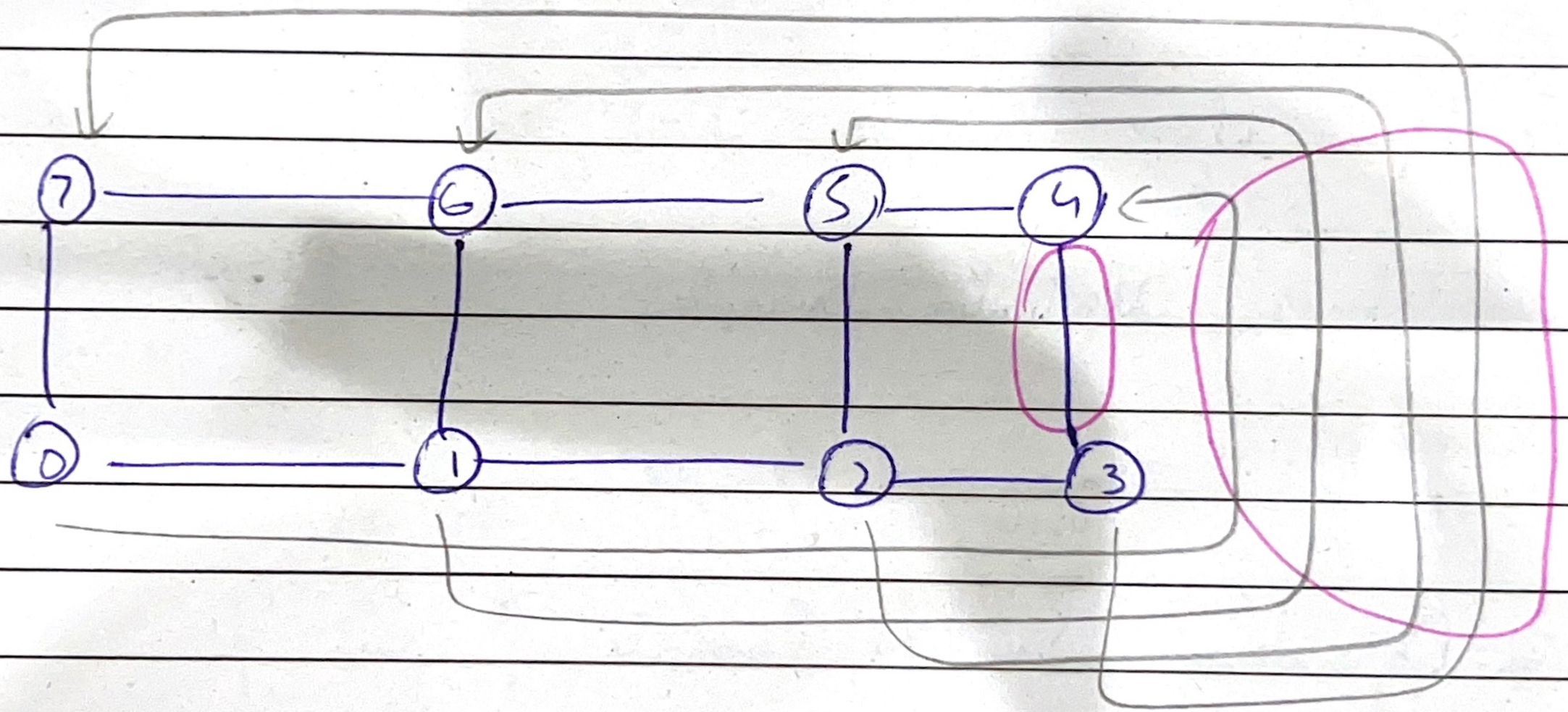
$\because x^0 + x^1 + x^2 \dots x^{p-1} = \frac{x^{n+1} - 1}{x - 1}$

$$= \log p (t_s) + m t_w \left(\frac{2^{\log p - 1 + 1} - 1}{2 - 1} \right)$$

$$= \log p (t_s) + m t_w (2^{\log p} - 1)$$

$$\boxed{\text{cost} = \log p (t_s) + m t_w (p - 1)}$$

In all to all broadcast every shape's calculation will be different as suppose Mesh can not be apply at ring as shown below



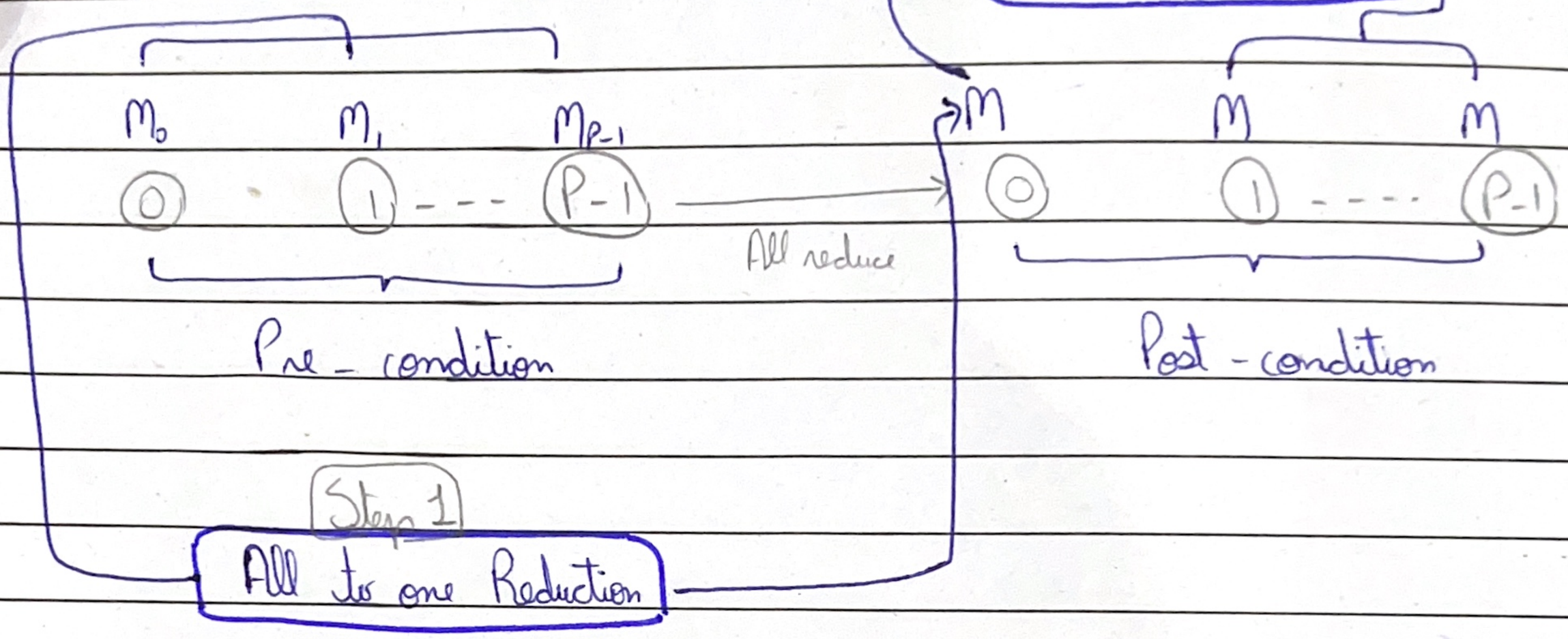
Here the link is getting conjested which we can not allow to happen. That is why in all to all broadcast the algorithms are different and in one to all same.

Date:

Day: M T W T F S

Basic Communication Operations :-

- All reduce (Third operation)
- Prefix Sum (Scan operation)
- Scatter & Gather



Step 1:

↳ All to one Reduction

cost = $\log p\ (t_s + mt_w)$

Step 2:

↳ one - to - all broadcast

cost = $\log p\ (t_s + mt_w)$

Total cost = Cost of Step 1 + Cost of Step 2

= $\log p\ (t_s + mt_w) + \log p\ (t_s + mt_w)$

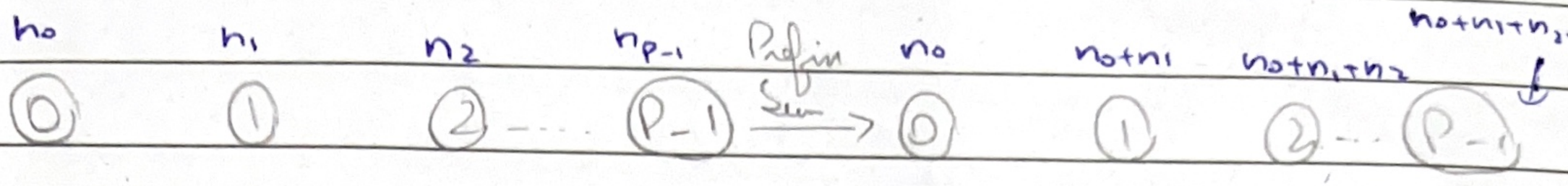
Total cost = $2 \log p\ (t_s + mt_w)$

* cost will be same in all structures like one - to - all broadcast.

Date: _____

Day: M T W T F S

Prefix Sum :-



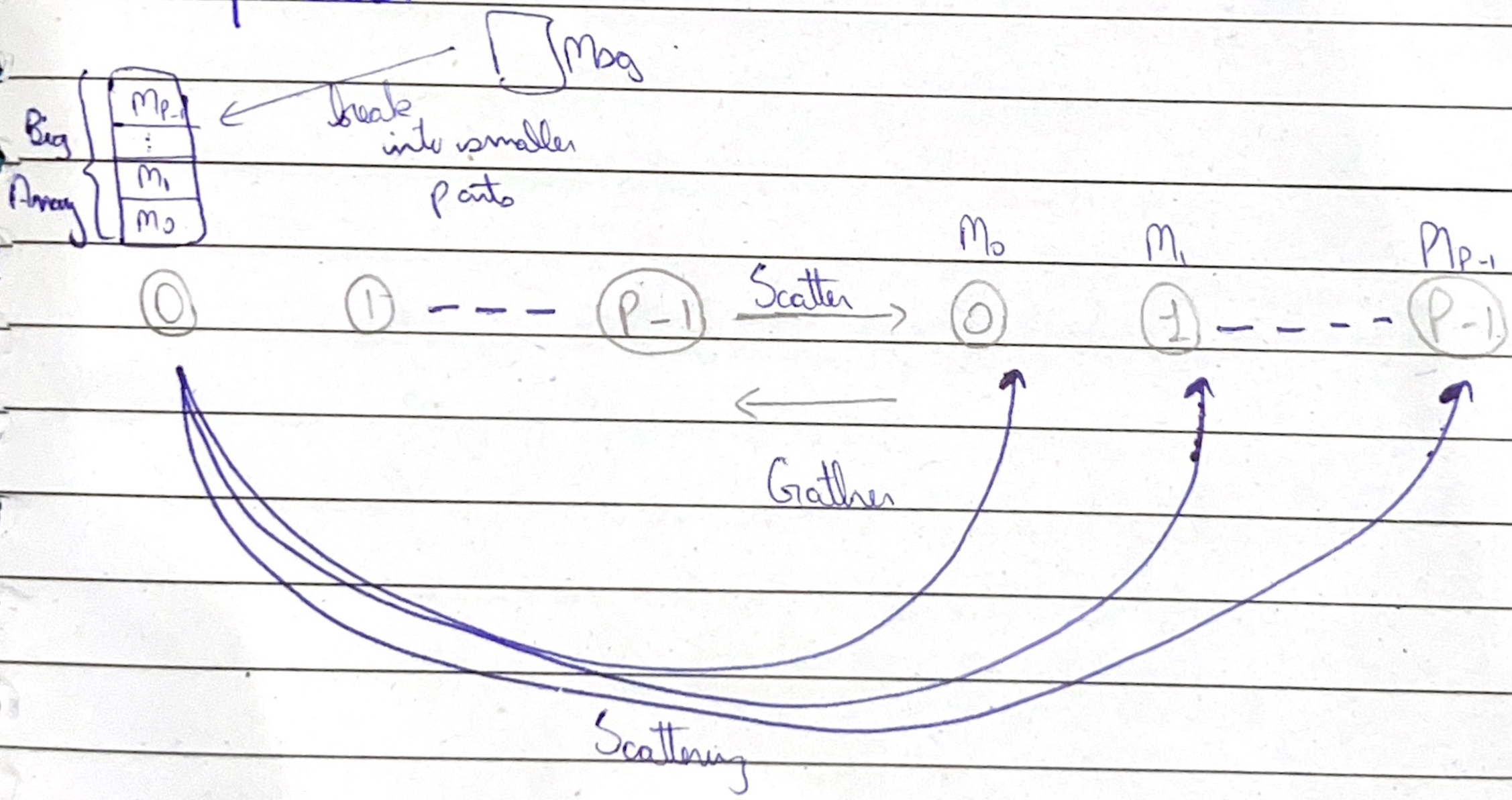
(0) → message

[0] ⇒ Result will accumulate

$$\text{Cost} = \sum_{i=1}^{\log p} (t_s + 2^{i-1} m t_w)$$

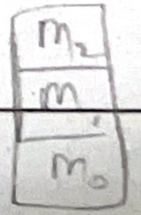
* Check slides for hypercube example.

Scatter & Gather :-



The difference between Scattering & One to all is that message is divided into chunks and chunks the send to each node

While the gather is different from All to one reduction in a way that all messages are not added upto to make a msg and send like $M_0 + M_1 + M_2 \Rightarrow M$ but in this case msg will just concatinate in an array of msg $M_0\ M_1\ M_2 \Rightarrow$

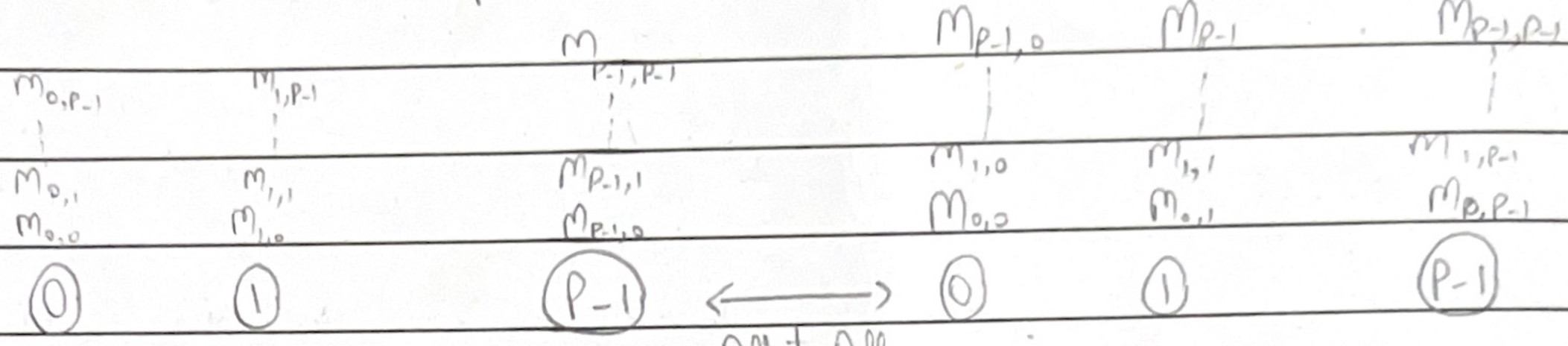


Date:

Day: M T W T F S

BASIC COMMUNICATION OPERATIONS :-

↳ All to All personalized :-



$M_{1,0}$

source ↳ destination

Prefix Sum = One to all = $\log p\,(t_s + m t_w)$

Scatter Cost = All to All = $\sum_{i=1}^{\log p} (t_s + 2^{i-1} m t_w)$

$p=1$

Cost in a linear array,

$$\sum_{i=1}^{P-1} [t_s + (P-i) m t_w)$$

$$\sum_{i-1}^{P-1} t_s + m t_w \sum_{i=1}^{P-1} (P-i)$$

$$\sum_{i=1}^{P-1} t_s + m t_w \sum_{i=1}^{P-1} (P-i)$$

$$\sum_{i=1}^{N} i = \frac{n(n+1)}{2}$$

$$= (P-1)(t_s) + m t_w \left(\sum_{i-1}^{P-1} P - \sum_{i=1}^{P-1} i\right) \left(m t_w \frac{(P-1)(P-1+1)}{2}\right)$$

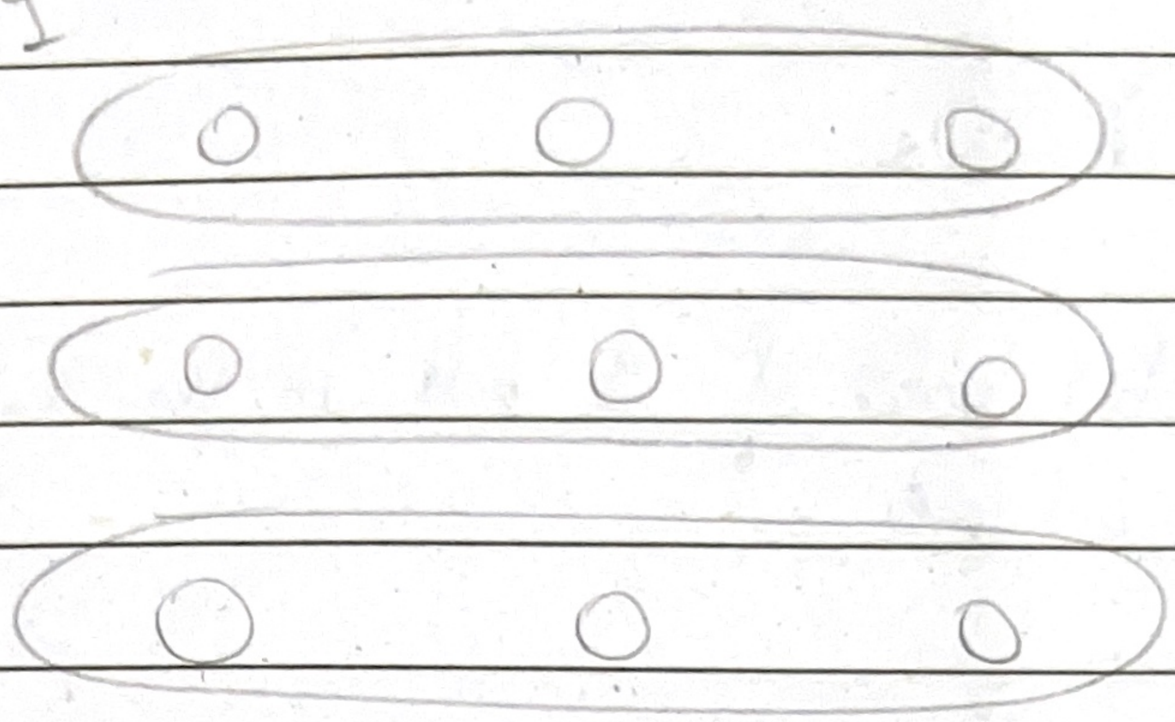
$$= (P-1)(t_s) + m t_w (P-1)P - \tfrac{1}{2} m t_w (P-1)P$$

$$= (P-1)\left[t_s + m t_w P - \tfrac{1}{2} m t_w P\right]$$

$$\text{cost} = (P-1)(t_s + \tfrac{1}{2} m t_w P)$$

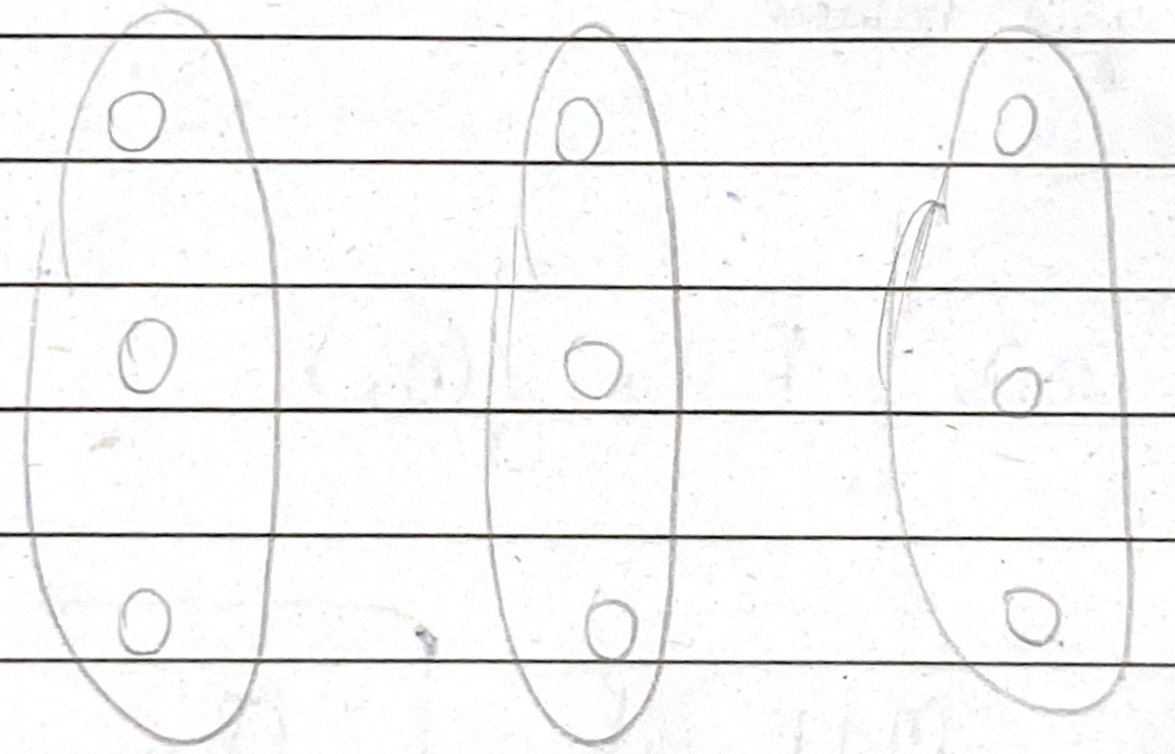
2D - Mesh :

Step 1



send messages row wise, each node will send three msgs one for next & two for upper in S_2

Step 2



cost = row-wise + col-wise

row-wise = $(t_s + 1/2\, m t_w)(\sqrt{p} - 1)$ → Iterations of mesh

$= (t_s + 1/2\, \sqrt{p}\, m t_w \sqrt{p})(\sqrt{p} - 1)$

$= (t_s + 1/2\, m t_w p)(\sqrt{p} - 1)$

row-wise cost == col-wise cost S_2.

Total cost = $2(t_s + 1/2\, m t_w p)(\sqrt{p} - 1)$