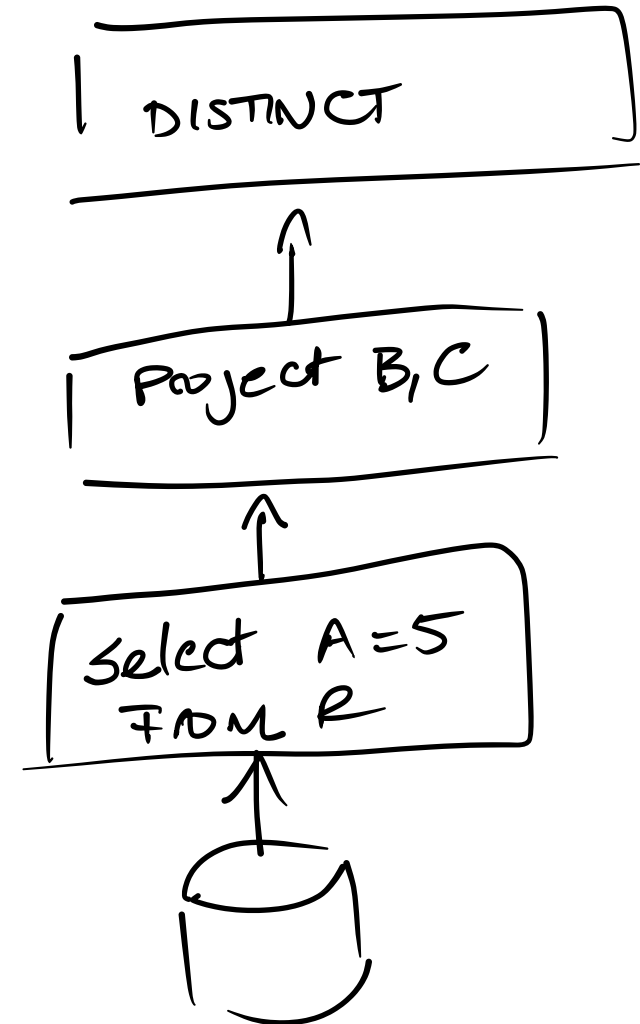
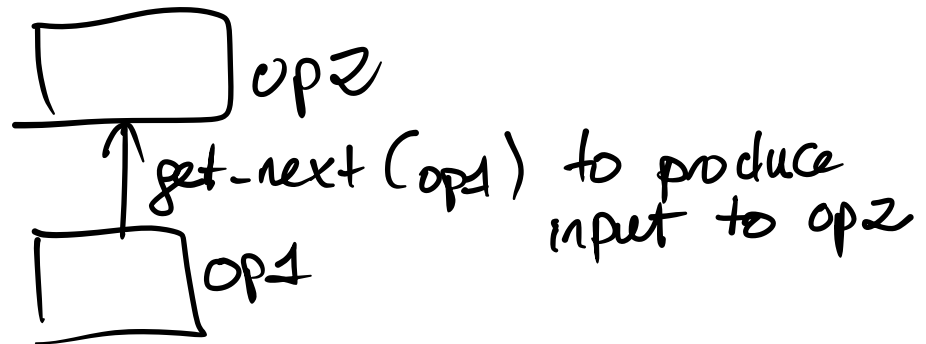
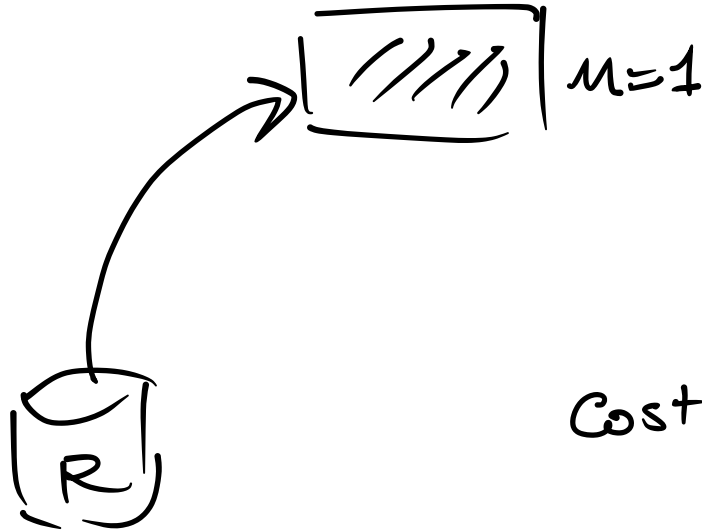


# Iterator Interface



Sequential Scan  
 $\sigma_C(R)$

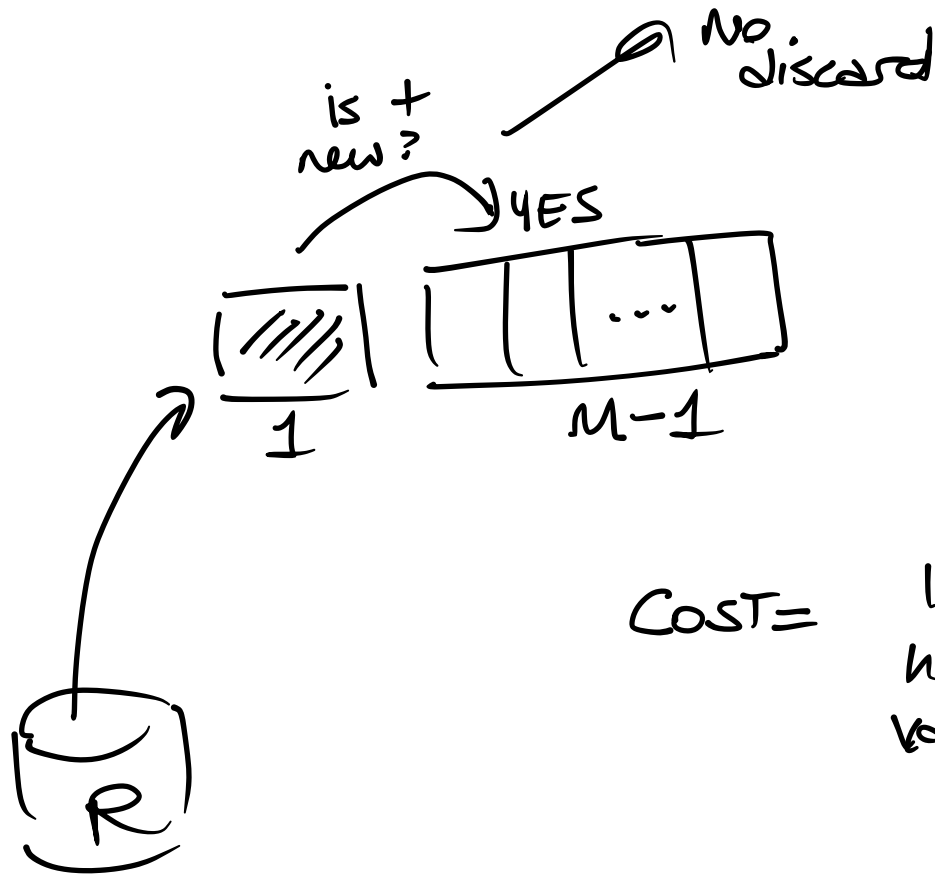
if t.c  
is true → put into  
output buffer



Cost = PAGES  $(R)$

# Duplicate Removal (SR)

"DISTINCT"

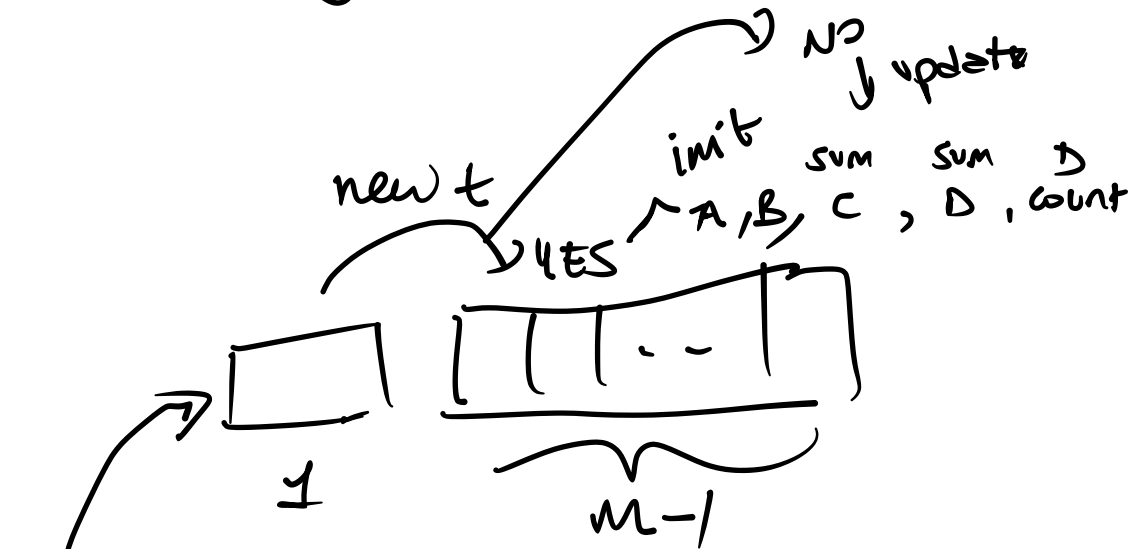


COST =

If  $M-1$  blocks  
hold all unique  
values  $\rightarrow$

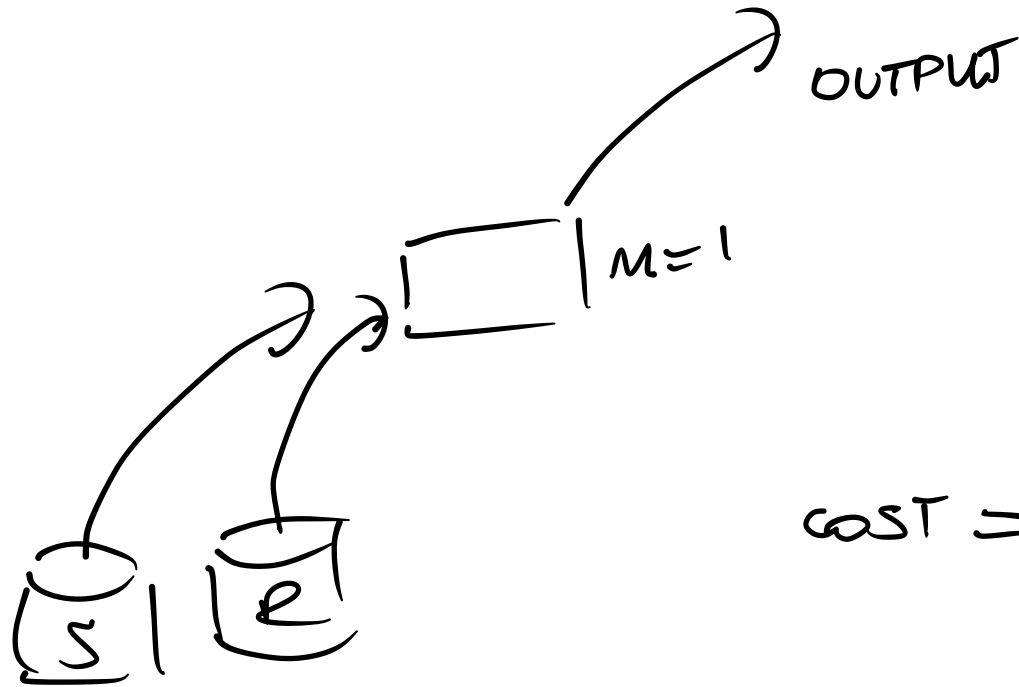
PAGES (R)

GROUP BY  
 $\delta_{A,B, \text{sum}(C), \text{avg}(D)}$



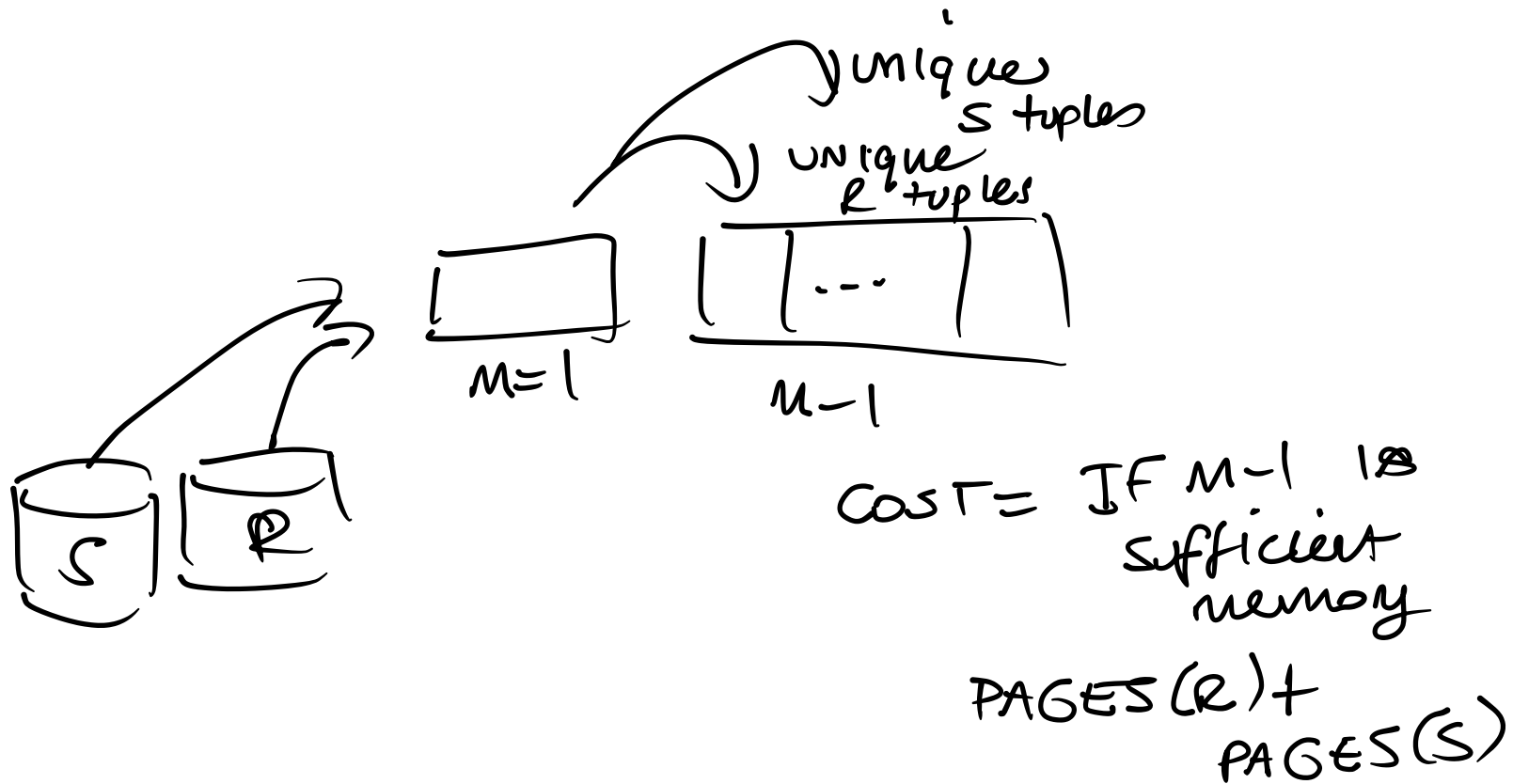
$\text{COST} =$  If results fit  
 in  $M-1$   
 blocks  
 $\text{PAGES}(R)$

R UNION ALL S



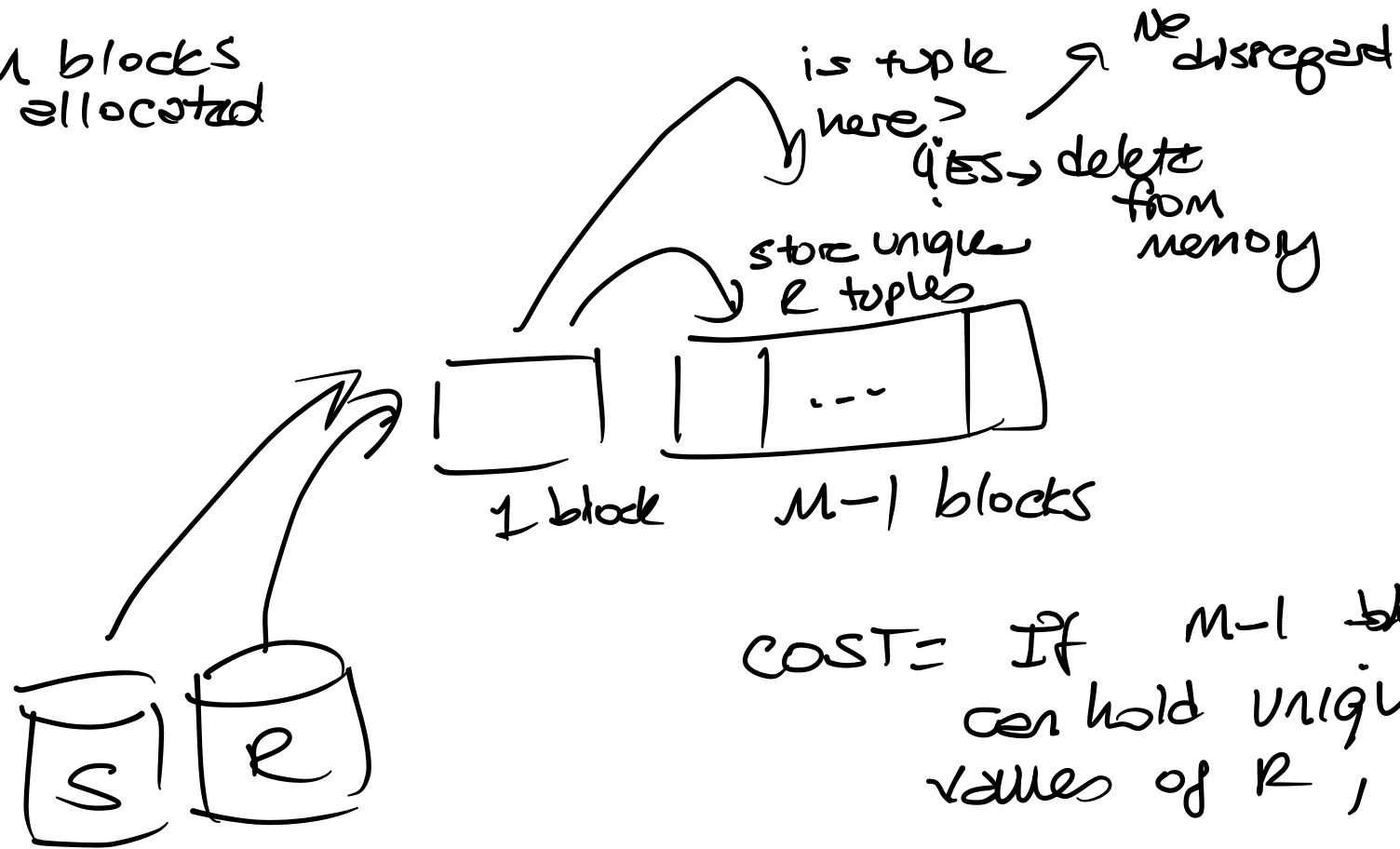
$$\text{COST} = \text{PAGES}(R) + \text{PAGES}(S)$$

# R UNION S



# R EXCEPT S

M blocks allocated



COST = If M-1 block  
can hold unique  
values of R,

then

$$PAGES(R) + PAGES(S)$$

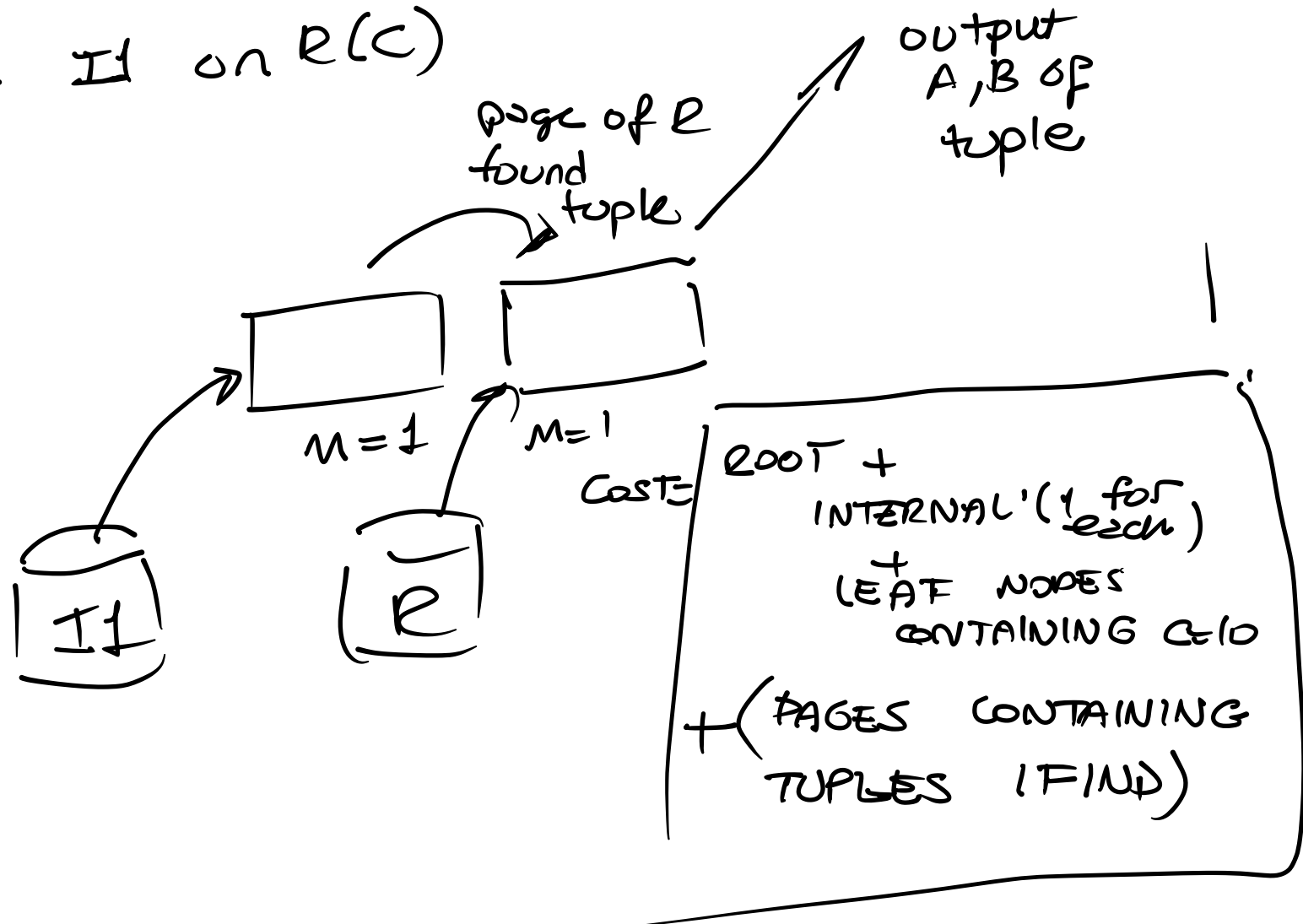
# Index Scan

SELECT A, B

FROM R

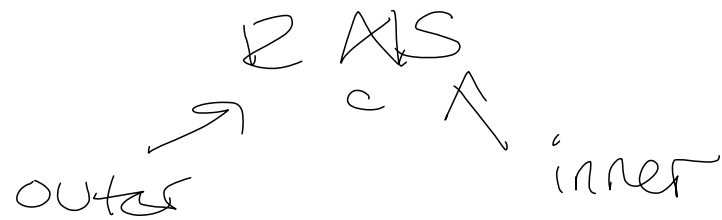
WHERE C = 10;

Index I1 on R(C)

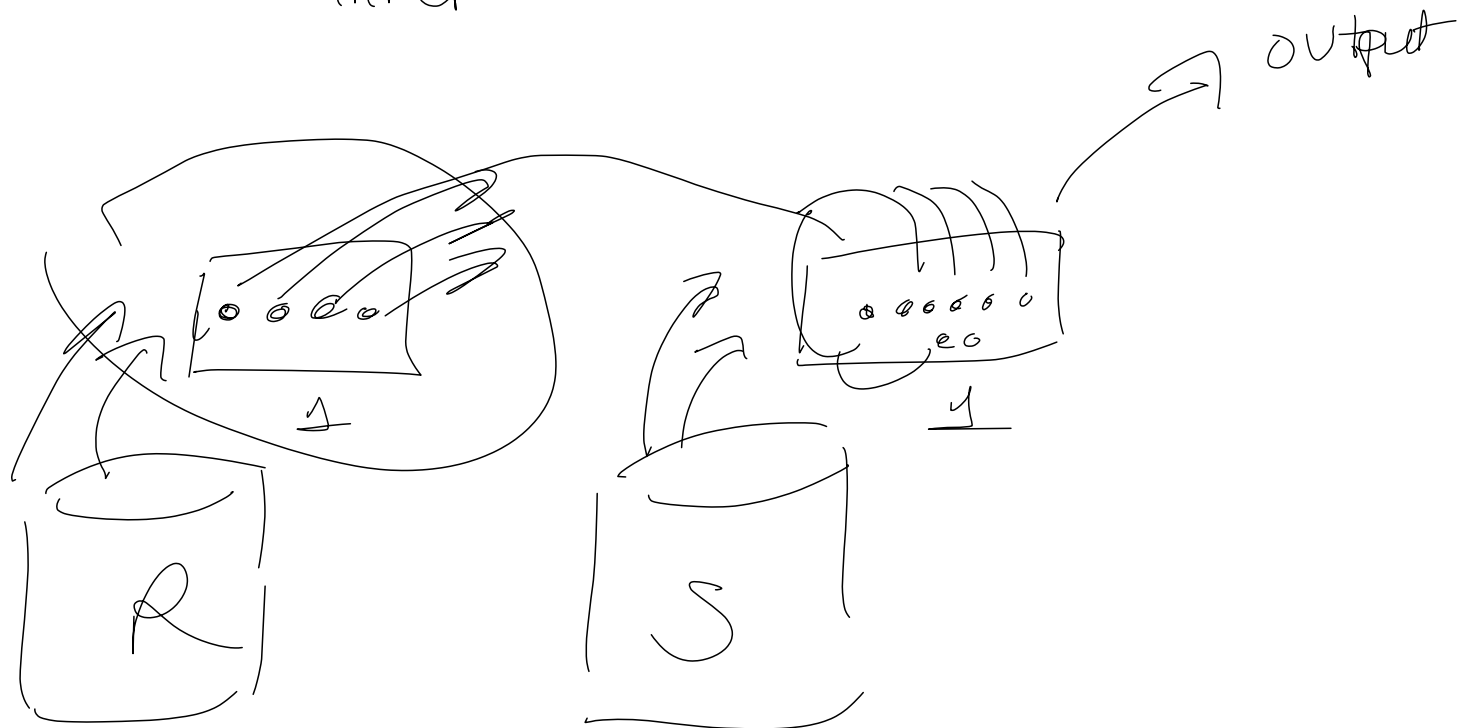




# Block nested loop join



$$M = 2$$



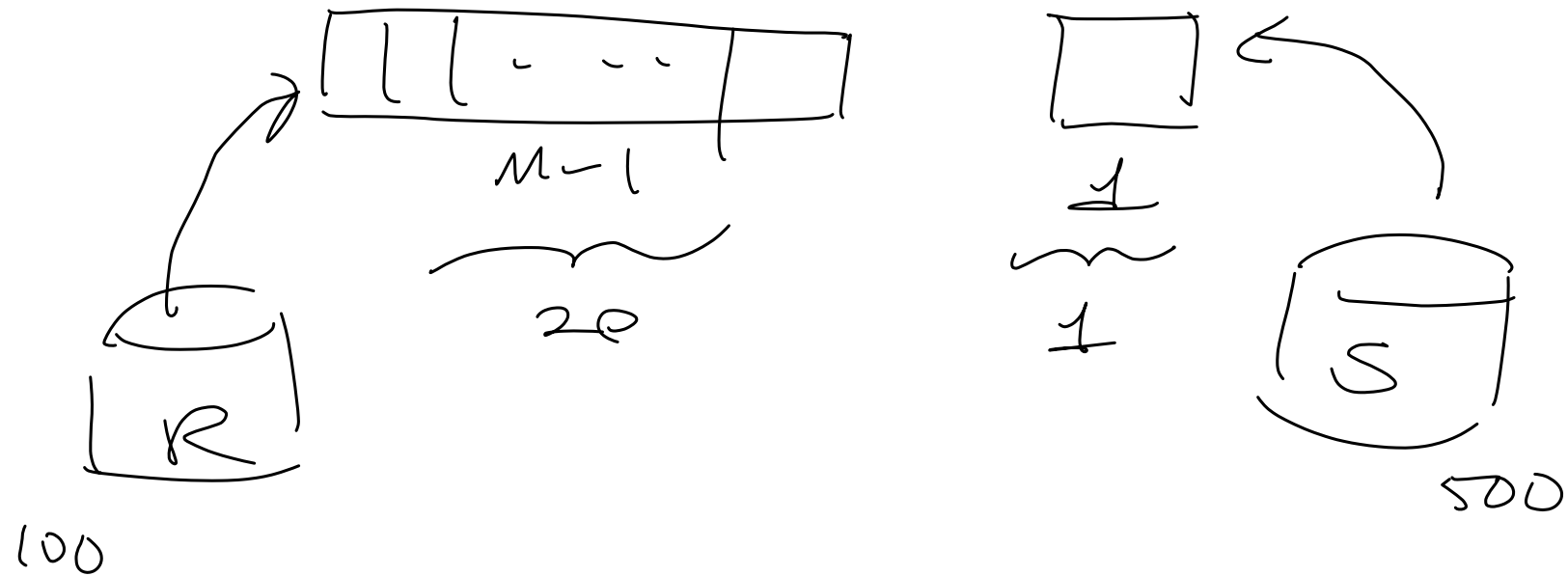
How many times do we read  $S$ ?  
pages( $R$ ) times

How many times do I read  $R$ ? 1 time

$$\text{Cost} = \text{PAGES}(R) + \text{PAGES}(S) * \text{PAGES}(R)$$

Block nested loop join

$M > 2$



How many times do we read S?  
(5)

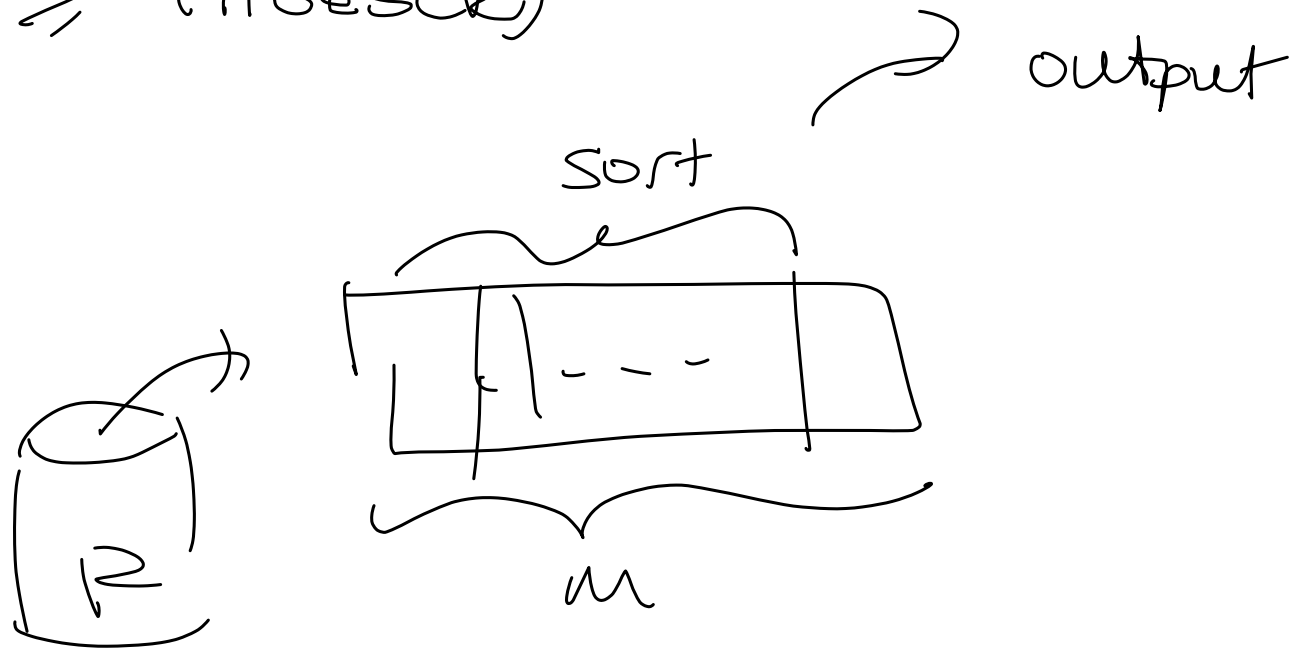
$$\text{PAGES}(R) + \left\lceil \frac{\text{PAGES}(R)}{M-1} \right\rceil * \text{PAGES}(S)$$

PAGES(R)	PAGES(S)	M	COST RAS
100	500	2	$100 + 100 * 500$
100	500	101	$100 + 500$
100	500	21	$100 + 5 * 500$
500	100	2	$500 + 500 * 100$
		101	$500 + 5 * 100$
		21	$500 + 25 * 100$

# EXTERNAL SORT

$M$  blocks & buffers

$$M \geq \text{PAGES}(R)$$

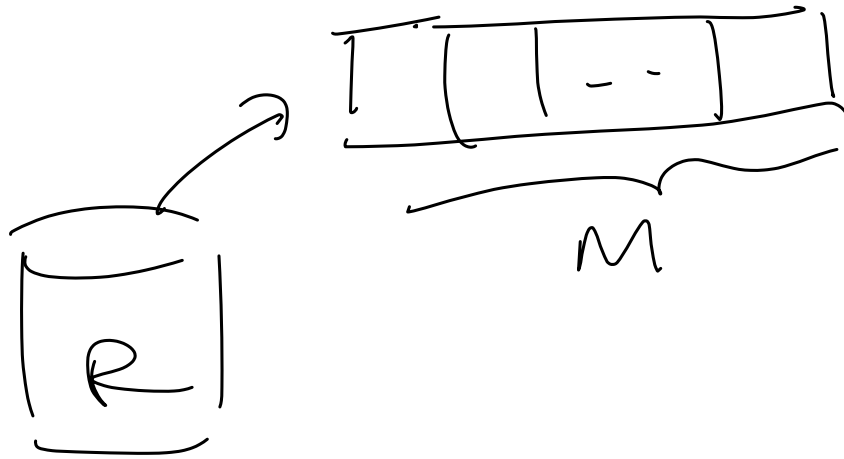


$$\text{COST} = \text{PAGES}(R)$$

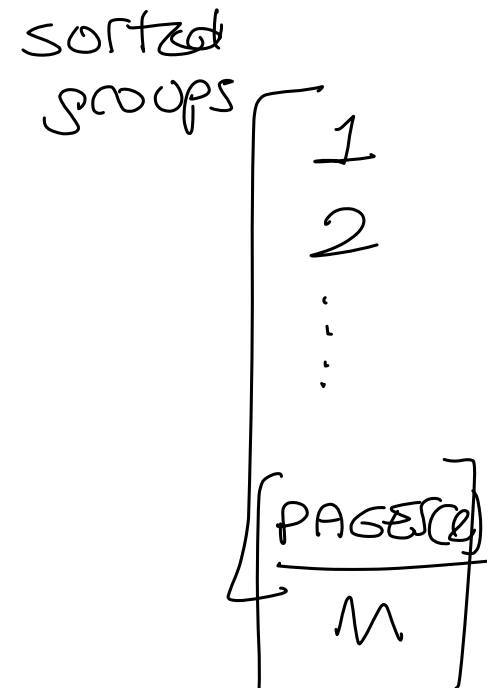
EXTERNAL SORT  $M \ll \text{PAGES}(R)$

2 phases  $\Rightarrow$  assume  $M$  is fixed throughout

Phase 1: Read  $\rightarrow$  sort  $\rightarrow$  write to temporary



write to disk



$$\text{cost} = 2 * \text{PAGES}(R)$$

# EXTERNAL SORT

## PHASE 2 → MERGE

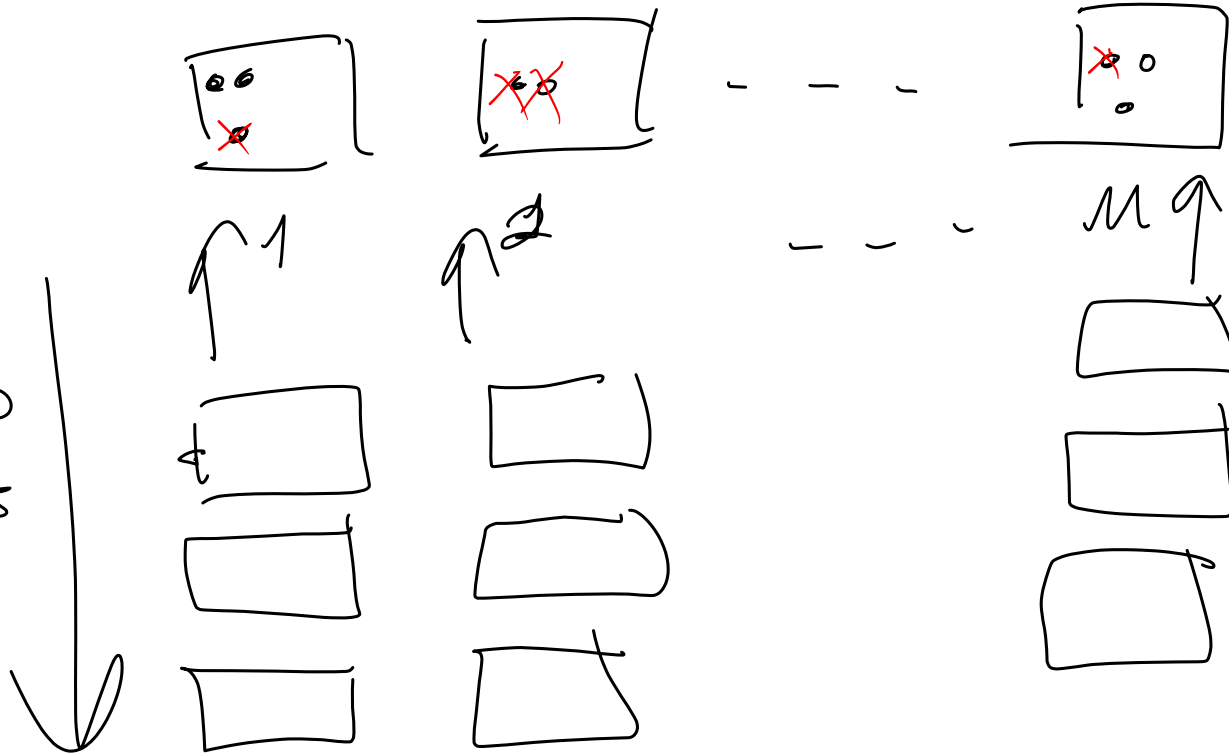
M

If  $M \geq \# \text{sorted groups}$

find smallest tuple

output

SORTED  
GROUPS



# EXTERNAL SORT

## PHASE 2 → MERGE

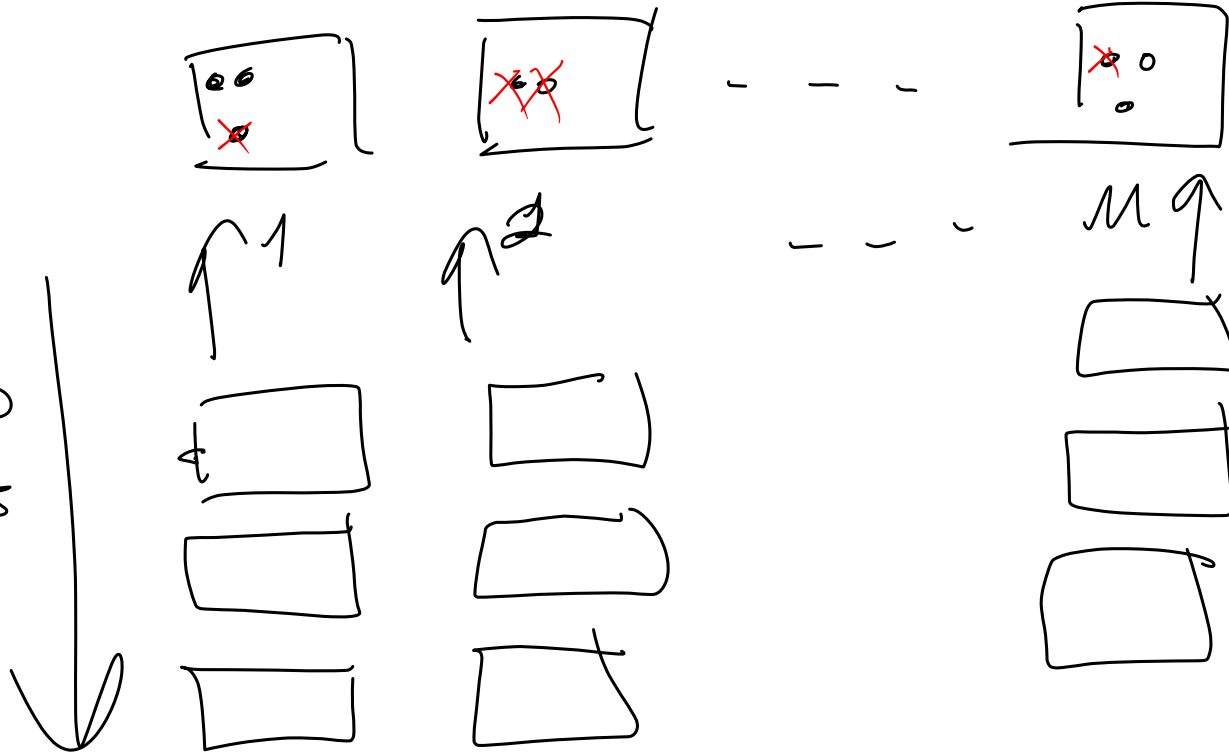
M

If  $M < \# \text{sorted groups}$

find smallest tuple

write to disk

SORTED GROUPS



# EXTERNAL SORT

PAGES(R)	M	PHASE 1 COST	# SORTED GROUPS	PHASE 2 read + merge + output
100	10	200	10	100
200	<u>10</u>	400	<u>20</u>	<p>read + merge + output</p> <p>400</p> <p>2 sorted groups now</p> <p><math>2 \leq M \rightarrow</math></p> <p>read + merge</p> <p>output <math>\rightarrow 200</math></p>
1000	10	2000	100	



PAGES(R)  $\rightarrow$  2000

M  $\rightarrow$  25

Phase 1  $\rightarrow$  COST = 4,000

(read + write)

$$\frac{2000}{25} = 80$$

Phase 2  $\rightarrow$  COST = 4,000

(read + write)

$$\left\lceil \frac{80}{25} \right\rceil = 4$$

80 sorted groups is reduced

to 4 sorted groups

Phase 2  $\rightarrow$  COST = 2,000

(read + output)

total = 10,000 pages

# EXTERNAL SORT EXAMPLE

PHASE

PAGES  $(C) = 8$

$M = 4$

1, 2, 3, 4,  
5, 6, 7, 8

~~14 15 18~~

~~1 2 4~~

~~5 7 8~~

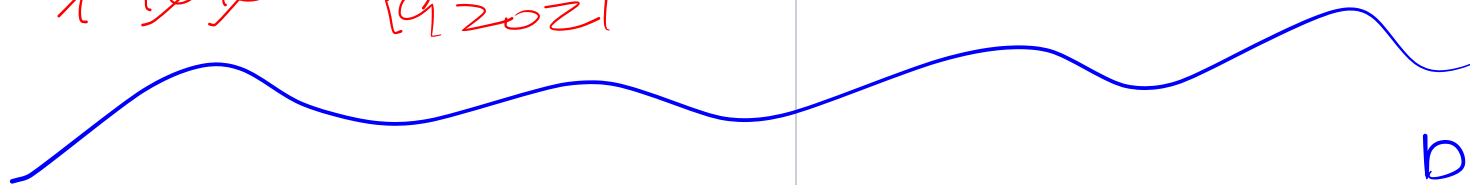
~~9 10 13~~

~~3 6 11~~

~~12 16 18~~

~~19 20 21~~

MEMORY



DISK

18 14

7 13 18

29 15

4 5 6

3 11 16

6 12 18

19 21 22

20 23 28

~~1 2 4~~

~~5 7 8~~

~~9 10 13~~

~~14 15 18~~

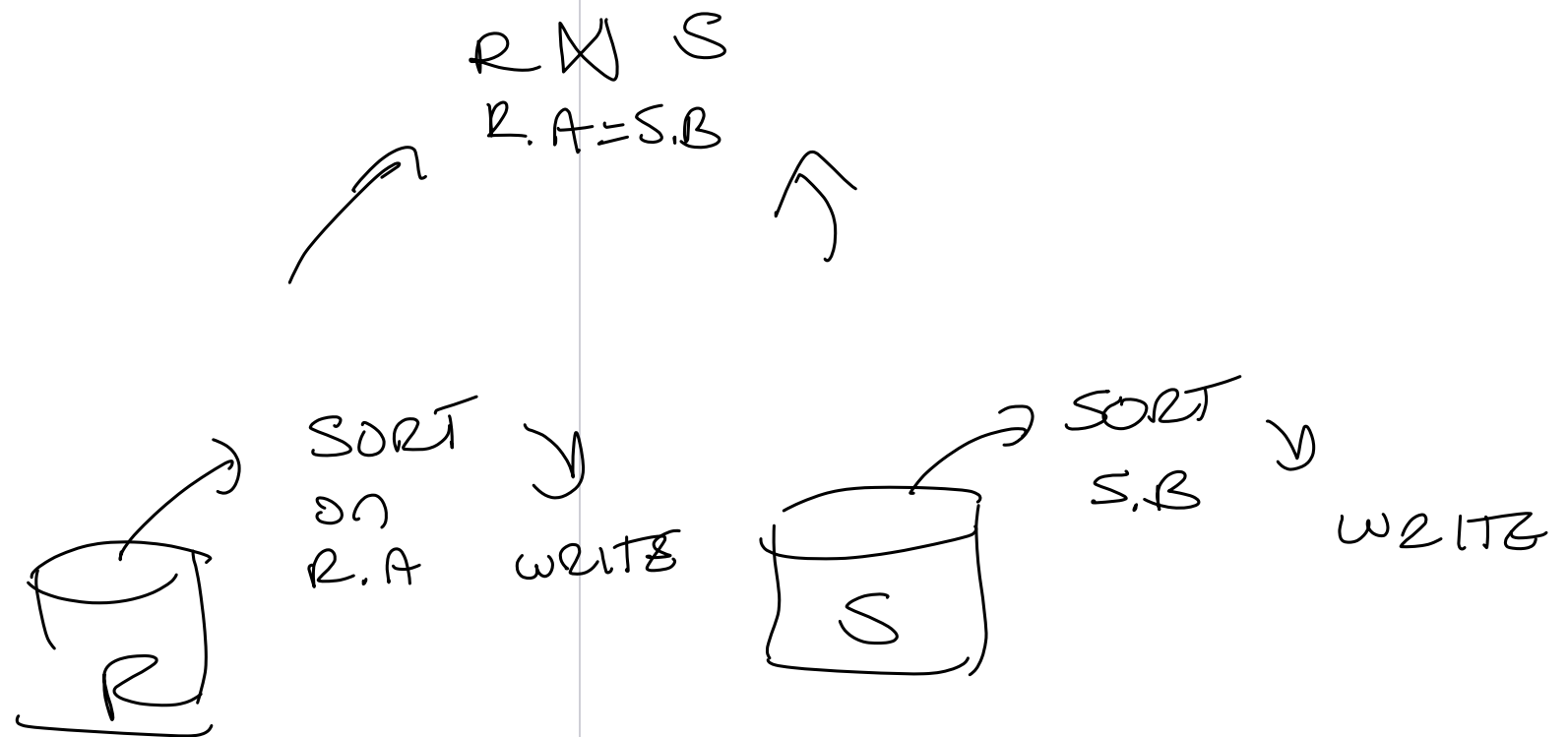
~~3 6 11~~

12 16 18

19 20 21

22 23 28

ORDER BY  $\rightarrow$  SORT  
SORT MERGE JOIN



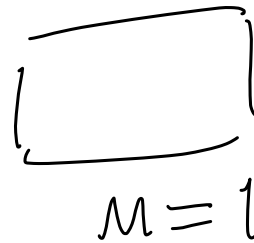
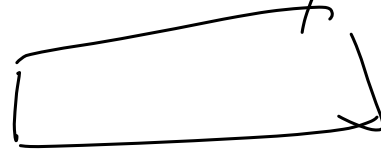
GROUP BY A

count (B)

sum (C)



sort  
on  
L.A



GROUP BY

current A

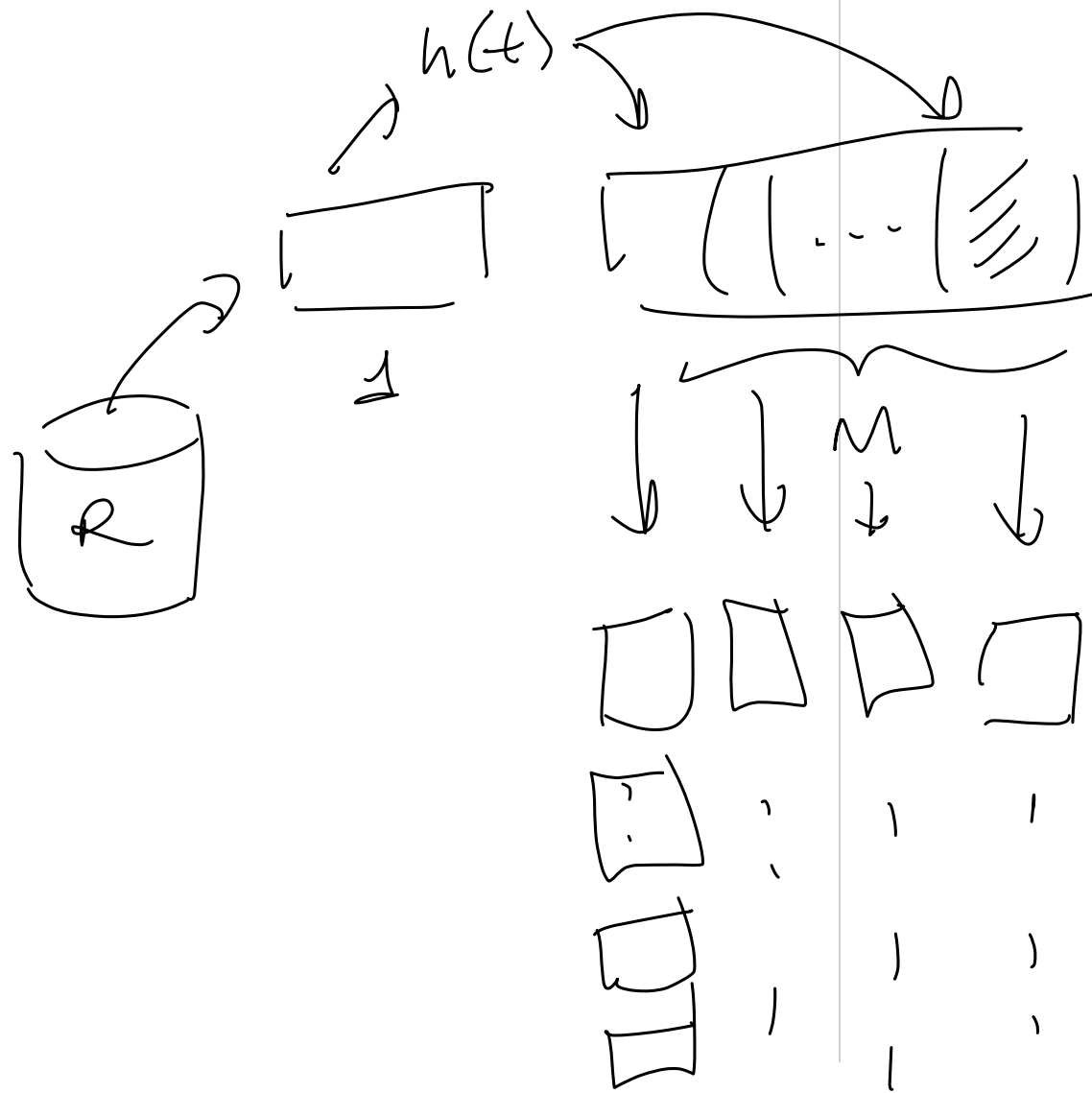
current count of

B

sum of  
C

HASHING

$M+1$  blocks



Each bucket  
roughly has

$$\left\lceil \frac{\text{PAGES}(R)}{M} \right\rceil$$

pages

BUCKETS  
 $1 \dots M$

HASH  $\rightarrow$  REMOVE DUPLICATES  
GROUP BY

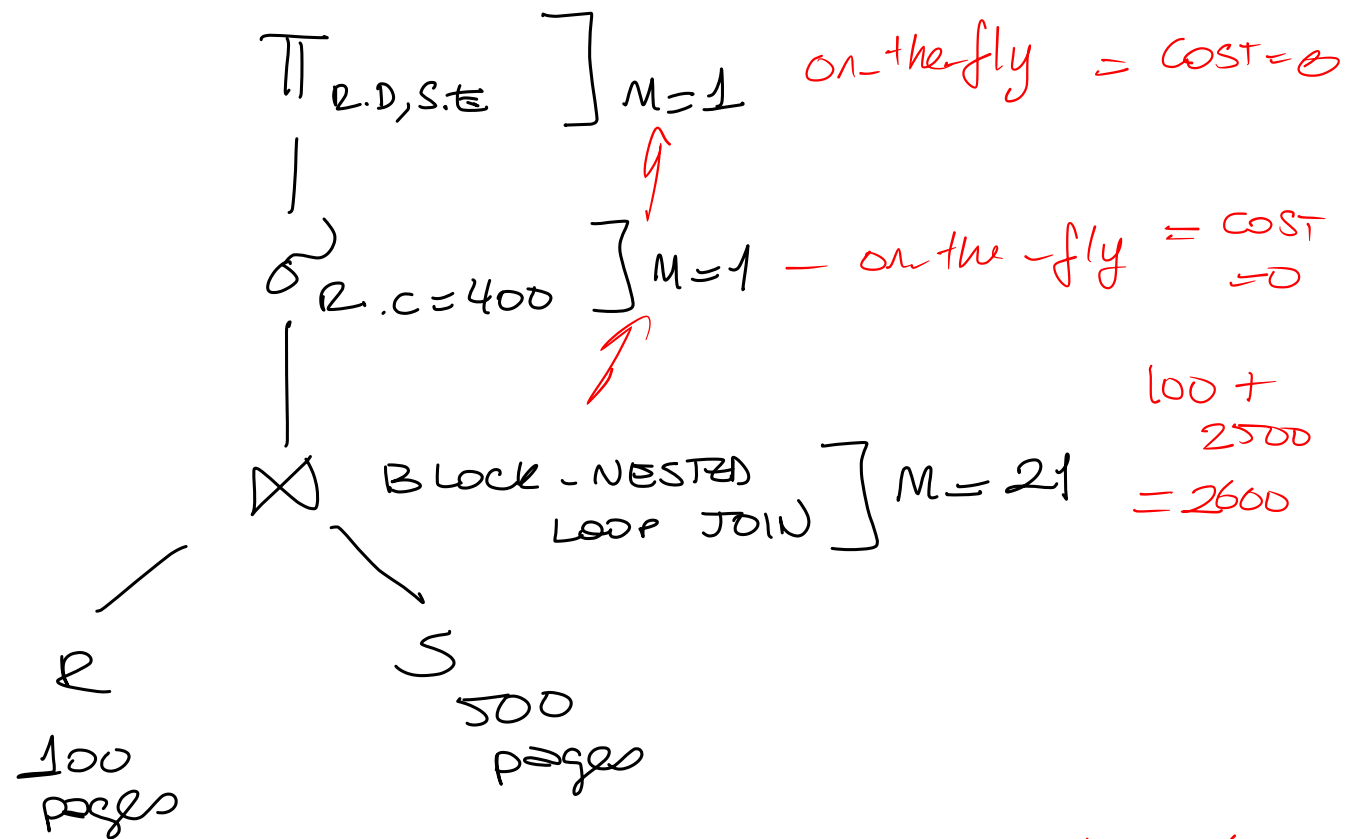
UNION / DIFFERENCE / INTERSECTION

} APPLY  
TO  
EACH  
BUCKET  
SEPARATELY

---

HASH JOIN  $\rightarrow$  HASH(R)  
HASH(S)

# QUERY PLAN COST ESTIMATION



total cost = 2600

# QUERY PLAN COST ESTIMATION

$$\text{Total Cost} = (\text{Cost of index scan}) + 500 \text{ (read S once)}$$

