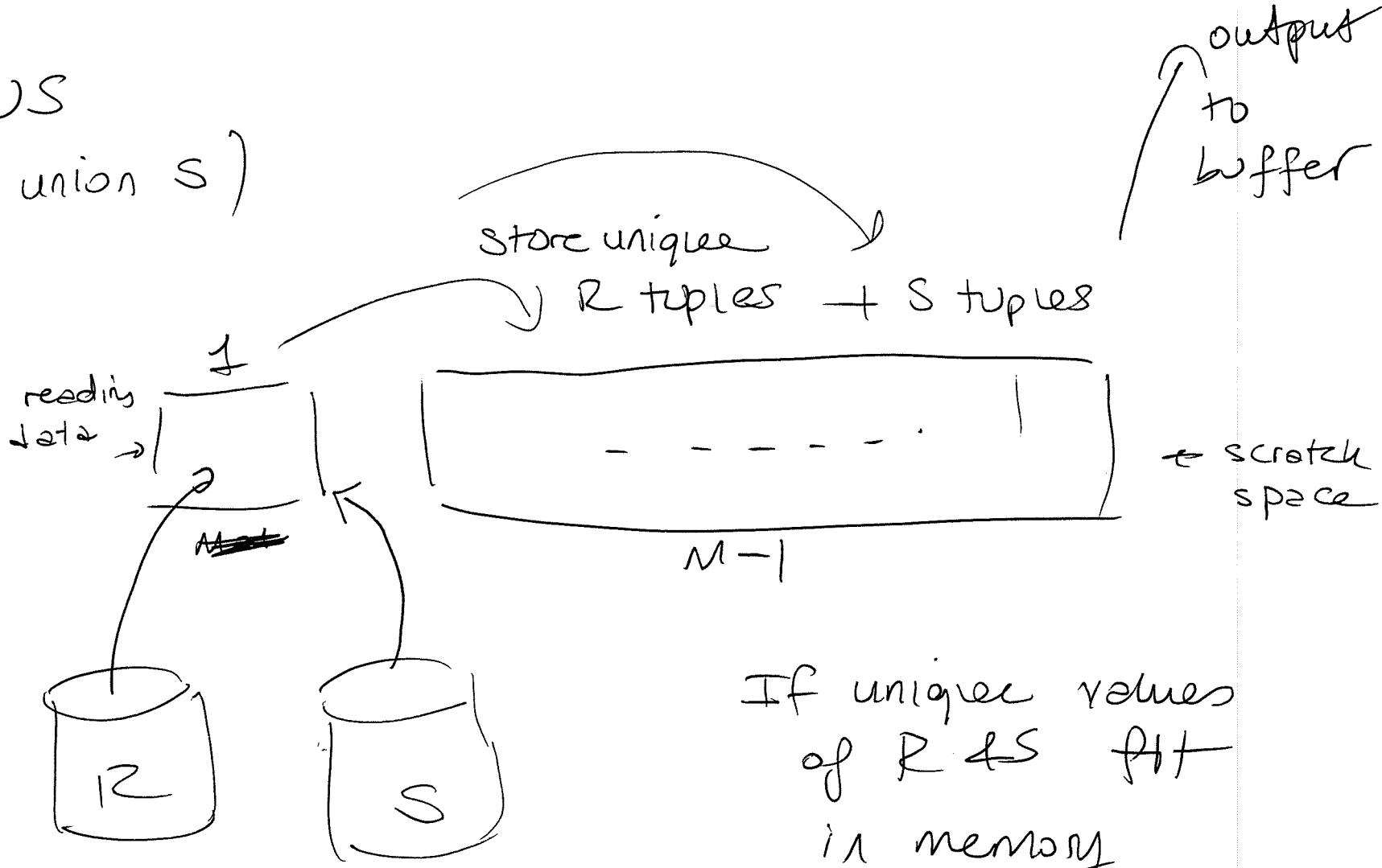
R ∪ S

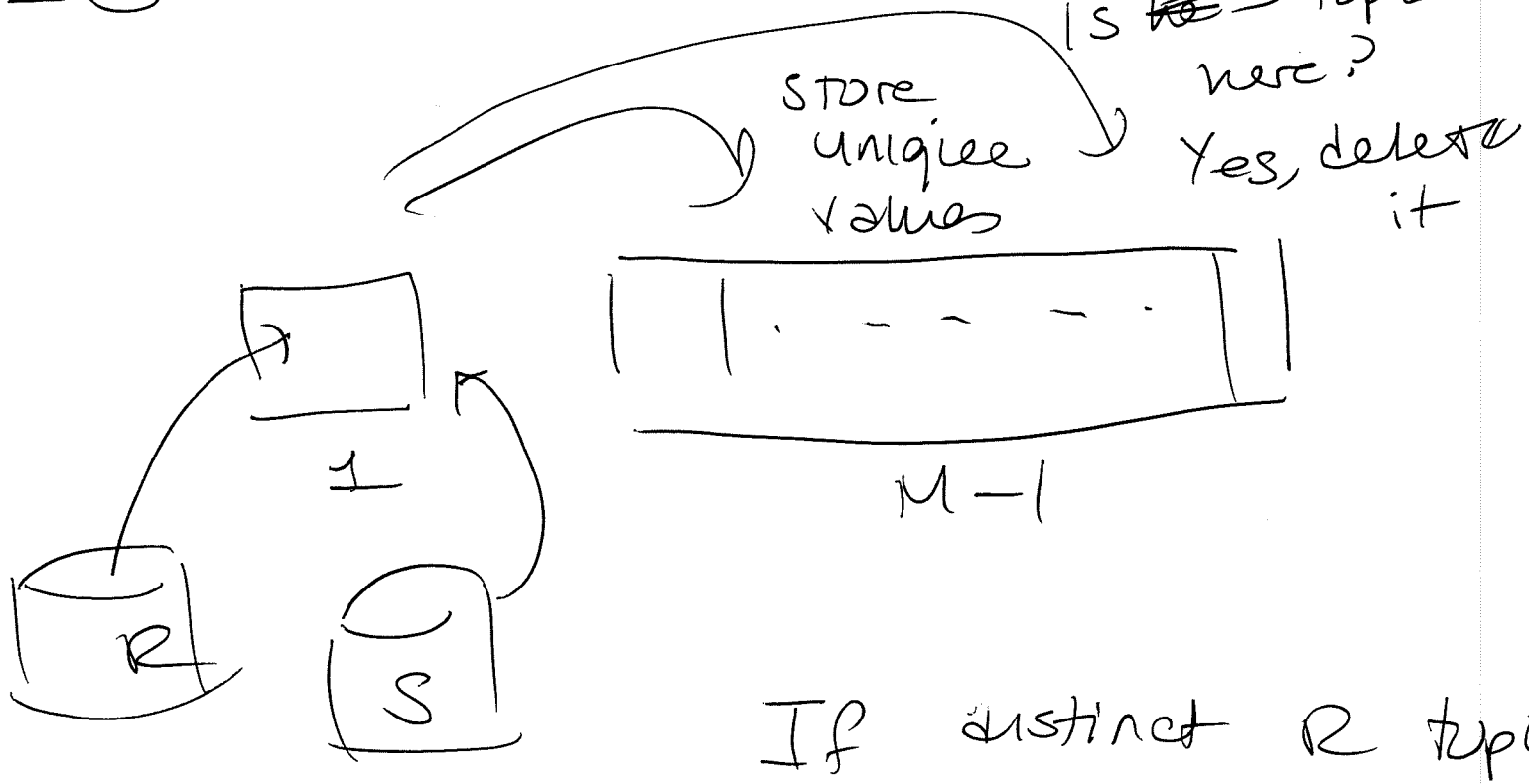
(R union S)



If unique values of R & S fit in memory

COST = PAGES(R) + PAGES(S)

R − S

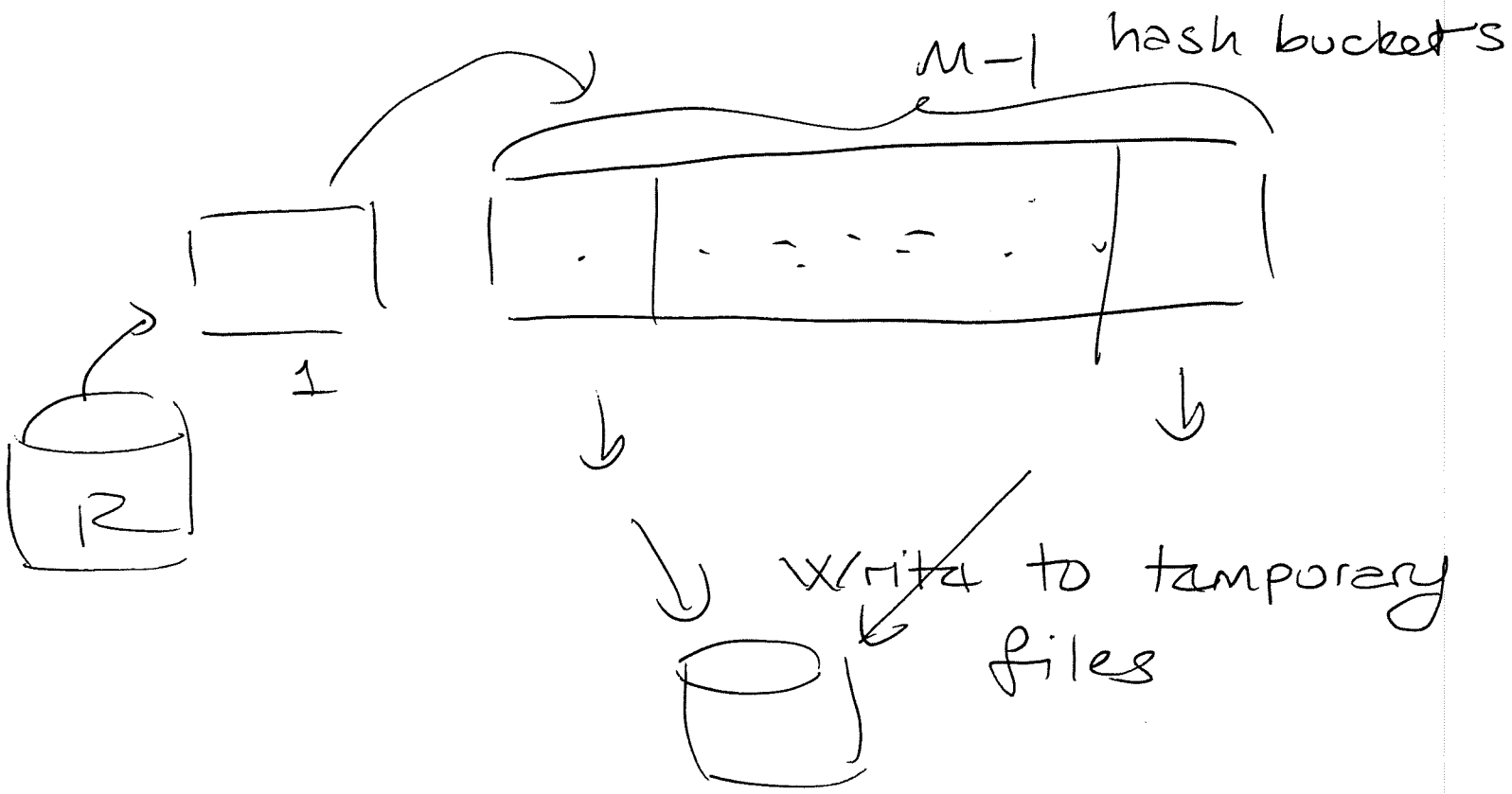


If distinct R tuples fit in memory

COST = PAGES(R) + PAGES(S)

HASHING

PAGES(R) $\gg$ M



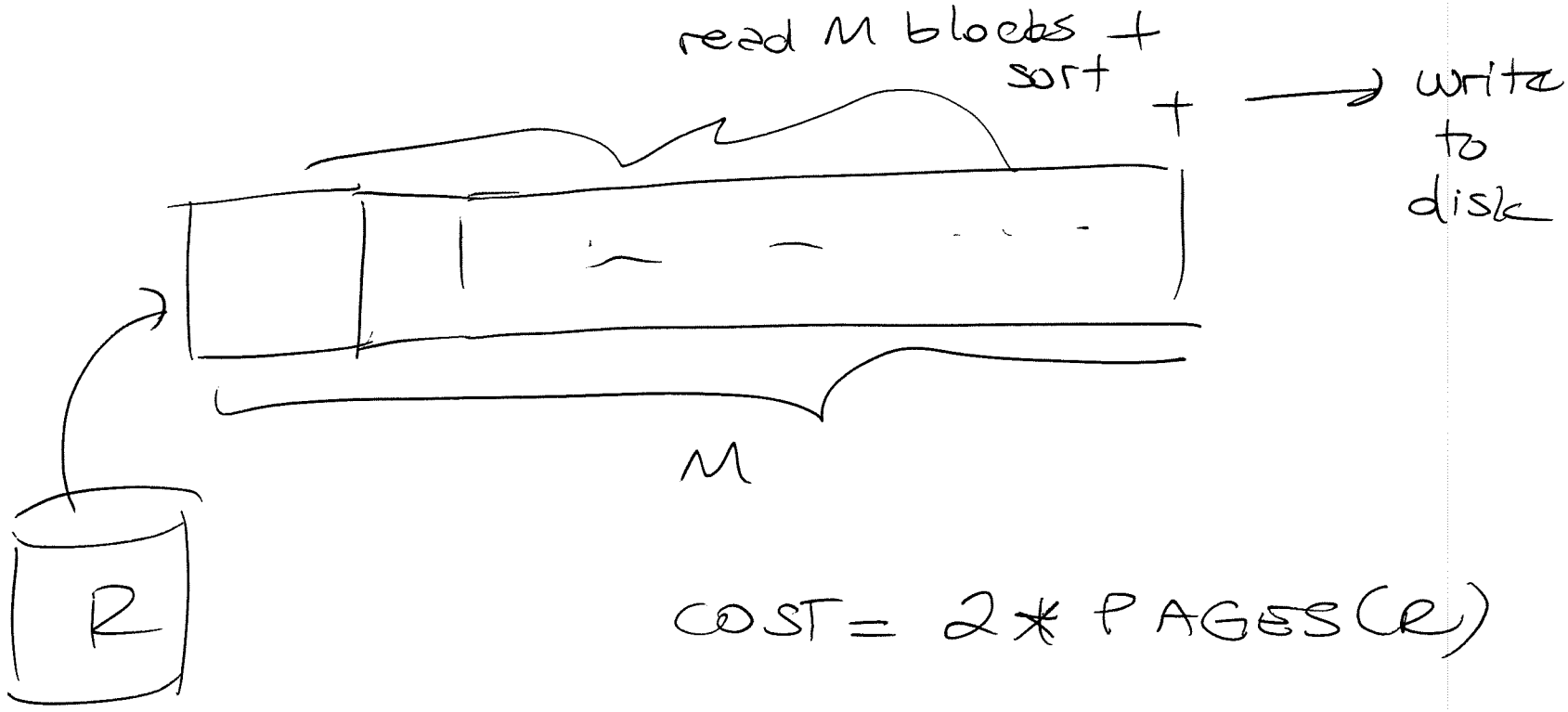
COST OF HASHING = 2 * PAGES (R)

SORTING

STEP 1:

EXTERNAL SORT

PAGES(R) >> M



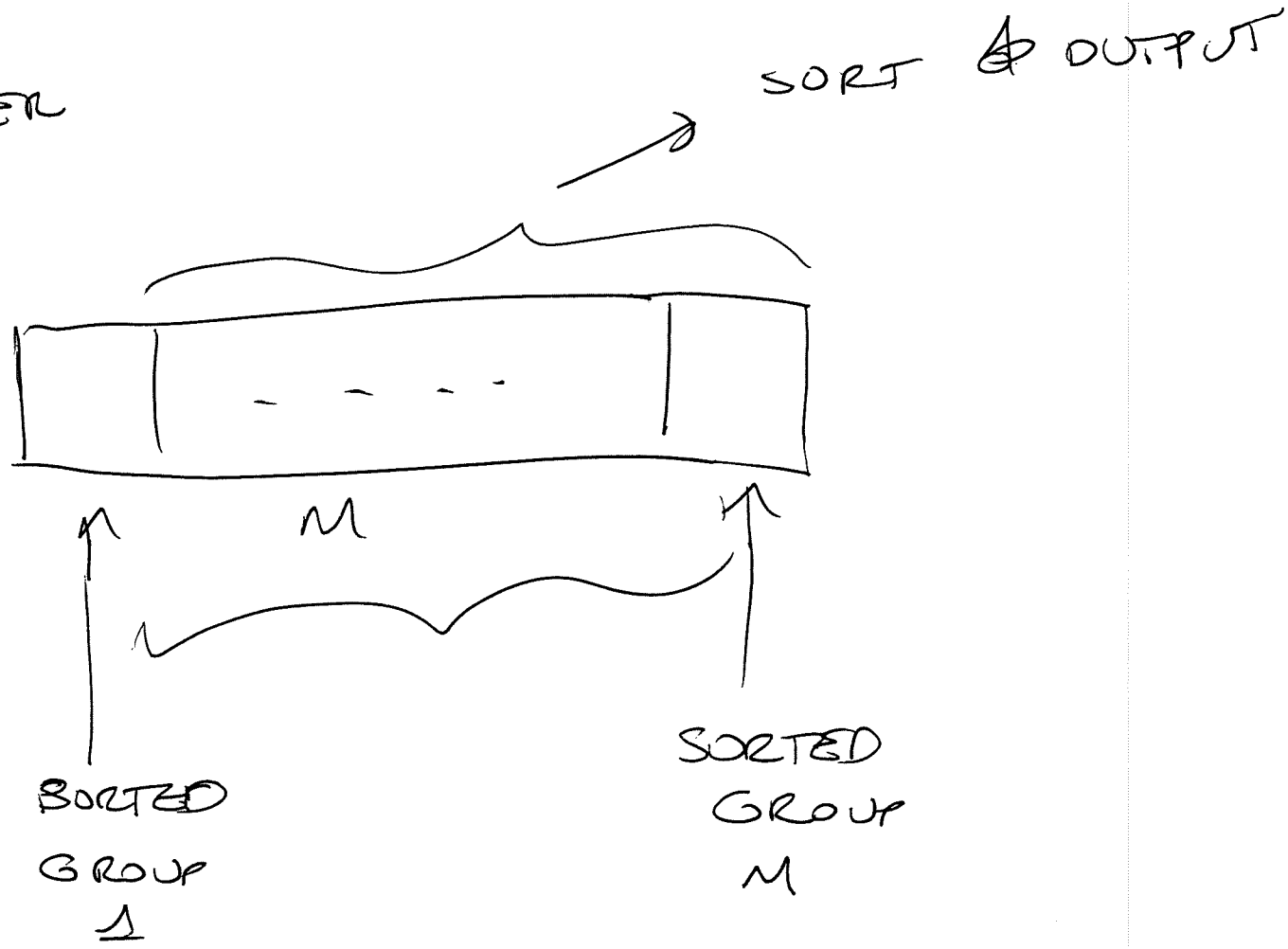
COST = 2 * PAGES(R)

ONE READ & ONE WRITE

EXTERNAL SORTING

STEP 2:

SUPPOSE NUMBER OF SORTED GROUPS $\leq M$



COST = PAGES (R)

PAGES (R)	M	STEP 1:	STEP 2:
10,000	10,000	Read + sort + output cost = PAGES(R)	TOTAL = 6,000
10,000	2,000	Read + write cost = 20,000 (5 sorted groups)	Read + merge + output cost = 10,000 Total cost = 30,000
10,000	200	cost = 20,000 50	Read + merge + output cost = 10,000 Total cost = 30,00
10,000	20	Cost = 20,000 sorted groups 500	Merge 20 groups at a time + write Cost = 20,000 sorted groups 25

Repeat!
Merge + output
cost = 10,000

Repeat

Step 2 → get 2 sorted groups → cost = 20,000

PAGES(R)	M
100,000	20

Step 1

20,000 (R/W)

Step 2

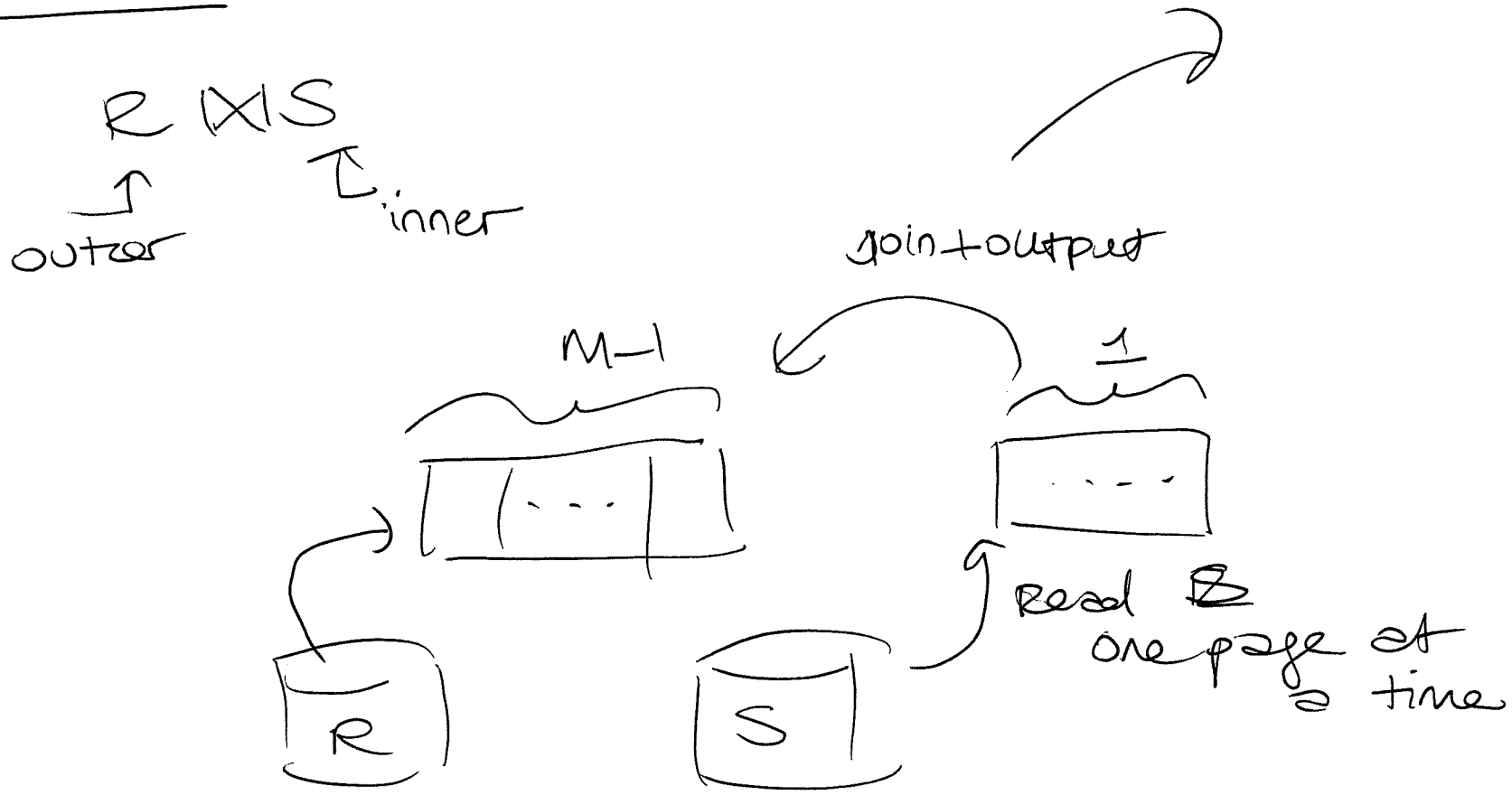
20,000 (R/W)
20,000 (R/W)
10,000 (R)

Total cost = 70,000

SORT R + GROUP BY →

(R(A) A

JOINS

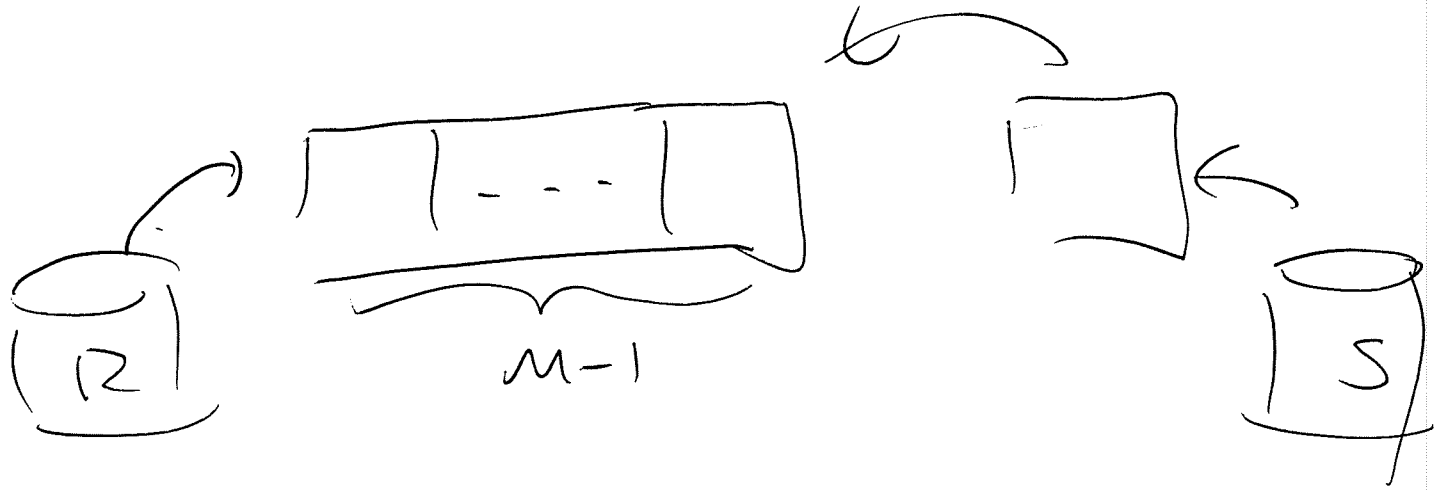


Suppose PAGES (R) $\leq M-1$

COST = PAGES (S) + PAGES (R)

BLOCK NESTED LOOP JOIN

PAGES(R) < M−1
PAGES(S) < M−1



(Read R, M−1 blocks (pages at a time) (Join with all of S)
repeat until all of R is read.

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

Block Nested Loop Join

PAGES(R) = 1,000

PAGES(S) = 2,000

1000/200

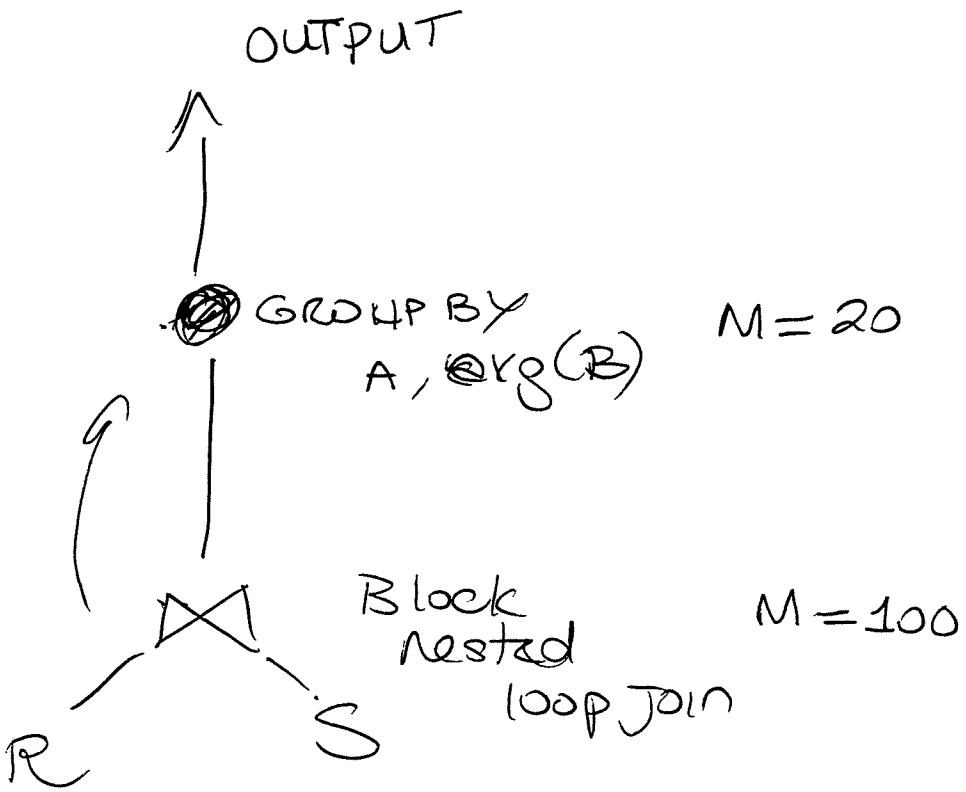
R ⋈ S M = 201 1,000 + 5 * 2,000 = 11K

S ⋈ R M = 201 2,000 + 10 * 1,000 = 12K

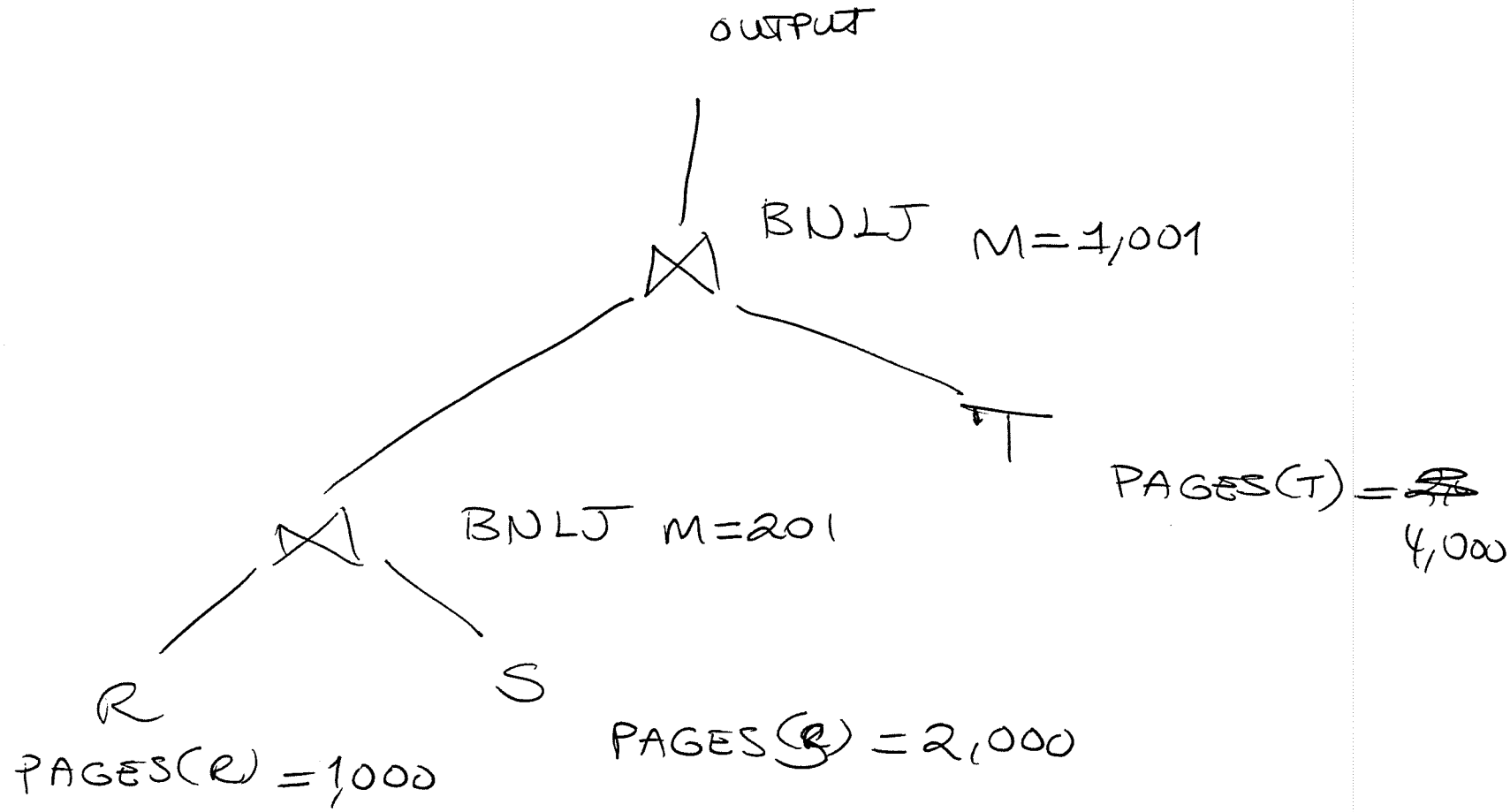
R ⋈ S M = 21 1000 + 50 * 2,000 = 101K

ITERATOR INTERFACE

open()
close()
getNext() ⟶ produce one block of output in the buffer


OUTPUT
GROUP BY
A, Avg(B)
M = 20
Block
Nested
loop Join
M = 100
R
S

PAGES (R ⋈ S) = 5,000



R ⋈ S) → join COST = 1,000 + 5 * 2000

= 11K

READ T → 5 TIMES → 20,000

TOTAL COST = 31K