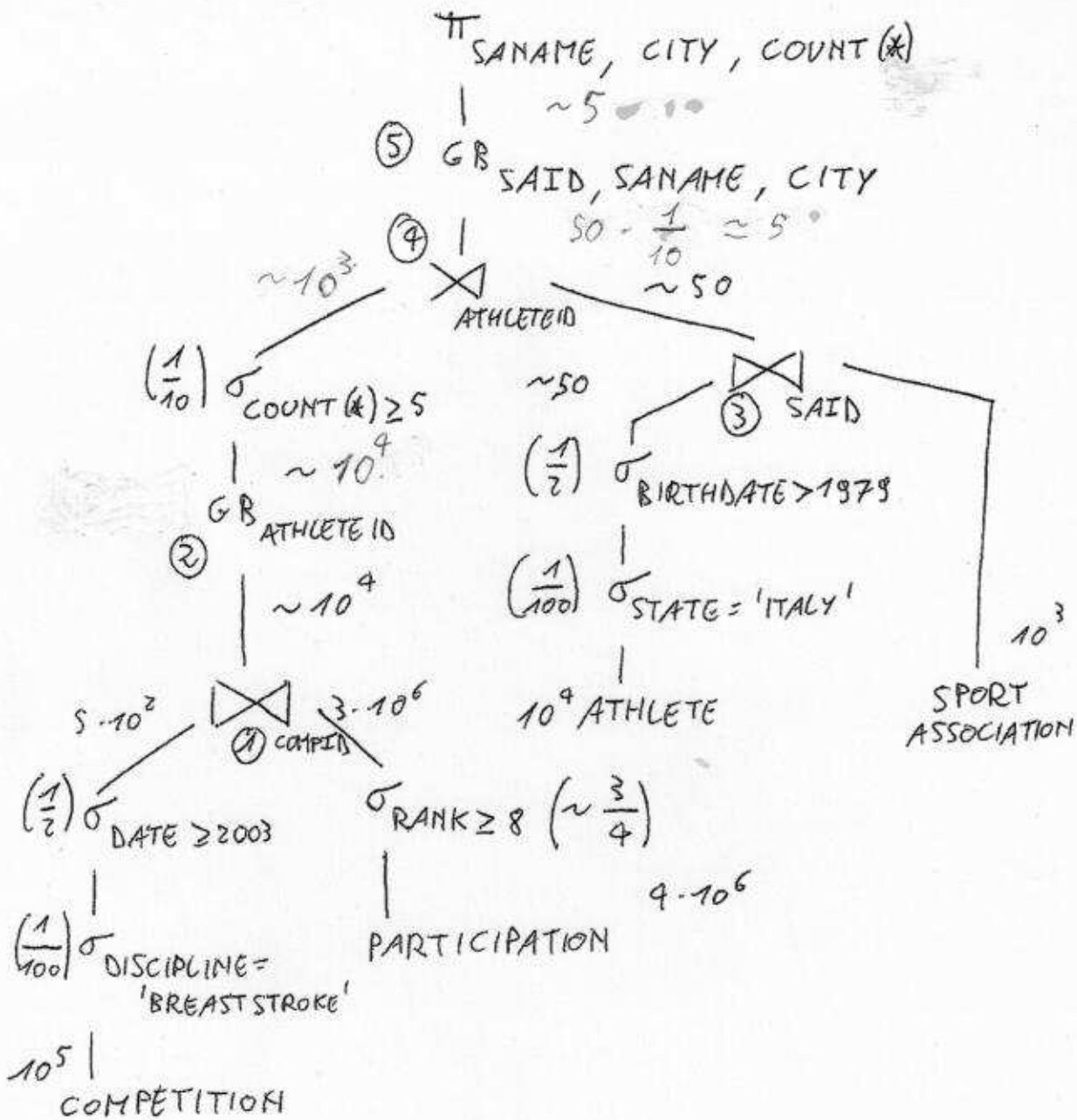


QUERY 1 - SPORT EXERCISE



ACCESS PATH

COMPETITION	:	TABLE ACCESS FULL + FILTER
PARTICIPATION	:	TABLE ACCESS FULL + FILTER
ATHLETE	:	TABLE ACCESS FULL + FILTER
SPORT ASSOCIATION	:	TABLE ACCESS FULL

① NL : INNER = COMPETITION \Rightarrow OK
OUTER = PARTICIPATION

HJ : NO IF I CONSIDER COMPETITION ($5 \cdot 10^2$)
SMALL AND DOES NOT HELP ②

MJ : NO BECAUSE I HAVE TO SORT THE
TABLES

② GB HASH

③ NL : INNER = ATHLETE \Rightarrow OK
OUTER = SPORT-ASSOCIATION

HJ : NO TABLES TOO SMALL ($< 10^3$)

MJ : YES BUT I NEED TO SORT THE
TABLES

④ NL : INNER = ③ \Rightarrow OK
OUTER = ②

HJ : NO TABLES TOO SMALL ($< 10^3$)

MJ : YES BUT I NEED TO SORT THE
TABLES

NO DOES NOT HELP ⑤

⑤ GB SORT

INDEX

COMPETITION : - SECONDARY HASH INDEX ON DISCIPLINE
⇒ YES BECAUSE SELECTIVITY IS HIGH
($\frac{1}{100}$)

- SECONDARY HASH ON (DISCIPLINE, DATE)
⇒ YES BECAUSE SELECTIVITY IS HIGH ($\frac{1}{200}$)
⇒ NO FOR MAINTENANCE COSTS

PARTICIPATION : - SECONDARY B⁺TREE ON RANK
⇒ NO SELECTIVITY TOO LOW ($\frac{3}{4}$)

ATHLETE : - SECONDARY HASH ON STATE
⇒ YES SELECTIVITY HIGH ($\frac{1}{100}$)
- SECONDARY HASH ON (STATE, BIRTHDATE)
⇒ YES SELECTIVITY HIGH ($\frac{1}{200}$)
⇒ NO FOR MAINTENANCE COSTS

SPORT_ASSOCIATION ; NO INDEX

GB ANTICIPATION

② IT IS NOT POSSIBLE BECAUSE JOIN ATTRIBUTE
AND GB ATTRIBUTE ARE TWO DISJOINT SETS

ACCESS PATH WITH INDEXES

COMPETITION : INDEX RANGE SCAN (ON DISCIPLINE)
+ ACCESS BY ROWID

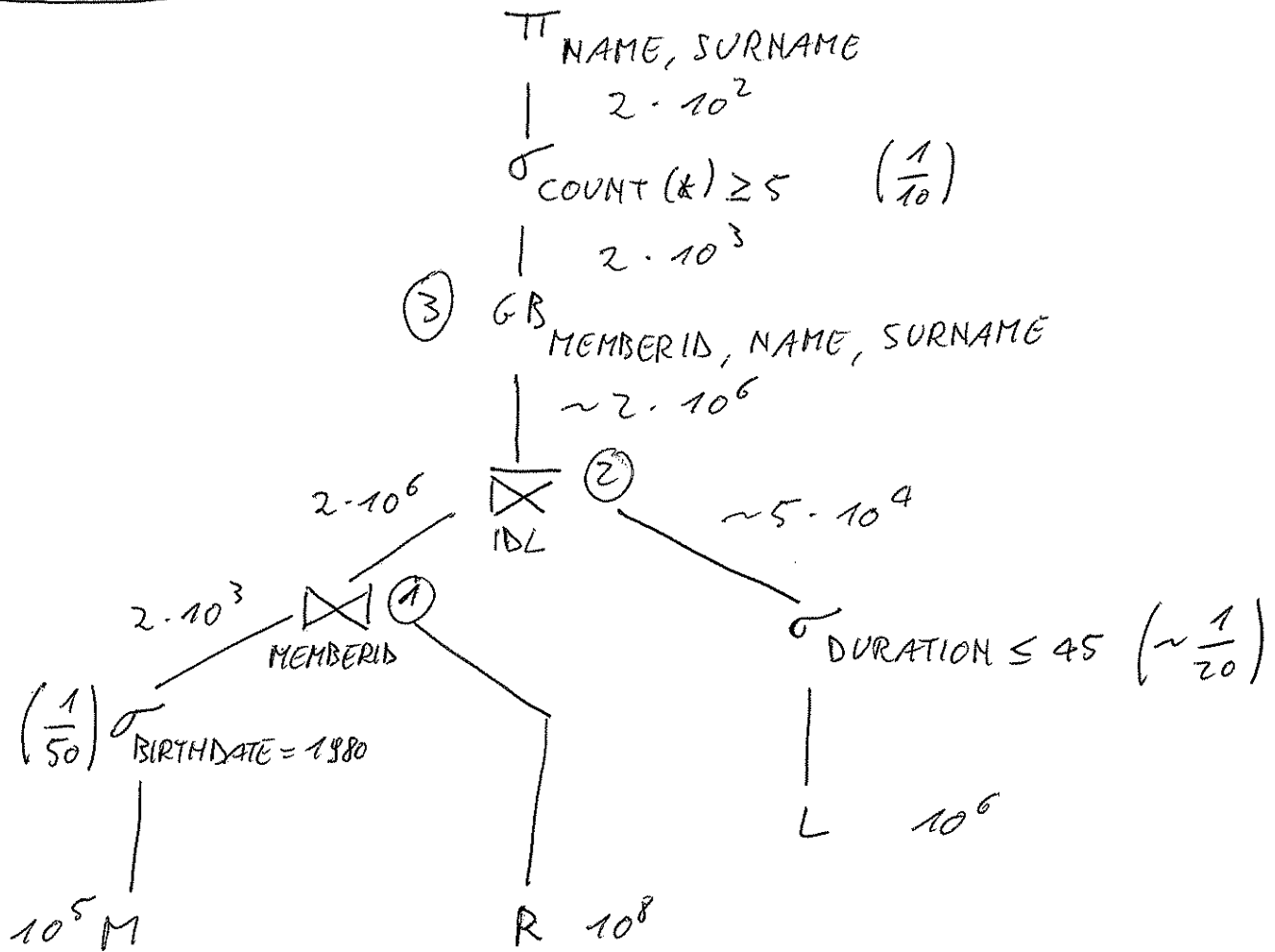
PARTICIPATION : TABLE ACCESS FULL + FILTER

ATHLETE : INDEX RANGE SCAN (ON STATE)
+ ACCESS BY ROWID

SPORT_ASSOCIATION : TABLE ACCESS FULL

EXERCISE 2 - QUERY 1

SOLUTION 1



ACCESS PATH

M : TABLE ACCESS FULL + FILTER
 R : TABLE ACCESS FULL
 L : TABLE ACCESS FULL + FILTER

JOINS + GB

- ① HT
- ② HT
- ③ GB HASH

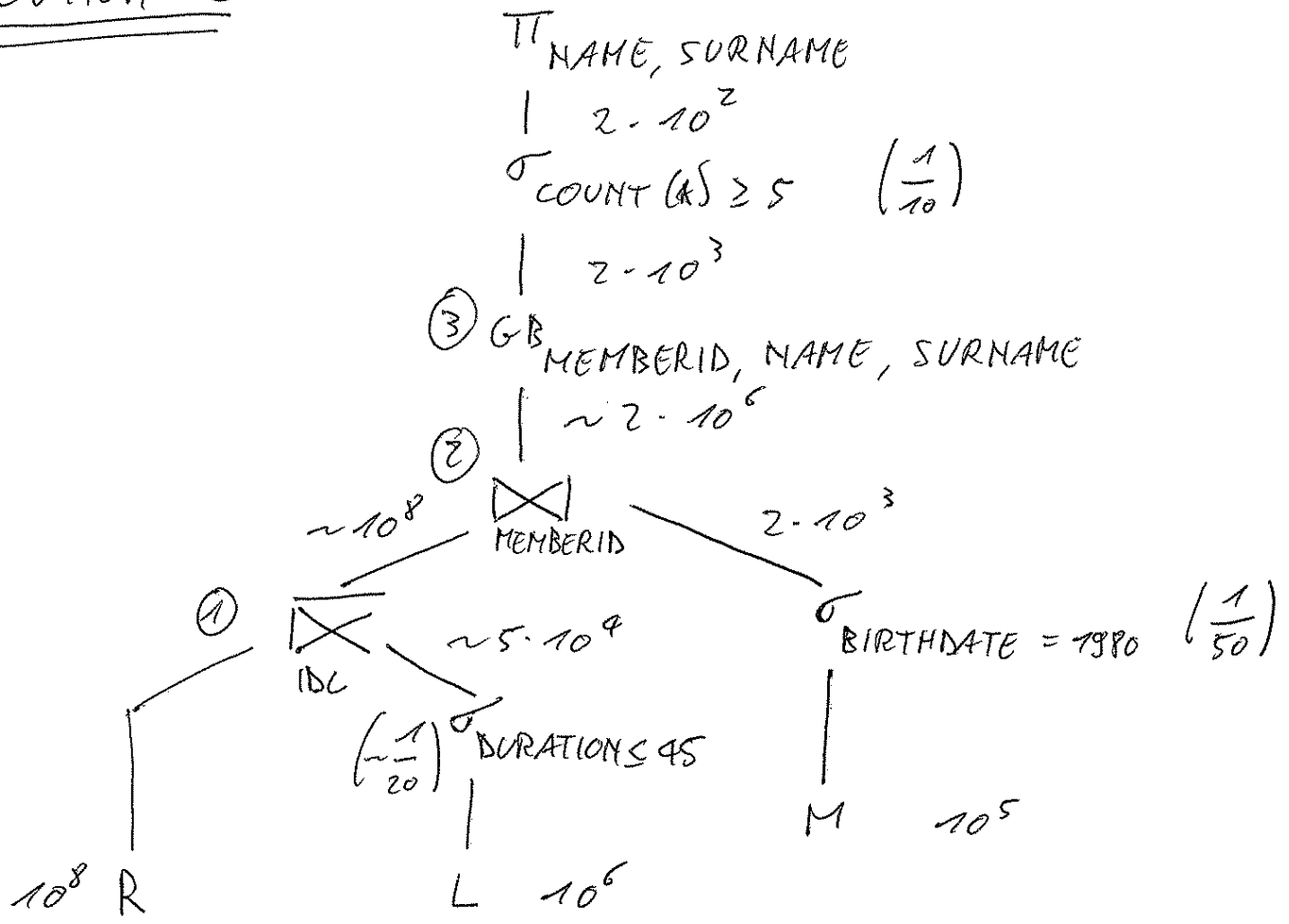
INDEXES

M : SECONDARY B⁺-TREE INDEX ON BIRTHDATE
 \Rightarrow INDEX RANGE SCAN + ACCESS BY ROWID

L : SECONDARY B⁺-TREE INDEX ON DURATION
 \Rightarrow INDEX RANGE SCAN + ACCESS BY ROWID

GB ANTICIPATION \rightarrow IT IS NOT POSSIBLE

SOLUTION 2



ACCESS PATH

- R : TABLE ACCESS FULL
- L : TABLE ACCESS FULL + FILTER
- M : TABLE ACCESS FULL + FILTER

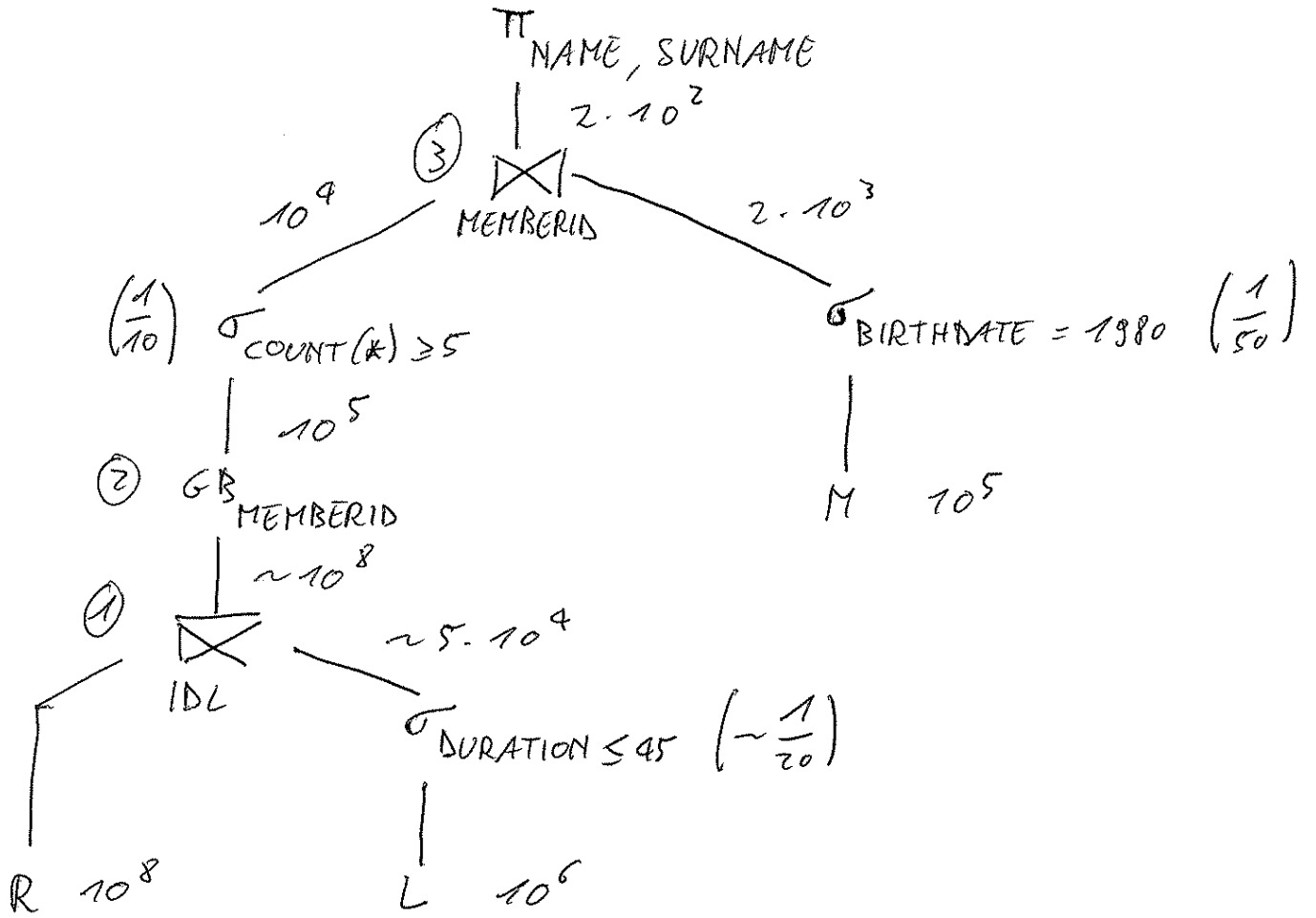
JOINS + GB

- ① HT
- ② HT
- ③ GB NO HASH

INDEXES

- M : SECONDARY B⁺-TREE INDEX ON BIRTHDATE
⇒ INDEX RANGE SCAN + ACCESS BY ROWID
- L : SECONDARY B⁺-TREE INDEX ON DURATION
⇒ INDEX RANGE SCAN + ACCESS BY ROWID

GB ANTICIPATION



JOINS + GB

- (1) HJ
- (2) GB HASH
- (3) HJ