WRITTEN ASSIGNMENT 2

Q1)

a) In this query, we are using only the age (in S.age >=30) and the grade (in the MIN(grade)). So, we can use the index (age, grade) for this query. As a result, it is possible to evaluate it with and index-only plan.

b) In this query, we are using the age (in the S.age >=30), the grade (in the MIN(grade)) and the gender (in the gender = 'Female') So, we cannot use only the index (age, grade) for this query. As a result, it is not possible to evaluate it with an index-only plan.

Q2)

a) In this query, the condition is an equality. So, the query will return only one record. We know that hash index is the best solution for equality search. As a result, using a hash index on attribut R.A has the least cost for this query.

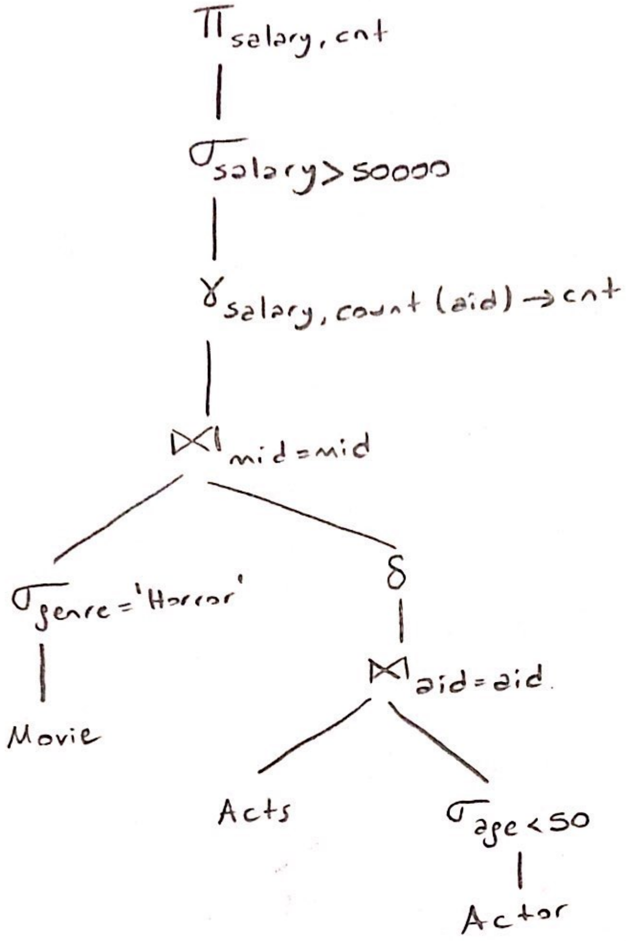
b) In this query, the condition is an comparison. So, the query will return the first 19999 records. We know that by using clustered B+ tree index on a conditional search we can faster and more accurate query results. As a result, using a clustered B+ tree index on attribute R.A has the least cost for this query.

c) In this query, the condition is an comparison again but the difference is the query will return only 9 records. As we know every page includes 10 records in it. So, we do not need to use clustered B+ tree index. As a result, using an unclustered B+ tree index on attribute R.A has the least cost for this query.

d) In this query, the condition is not important because the query will return all records except the 500000. So, we don't need to use any index methods In order to reduce the disk access cost, the query should reach the records on RAM. As a result, using a heap file storing relation R has the least cost for this query.

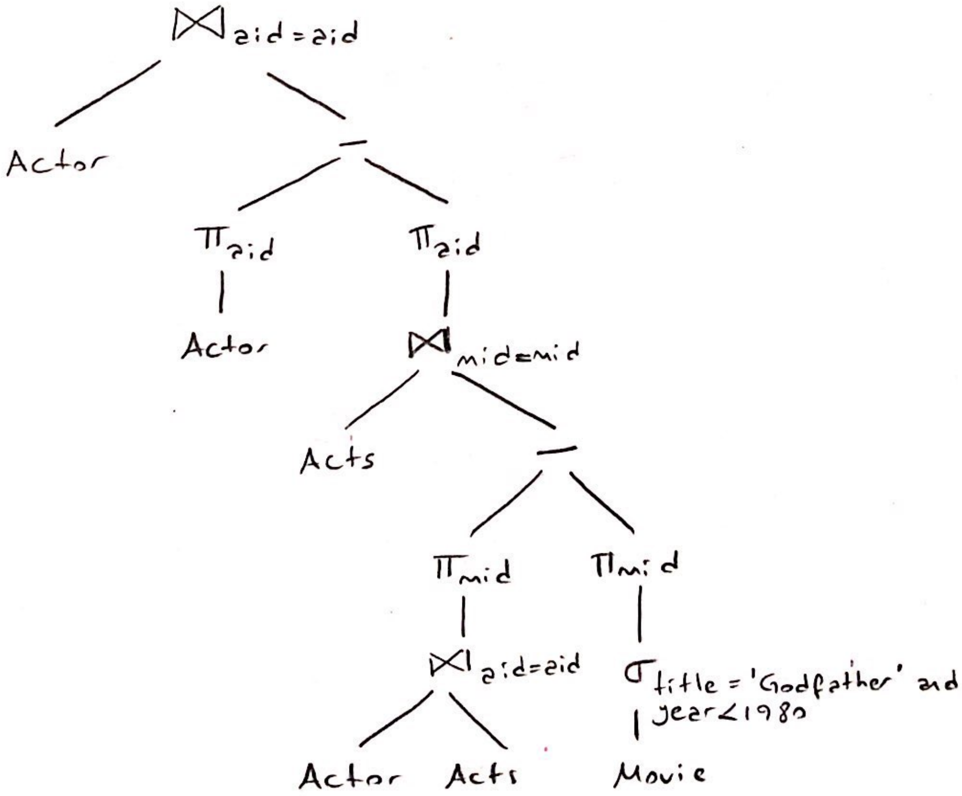
Q3)

a)



$\pi_{salary, cnt}$
$\sigma_{salary > 50000}$
$\gamma_{salary, count(aid) \rightarrow cnt}$
$\bowtie_{mid = mid}$
$\sigma_{genre = 'Horror'}$
Movie
δ
$\bowtie_{aid = aid}$
Acts
$\sigma_{age < 50}$
Actor

b)



$\bowtie_{aid = aid}$
Actor
$-$
π_{aid}
Actor
π_{aid}
$\bowtie_{mid = mid}$
Acts
$-$
π_{mid}
$\bowtie_{aid = aid}$
Actor
Acts
π_{mid}
$\sigma_{title = 'Godfather' \text{ and } year < 1980}$
Movie

Q4)

a) $cost = B(R) + B(R)B(S)/(M-2)$

$B(R) = \frac{20000}{10} = 2000$, $B(S) = \frac{5000}{10} = 500$, $M-2 = 42-2 = 40$

$Cost = 2000 + \frac{2000.500}{40} = 27000$

b) $cost = B(S) + B(S)B(R)/(M-2) = 500 + \frac{500.2000}{40} = 25500$

c) If $B(S)+B(R) <= M$, then we can use one-pass algorithm for sort-merge join. But $B(S)+B(R) = 2500$ and $M=42$. So, the condition $(2500<42)$ cannot be provided. So that, we should not use one-pass algorithm. If $B(R) <= M^2$ then we should use two-pass algorithm. But $B(R)=2000$ and $M^2 = 1764$. So, the condition $(2000<1764)$ cannot be provided. As a result, we should use the two-pass algorithm but, the memory is not enoug. So, the cost will be more than $3B(R) = 6000$.

In the process, it loads 42 pages in memory and sorts it. Then, merges $42-1=41$ runs into a new run. Total runs of length will be $M(M-1) \approx M^2$. For our join it will be more than that

d) If $B(R) <= M$, then we can use one-pass algorithm for hash join. But $B(R)=2000$ and $M=42$. So, the condition $(2000<42)$ cannot be provided. So that we should not use one-pass algorithm. If $\min(B(R), B(S)) <= M^2$, then we can use partitioned hash join. $\min(B(R), B(S)) = 500$ and $M^2 = 1764$. So, the condition $(500<1764)$ can be provided. As a result, we can use partitioned hash join.

In the process, read relation S one page at a time and hash into $M-1 = 41$ buckets. When a bucket fills up, it will be flushed into the disk. Finally, we get relation S back on disk split into 4 buckets. Then, same processes will be done for R by using a different hash function. Then, it will be scanned to find matching partition of S and probed the hash table.

So, total cost $= 3(B(R)+B(S)) = 3.(2000+500) = 7500$

e) If S has an index on the join attribute, it will be iterated over R, for each tuple fetch corresponding tuples from S

If index on S is clustered then;

$$\text{Cost} = B(R) + T(R)\, B(S) \,/\, V(s,b) = 2000 + \frac{20000 \cdot 500}{5000} = 4000$$

If index on S is unclustered then;

$$\text{Cost} = B(R) + T(R)\, T(S) \,/\, V(s,b) = 2000 + \frac{20000 \cdot 5000}{5000} = 22000$$

Q5)

a) $$\underbrace{\left(\frac{t\text{swim}}{N}\right)}_{\text{tswim rate on all tuples.}} \cdot (m10 + m11 + m12)$$

b) I assumed that, the count of a type is equal in every month. In this estimation the cost will be true. However, if the estimation is wrong and the count of a type is not equal in every month. The estimation may be incorrect.

$\boxed{14}$