DBMS Tutorial 30.01.2019

Topics

Relational Algebra

Projection π

The projection operator is used to produce from a relation R a new relation that has only some of R 's columns. The value of expression :

$$\pi_{A1,A2,\ldots,An}(R)$$

is a relation that has only the columns for attributes $A_1,A_2,\ldots,A_n$ of R.

The *schema for the resulting value is the set of attributes* $\{A_1,A_2,\ldots,A_n\}$.

UPDATED

Projection π

title	*year*	*length*	*genre*	*studioName*	*producerC#*
Star Wars	1977	124	sciFi	Fox	12345
Galaxy Quest	1999	104	comedy	DreamWorks	67890
Wayne's World	1992	95	comedy	Paramount	99999

$\pi_{title,year,length}(\text{Movies})$

title	*year*	*length*
Star Wars	1977	124
Galaxy Quest	1999	104
Wayne's World	1992	95

$\pi_{genre}(\text{Movies})$.

genre
sciFi
comedy

Duplicates eliminated, like in relational algebra.

$\pi_{genre}(\text{Movies})$.

genre
sciFi
comedy
comedy

Duplicates not eliminated, like a default projection in SQL.

Selection σ

The selection operator, applied to a relation R, produces a new relation with a subset of R 's tuples. The tuples in the resulting relation are those that satisfy some condition C that involves the attributes of R. We denote this operation as :

$$\sigma_C(R)$$

C can be : =, <, >, <=, >=

The *schema for the resulting relation is the same as R 's schema.*

Selection σ

title	*year*	*length*	*genre*	*studioName*	*producerC#*
Star Wars	1977	124	sciFi	Fox	12345
Galaxy Quest	1999	104	comedy	DreamWorks	67890
Wayne's World	1992	95	comedy	Paramount	99999

$\sigma_{length \geq 100}(\texttt{Movies})$

title	*year*	*length*	*genre*	*studioName*	*producerC#*
Star Wars	1977	124	sciFi	Fox	12345
Galaxy Quest	1999	104	comedy	DreamWorks	67890

Same schema therefore size of a single tuple is still the same.

First Name	Last Name	Address	City	Age
Mickey	Mouse	123 Fantasy Way	Anaheim	73
Bat	Man	321 Cavern Ave	Gotham	54
Wonder	Woman	987 Truth Way	Paradise	39
Donald	Duck	555 Quack Street	Mallard	65
Bugs	Bunny	567 Carrot Street	Rascal	58
Wiley	Coyote	999 Acme Way	Canyon	61
Cat	Woman	234 Purrfect Street	Hairball	32
Tweety	Bird	543	Itotltaw	28

Selection

First Name	Last Name	Address	City	Age
Mickey	Mouse	123 Fantasy Way	Anaheim	73
Bat	Man	321 Cavern Ave	Gotham	54
Wonder	Woman	987 Truth Way	Paradise	39
Donald	Duck	555 Quack Street	Mallard	65
Bugs	Bunny	567 Carrot Street	Rascal	58
Wiley	Coyote	999 Acme Way	Canyon	61
Cat	Woman	234 Purrfect Street	Hairball	32
Tweety	Bird	543	Itotltaw	28

Projection

Cartesian Product X

The Cartesian product of two sets R and S is the set of pairs that can be formed by choosing the first element of the pair to be any element of R and the second any element of S.

Since the members of R and S are tuples, usually consisting of more than one component (fields/attributes), the result of pairing a tuple from R with a tuple from S is a longer tuple, with one component for each of the components of the constituent tuples.

The relation schema for the resulting relation is the union of the schemas for R and S. However, if R and S should happen to have some attributes in common, then we need to invent new names for at least one of each pair of identical attributes. To disambiguate an attribute A that is in the schemas of both R and 5, we use R.A for the attribute from R and S.A for the attribute from S.

Cartesian Product X

A	B
1	2
3	4

(a) Relation R

B	C	D
2	5	6
4	7	8
9	10	11

(b) Relation S

A	$R.B$	$S.B$	C	D
1	2	2	5	6
1	2	4	7	8
1	2	9	10	11
3	4	2	5	6
3	4	4	7	8
3	4	9	10	11

(c) Result $R \times S$

Joins ⋈

Pairing only those tuples that match in some way. The simplest sort of match is the natural join of two relations R and 5, denoted R⋈S, in which we pair only those tuples from R and S that agree in whatever attributes are common to the schemas of R and S.

The relation schema for the resulting relation is the union of the schemas for R and S.

Joins $\bowtie$

A	B
1	2
3	4

(a) Relation R

B	C	D
2	5	6
4	7	8
9	10	11

(b) Relation S

A	B	C	D
1	2	5	6
3	4	7	8

R⋈S

The third tuple of S doesn't match with any tuple of R.

Resultant schema is the Union of the schemas of R and S.
Size of resultant tuples will be = S(R) + S(S) - S(Common)
Here = S(R) + S(S) - S(B) Since B is the common attribute it will appear only once in the result (unlike cartersian products).

Joins $\bowtie$

A	B	C
1	2	3
6	7	8
9	7	8

(a) Relation U

B	C	D
2	3	4
2	3	5
7	8	10

(b) Relation V

A	B	C	D
1	2	3	4
1	2	3	5
6	7	8	10
9	7	8	10

(c) Result $U \bowtie V$

First tuple of U has two matching tuples in V.

Third tuple of B has two matching tuples in U.

Set Operations: ∩, U, -

Intersection ∩

R ∩ S = is the set of tuples that are in both R and S.

Union U

R **U** S = the set of tuples that are in R or S or both. A tuple/element appears only once in the union even if it is present in both R and S.

Minus -

R-S = all the tuples in R that are not in S, ie we remove all common tuples from R.

R and S are two separate relations. For these operations to be possible R and S must have schemas with identical sets of attributes, and the types (domains) for each attribute must be the same in R and S.

R U S

name	*address*	*gender*	*birthdate*
Carrie Fisher	123 Maple St., Hollywood	F	9/9/99
Mark Hamill	456 Oak Rd., Brentwood	M	8/8/88

Relation R

name	*address*	*gender*	*birthdate*
Carrie Fisher	123 Maple St., Hollywood	F	9/9/99
Harrison Ford	789 Palm Dr., Beverly Hills	M	7/7/77

Relation S

name	*address*	*gender*	*birthdate*
Carrie Fisher	123 Maple St., Hollywood	F	9/9/99
Mark Hamill	456 Oak Rd., Brentwood	M	8/8/88
Harrison Ford	789 Palm Dr., Beverly Hills	M	7/7/77

$R \cap S$

name	*address*	*gender*	*birthdate*
Carrie Fisher	123 Maple St., Hollywood	F	9/9/99
Mark Hamill	456 Oak Rd., Brentwood	M	8/8/88

Relation R

name	*address*	*gender*	*birthdate*
Carrie Fisher	123 Maple St., Hollywood	F	9/9/99
Harrison Ford	789 Palm Dr., Beverly Hills	M	7/7/77

Relation S

name	*address*	*gender*	*birthdate*
Carrie Fisher	123 Maple St., Hollywood	F	9/9/99

R - S

name	*address*	*gender*	*birthdate*
Carrie Fisher	123 Maple St., Hollywood	F	9/9/99
Mark Hamill	456 Oak Rd., Brentwood	M	8/8/88

Relation R

name	*address*	*gender*	*birthdate*
Carrie Fisher	123 Maple St., Hollywood	F	9/9/99
Harrison Ford	789 Palm Dr., Beverly Hills	M	7/7/77

Relation S

name	*address*	*gender*	*birthdate*
Mark Hamill	456 Oak Rd., Brentwood	M	8/8/88

Examples

Is the following statement correct:

a) $R \cap S = R - (R - S)$

Examples

Is the following statement correct:

a) $R \cap S = R - (R-S)$

R-S = Everything that's only in R.

R-(R-S) = R - everything that is unique to R

= Everything that's common to both R and S

$= R \cap S$

Examples

Which is an equivalent statement:

```
SELECT ENAME
FROM   EMP,ASG
WHERE  EMP.ENO = ASG.ENO AND DUR > 37
```

Expression 1: $\Pi_{ENAME}(\sigma_{DUR>37 \wedge EMP.ENO=ASG.ENO}(EMP \times ASG))$

Expression 2: $\Pi_{ENAME}(EMP \bowtie_{ENO} (\sigma_{DUR>37}(ASG)))$

Examples

Which is an equivalent statement:

```
SELECT ENAME
FROM   EMP,ASG
WHERE  EMP.ENO = ASG.ENO AND DUR > 37
```

Expression 1: $\Pi_{ENAME}(\sigma_{DUR>37 \wedge EMP.ENO=ASG.ENO}(EMP \times ASG))$

Expression 2: $\Pi_{ENAME}(EMP \bowtie_{ENO} (\sigma_{DUR>37}(ASG)))$

Now that we have established both are same, which is better!

Examples

Which is an equivalent statement:

```
SELECT  ENAME
FROM    EMP,ASG
WHERE   EMP.ENO = ASG.ENO AND DUR > 37
```

Expression 1: $\Pi_{ENAME}(\sigma_{DUR>37 \wedge EMP.ENO=ASG.ENO}(EMP \times ASG))$

Expression 2: $\Pi_{ENAME}(EMP \bowtie_{ENO} (\sigma_{DUR>37}(ASG)))$

Now that we have established both are same, which is better!
Hint: Avoid Cartesian Product if you can!

Exercise 1

R(A, B) and *S*(B, C). Which two are equivalent. Give an example of how the different one can be different.

a. $\pi_{A,C}(R \bowtie \sigma_{B=1} S)$

b. $\pi_A(\sigma_{B=1} R) \times \pi_C(\sigma_{B=1} S)$

c. $\pi_{A,C}(\pi_A R \times \sigma_{B=1} S)$

Exercise 1 Ans ?

R(A, B) and S(B, C). Which two are equivalent. Give an example of how the different one can be different.

a. $\pi_{A,C}(R \bowtie \sigma_{B=1} S)$

b. $\pi_A(\sigma_{B=1} R) \times \pi_C(\sigma_{B=1} S)$

c. $\pi_{A,C}(\pi_A R \times \sigma_{B=1} S)$

Query (c) is different.
Let R = {(3, 4)} and S = {(1, 2)}.
Then query (a) and (b) produce an empty result while (c) produces {(3, 2)}.

Exercise 2

Consider a relation *R*(A, B) that contains r tuples, and a relation *S*(B, C) that contains s tuples; and r > 0 and s > 0.

In terms of R and S the minimum and maximum number of tuples that could be in the result of each expression:

a. $\pi_{A,C}(R \bowtie S)$

b. $\pi_B R - (\pi_B R - \pi_B S)$

c. $(R \bowtie R) \bowtie R$

d. $\sigma_{A>B} R \cup \sigma_{A<B} R$

Exercise 2 Ans

a. $\pi_{A,C}(R \bowtie S)$

b. $\pi_B R - (\pi_B R - \pi_B S)$

c. $(R \bowtie R) \bowtie R$

d. $\sigma_{A>B} R \cup \sigma_{A<B} R$

a). Minimum = 0 (if there are no shared B values), Maximum = r x s **(if both the relations have the same unique B value for all tuples)**

b) Minimum = 0 (if there are no shared B values), Maximum = min(r, s) (if one relation's B values are a subset of the other's, and all B values are distinct)

c) (equivalent to R) Minimum = r, Maximum = r

d) Minimum = 0 (if A = B in all tuples of *R*), Maximum = r (if A>B or A<B in all tuples of *R*)

Size Estimates

Statistics for relation R

- T(R) : # tuples in R
- S(R) : # of bytes in each R tuple
- B(R) : # of blocks to hold all R tuples
- V(R, A) : # distinct values in R for attribute A
- DOM (R, A): # possible distinct values for attribute A (size of domain for A)

W will represent the result relation.

Size Estimate : Π

NEW SLIDE

The answer to this will depend on whether we assume the projection to eliminate duplicates or not.

$\pi_A(R)$

1. $T(R)$ (assuming π won't eliminate duplicates)
2. $V(R,A)$ (assuming it will eliminate)

Note : A SQL query won't eliminate duplicates unless keyword 'distinct' is added.

Size Estimate : σ

1. $\sigma_{A=a}(R)$

$$W = \sigma_{Z=val}(R) \quad T(W) = \frac{T(R)}{V(R,Z)}$$

Assumption: Values in select expression Z = val are uniformly distributed over possible V(R,Z) values.

$$W = \sigma_{Z=val}(R) \quad T(W) = \frac{T(R)}{DOM(R,Z)}$$

Assumption: Values in select expression Z = val are uniformly distributed over possible DOM(R,Z) values.

Uniform = Every value equally likely.

2. $\sigma_{A\leq a}(R)$ or $\sigma_{A<a}(R)$

Solution # 1: T(W) = T(R)/2

- Assumption: All split values are equally likely

Solution # 2: T(W) = T(R)/3

- Assumption: Queries involving inequality ask more likely for a small fraction of possible tuples
- This assumption is usually preferred

Solution # 3: Estimate values in range

$T(W) = f \times T(R)$ f = (fraction of range)

Size Estimate : Inequality (>,<,<=,>=)

Based on the assumption that every tuple has a 50% chance of fulfilling the criterion.

Based on what's observed practically.

Use Sol#2, unless some 'extra info' given or specifically asked to do otherwise.

For Sol#3 you need that 'extra info', ie what fraction of tuples fulfil the selection criterion, like Exercise4 in this pdf.

2. $\sigma_{A \le a}(R)$ or $\sigma_{A<a}(R)$

Solution # 1: T(W) = T(R)/2

- Assumption: All split values are equally likely

Solution # 2: T(W) = T(R)/3

- Assumption: Queries involving inequality ask more likely for a small fraction of possible tuples
- This assumption is usually preferred

Solution # 3: Estimate values in range

$T(W) = f \times T(R)$ f = (fraction of range)

Exercise 3

Calculate the number of Tuples for 1 and 2.
$T(R)=100$, $V(R,A) =5$

1. $\sigma_{A=10}(R)$

2. $\sigma_{A\leq 10}(R)$

Exercise 3 Ans

Calculate the number of Tuples for 1 and 2 if $T(R)=100$, $V(R,A) = 5$

1. $\sigma_{A=10}(R)$

   $T(W) = 100/5$

2. $\sigma_{A \leq 10}(R)$

   $T(W) = 100/3$

Exercise 4

Calculate the number of Tuples for 1.
T(R)=100. Values of A are uniformly distributed over [1,20]

1. $\sigma_{A=10}(R)$

2. $\sigma_{A\leq 10}(R)$

3. $\sigma_{A>10}(R)$

Exercise 4 Ans

Calculate the number of Tuples for 1 and 2 if T(R)=100 and values of A are uniformly distributed over [1,20]

1. $\sigma_{A=10}(R)$

   T(W) = 100/20

Here we have extra info that A can have only 20 possible values, and that it is uniformly distributed over this domain, this means each value occurs 100/20 times.
Therefore half the tuples will have A≤10 and the other half will have A>10. Try solving for A≤5.

2. $\sigma_{A\leq10}(R)$

   T(W) = 100/2 (as 50% tuples will satisfy this condition)

3. Will be the same as 2.

Exercise 5

$R(A, B, C)$ is a relation. $T(R) = 10000$. $V(R, A) = 50$.

W = $\sigma_{A=10 \text{ AND } B<10}(R)$. What will be T(W)?

Exercise 5 Ans

$R(A, B, C)$ is a relation. $T(R) = 10000$. $V(R, A) = 50$.

W = $\sigma_{A=10 \text{ AND } B<10}(R)$. What will be T(W)?

T(W) $= T(R) \times \frac{1}{V(R, A)} \times \frac{1}{3} = 67.$

Size Estimate X and $\bowtie$

For two relations R and S with tuples T(R) and T(S).

- $T(R \times S) = T(R) \times T(S)$
- $T(R \bowtie S) =$
  1. T(R) x T(S) if R and S have no common attribute
  2. Otherwise = $$\frac{T(R)xT(S)}{\max\{V(R,A),V(S,A)\}}$$

Where A is the common attribute. If there are more than one common attributes we add a denominator entry like the one for A for the other attributes as well.

Special case: the common attribute has a single unique value and all the tuples from both the relations have that same unique value. That is, V(R,A) = V(S,A) =1, and R.A = S.A, the number of tuples resulting from the join will be T(R)xT(S). This we can get from the 2. formula anyway.

Justification for the number of tuples resulting from a Join

This slide and the next one are not important, but read if you are interested or want a better understanding!

$$\frac{T(R) \times T(S)}{\max\{V(R,A), V(S,A)\}}$$

This estimate is based on the assumption of "containment of value set" which states that the relation which has higher unique values of the **join attribute** contains all the values appearing in the other.

- $V(R,A) > V(S,A)$ then relation R has all values of A appearing in S (and obviously some more too). In other words $\boldsymbol{\pi_A(S) - \pi_A(R) = \text{empty}}$.
- $V(R,A) < V(S,A)$, then relation S has all values of A appearing in R (plus some more!) or $\boldsymbol{\pi_A(R) - \pi_A(S) = \text{empty}}$.

Assuming projection will eliminate duplicates!

Justification for the number of tuples resulting from a Join

- If V(R,A) ≥ V(S,A) and we apply the containment assumption,
  - Each unique value of A in R has T(R)/V(R,A) tuples. Since R contains all the values of A that relation S has, then **each one of the T(S) tuples will be matched with T(R)/V(R,A)** tuples from R ie total tuples after join -
  - $$= \frac{T(R)}{V(R,A)} * T(S)$$

- V(R,A) ≤ V(S,A)
  - Each unique value of A in S has T(S)/V(S,A) tuples. Since S contains all the values of A that relation R has, each of the T(R) tuples will be matched with T(S)/V(S,A) tuples from S, ie total tuples after join -
  - $$= \frac{T(S)}{V(S,A)} * T(R)$$

Preservation of value sets

For R(A,B) and S(B,C)

1. $W(A,B) = \sigma_{A=10}(R)$
   a. $V(W,A) = 1$
   b. $V(W,B) = V(R,B)$

After a selection, the values for attributes other than the one on which selection was performed retain their distinct values.

2. $W(A,B,C) = R \bowtie S$
   a. $V(W,A) = V(R,A)$
   b. $V(W,B) = \min\{V(R,B), V(S,B)\}$
   c. $V(W,B) = V(R,B)$

Similarly after a Join, attributes other than ones on which Join was performed retain their distinct values.

UPDATED

Exercise 6

For R(A,B), S(B,C) and T(D,E) we have the following specs:

T(R) = 1000, V(R,A) =1, V(R,B) = 30

T(S) = 200, V(S,B) =40, V(S,C) = 10

T(T) = 10000, V(T,D) =10, V(T,E) = 20

Estimate the number of Tuples for the results.

a) $R \bowtie S$
b) $R \bowtie \sigma_{C=10}(S)$
c) $R \bowtie \sigma_{B=5 \text{ AND } C=10}(S)$
d) $\sigma_{A=10}(R) \bowtie S$ Assume result of $\sigma_{A=10}(R)$ is not empty.
e) $\pi_A(R) \bowtie S$
    i) Assuming projection will not eliminate duplicates
    ii) Assuming projection will eliminate duplicates
f) $R \bowtie S \bowtie T$

UPDATED

Exercise 6 Ans

For R(A,B), S(B,C) and T(D,E) we have the following specs:

T(R) = 1000, V(R,A) =1, V(R,B) = 30

T(S) = 200, V(S,B) =40, V(S,C) = 10

T(T) = 10000, V(T,D) =10, V(T,E) = 20

Estimate the number of Tuples for the results.

a) $R \bowtie S$ (1000*200)/40

b) $R \bowtie \sigma_{C=10}(S)$ (1000*(200/10))/40 = 500 tuples

c) $R \bowtie \sigma_{B=5 \text{ AND } C=10}(S)$ (1000*(200/(10*40)))/30 = 50/3 = 17 tuples

V{R,B}

d) $\sigma_{A=10}(R) \bowtie S$ ((1000/1)*200)/40 = 5000

(since V(R,A)=1 → all tuples have A=10). -------

e) $\pi_A(R) \bowtie S$

  i) (1000)*200 = 200,000 tuples

  ii) 1x200

f) $R \bowtie S \bowtie T$ ((1000*200)/40)*10000 = 5*10^7 tuples

UPDATED

Exercise 7

For R(X,Y) we have the following specs:

T(R) = 20000, V(R,X) =500, V(R,Y) = 10

Assume Attribute X = 20bytes, Y = 30bytes. And there is a clustering index on attribute X.

With block size 4096 bytes and a block header of 96bytes.

a) What is B(R)?
   i) For Spanned
   ii) Unspanned
b) How many tuples will the following queries return? What will be the size of the result in bytes?
   i) $\sigma_{X=10}(R)$
   ii) $\pi_Y(R)$ (Assume projection will not eliminate duplicates.)
   iii) How many I/O would you roughly need to retrieve the records for (i)? (Assume index to be in memory.)

Exercise 7 Ans

a) What is B(R)?

i) For Spanned $= c\left(\dfrac{\text{File size}}{\text{Block size}}\right) = c\left(\dfrac{20000\text{x}50}{4000}\right)$ = 250 Blocks

ii) Unspanned

$$= c\left(\frac{\text{Total number of Records}}{\text{Records/Block}}\right) \text{ where Records/Block} = f\left(\frac{\text{Block Size}}{\text{Record Size}}\right)$$

$$= c\left(\frac{20000}{\underbrace{f(4000/50)}_{80}}\right)$$

= 280 Blocks

Number of Records/Block are floor(BlockSize/RecordSize) for unspanned only because we want to store only complete records,

For spanned this value is simply = BlockSize/RecordSize. Please note for spanned its neither floor nor ceiling because we don't mind placing part of a record too.

Exercise 7 Ans

4000 byte per block / 50 byte per records = 80 record/blocl

b) How many tuples will the following queries return? What will be the size of the result in bytes?

i) $\sigma_{X=10}(R)$

Number of tuples = 20000/500 = 40 tuples

Size=Number of tuples x Size of one tuple = 40x(20+30) = 2000 bytes

ii) $\pi_Y(R)$

Number of tuples = 20000

Size=Number of tuples x Size of one tuple = 20000x30 = 600.000

iii) How many I/O would you roughly need to retrieve the records for (i)?

Using the fact that the file has a clustering index on X (ie file will be sorted on X), and V(R,X) = 500, then each X value appears in 40 records (and there are 80 records/block), so we can reasonably assume we can find all the records with the selection criteria X=10 within one block, therefore in one IO.

UPDATED

Exercise 8

For R(A,B,C,D) with following specs:

T(R) = 1000, V(R,A) =1, V(R,B) = 30, V(R,C)=10, V(R,D)=20

Estimate the number of Tuples for the results and the V(W,X), W=result, X={A,B,C,D}. For parts c and d assume projection to eliminate duplicates.

a) $\sigma_{A=10}(R)$
b) $\sigma_{B=5 \text{ AND } C<10}(R)$
c) $\pi_{A}(R)$
d) $\pi_{A,B}(R)$

7, b-3:

With clustering the X=10 tuples are stored contiguously. 40 tuples fit in 1 block (80 per block), so you need to read ≈ 1 data block.
Index is in memory → no extra disk I/O for the index.
→ ~1 I/O (one block read).

Exercise 8 Ans

UPDATED

For R(A,B,C,D) with following specs:

T(R) = 1000, V(R,A) =1, V(R,B) = 30, V(R,C)=12, V(R,D)=20

Estimate the number of Tuples for the results and the V(W,X), W=result, X={A,B,C,D}. For parts c and d assume projection to eliminate duplicates.

a) $\sigma_{A=10}(R)$

T(W) =1000/1 , V(W,A) =1 , V(W,B) = V(R,B), V(W,C)=V(R,C), V(W,D) =V(R,D)

b) $\sigma_{B=5 \text{ AND } C<10}(R)$

T(W) =1000/(30*3) , V(W,A) =V(R,A) , V(W,B) = 1, V(W,C)=V(R,C)/3 = 12/3, V(W,D) =V(R,D)

c) $\pi_{A}(R)$

T(W) =1 , V(W,A) =1 , V(W,B) = 0, V(W,C)=0, V(W,D) =0

d) $\pi_{A,B}(R)$

T(W) =30 , V(W,A) =1 , V(W,B) = 30, V(W,C)=0, V(W,D) =0

The unique tuples will depend only on the unique values of B, since V(R,A)=1

Exercise 9

For S(A,B) with following specs:

T(S) = 40000, B(S) = 500

V(S,A) =4000, V(S,B) = 2000

Assume S has a primary dense index on A and secondary index on B.

How many IOs would you expect the following operations to take if the indices are in memory.

a) $\sigma_{A=10}(S)$
b) $\sigma_{B=5}(S)$

Exercise 9

For S(A,B) with following specs:

T(S) = 40000, B(S) = 500

V(S,A) =4000, V(S,B) = 2000

Assume S has a primary dense index on A and secondary index on B.

How many IOs would you expect the following operations to take if the indices are in memory.

Assuming W to be the result. Blocks per record for S will be 40000/500=80

a) $\sigma_{A=10}(S)$

T(W) = 40000/4000=10. Since the index is clustering we can expect to find all the values with A=10 in one block, therefore 1 IO (= B(S)/V(S,A))

b) $\sigma_{B=5}(S)$

T(W) = 40000/2000=20. Since the index is non-clustering we can expect 1 IO/per tuple, therefore 20 IOs.