

Bases de Datos

Práctico 5: Optimización de Consultas

Ejercicio 1: Suponga que hay un árbol B+ en (dept_name, building) en la tabla department. ¿Cuál es la mejor manera de manejar la siguiente selección?

$$\sigma_{(building < "Watson") \wedge (budget < 55000) \wedge (dept_name = "Music")}(department)$$

Ejercicio 2: Usando reglas de equivalencia demostrar las siguientes transformaciones:

1. $\sigma_{nombre="Optimización de Consultas"}(\sigma_{apellido="Selinger"}(profe \bowtie curso)) = \sigma_{apellido="Selinger"}(profe) \bowtie \sigma_{nombre="Optimización de Consultas"}(curso)$
2. Con telefono(legajo, numero),
 $\Pi_{legajo, numero}(\sigma_{nombre="Optimización de Consultas"}(curso \bowtie (profe \bowtie telefono))) = \Pi_{legajo}(\sigma_{nombre="Optimización de Consultas"}(curso)) \bowtie \Pi_{legajo, numero}(profe \bowtie telefono)$

Ejercicio 3: contestar las siguientes preguntas. Si la respuesta es falso, dar un contraejemplo:

1. Eliminación de duplicados puede ser empujada adentro de una proyección
2. Eliminación de duplicados puede ser empujada adentro de unión
3. Proyección puede ser empujada adentro de unión

Ejercicio 4: considerar la optimización heurística siguiente:

1. Realizar selección tempranamente
2. Realizar proyección tempranamente
3. Hacer selección más restrictiva
4. Hacer operaciones de reunión más restrictivas
5. Ciertas selecciones pueden ser combinadas con producto cartesiano para tornar las operaciones en una reunión (natural o selectiva)

Ver detalles de esta optimización heurística en filmina 23.

Aplicar esta optimización heurística a las siguientes consultas:

1. $\sigma_{instructor.ID = teaches.ID}(\sigma_{dept_name="Physics"}(instructor \times teaches))$
2. $\Pi_{name, course_id}(\sigma_{instructor.ID = teaches.ID}(\sigma_{dept_name="Physics"}(instructor \times teaches)))$
3. $\Pi_{name, title}(\sigma_{dept_name="Comp. Sci."}(instructor \bowtie teaches \bowtie course))$

Donde *name* es nombre de instructor, *title* es título del curso. Instructor tiene el atributo *dept_name*.

Para cada expresión de consulta transformarla paso a paso indicando heurística usada y para cada heurística usada indicar reglas de equivalencia usadas y en qué orden se aplicaron (mostrar aplicación de heurística paso a paso).

Ejercicio 5: Considerar las tablas con la siguiente información estadística:

$W(a, b)$	$X(b, c)$	$Y(c, d)$	$Z(d, e)$
$T(W) = 400$	$T(X) = 300$	$T(Y) = 200$	$T(Z) = 100$
$V(W, a) = 50$	$V(X, b) = 60$	$V(Y, c) = 50$	$V(Z, d) = 10$
$V(W, b) = 40$	$V(X, c) = 100$	$V(Y, d) = 20$	$V(Z, e) = 50$

$T(R)$ significa número de tuplas de la tabla R.

Dar las entradas de la tabla de programación dinámica que evalúa todos los órdenes posibles de natural join permitiendo:

1. Árboles de reunión profunda a la izquierda solamente
2. Todos los árboles.

¿Cuál es la mejor elección en cada caso?

Ejercicio 6: Sea la BD con las siguientes tablas:

persona(DNI, nombre, edad)

bibliotecario(DNI, antigüedad)

trabajaEn(DNI, nombreBib, rol)

Sea la siguiente consulta:

Π nombre, DNI, nombreBib (σ antigüedad > 5 and rol = 'bibliotecario' (Persona \bowtie bibliotecario \bowtie trabajaEn))

Aplicar paso a paso la optimización de consultas considerando el siguiente orden:

1. Optimización por costo usando programación dinámica para las reuniones naturales asumiendo la tabla de abajo.
2. Transformar la consulta obtenida en el paso anterior aplicando optimización heurística de las filminas.

Se cuenta con la siguiente información para las tablas de la BD:

persona	bibliotecario	trabajaEn
1000 tuplas	400 tuplas	1500 tuplas
$V(\text{persona}, \text{edad}) = 60$	$V(\text{bibliotecario}, \text{antigüedad}) = 40$	$V(\text{trabajaEn}, \text{rol}) = 5$
$V(\text{persona}, \text{nombre}) = 900$		$V(\text{trabajaEn}, \text{DNI}) = 750$
		$V(\text{trabajaEn}, \text{nombreBib}) = 200$

No se pide elegir algoritmos más adecuados para operadores de la consulta obtenida.