**Assignment 9**

Given:

B(R) = 1000 B(S)= 750 B(W) = 500 B(U) = 250

**Formula for cost = Cost (R ⋈ S) = 5\*(T(R) + T(S))**

Therefore, no of blocks in R is T(R) = 4000/4 = 1000.

Using this formula, we will be calculating the costs of each of these queries:

1. Cost of R S = 5B(R) + 5B(S) = 5(1000) + 5(750) = **8750**
2. Cost of R W = 5B(R) + 5B(W) = 5(1000) + 5(500) = **7500**
3. Cost of R U = 5B(R) + 5B(U) = 5(1000) + 5(250) = **6250**
4. Cost of S W = 5B(S) + 5B(W) = 5(750) + 5(500) = **6250**
5. Cost of S U = Cartesian Product
6. Cost of U W = 5B(U) + 5B(W) = 5(250) + 5(500) = **3750**

The size of a join R ⋈ S is given by:

**T(R ⋈ AS) = T(R) \* T(S) / max{V(R,A), V(S,A)}**

And

**T(R ⋈ A,BS) = T(R) \* T(S) / max(V(R,A), V(S,A))max(V(R,B),V(S,B))**

where T(R) is the number of tuples in relation R, T(S) is the number of tuples in relation S, V(R,A) is the number of distinct values of attribute A in relation R, and V(S,A) is the number of distinct values of attribute A in relation S.

**Let’s create the Table:**

|  |  |  |  |
| --- | --- | --- | --- |
| **Query** | **Size (Block)** | **Cost** | **Plan** |
| RS | 50 | 8750 | S ⋈ R |
| RW | 10000 | 7500 | W ⋈ R |
| RU | 10000 | 6250 | U ⋈ R |
| SW | 15000 | 6250 | W ⋈ S |
| SU | 750000 | Cartesian Product | U ⋈ S |
| WU | 5000 | 3750 | U ⋈ W |

**3 Table Combinations are as below:**

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **Plan type** | **Query** | **Size (Block)** | **Cost** | **Plan** |
| **2-way plan** | RS | 50 | 8750 | S ⋈ R |
| RW | 10000 | 7500 | W ⋈ R |
| RU | 10000 | 6250 | U ⋈ R |
| SW | 15000 | 6250 | W ⋈ S |
| SU | 750000 | Cartesian Product | U ⋈ S |
| WU | 5000 | 3750 | U ⋈ W |
| **3-way plan** | RSW |  |  | (RS)W |
| RSU |  |  | (RS)U |
| RWU |  |  | (RW)U |
| SWU |  |  | (SW)U |

**Cost Calculation:**

**1. Combination RSW**

**Join (RS)W – Since R, S, and W share a common B column, we consider the join of R and S over B:**

(RS)W = Expense of (R S) + Expense of ( (R S) W)

= 8750+ ( 5 ( Total(R S)/4) + 5 B(W))

= 8750 + 3500 = **11350**

**2.** **Combination RSU**

**Join (RS)U = Expense of (R S) + Expense of ( (R S) U)**

= 8750 + ( 5 ( Total(R S)/4) + 5 B(U))

= 8750 + (5( 200/4) + 5(250))

= 8750 + 250 + 1250 = **10250**

**3. Combination RWU**

**Join (UW)R = Expense of (UW) + Expense of( (UW) R)**

= 3750 + (5( 20000/4) + 5(1000)) = **33750**

**4. Combination SWU**

**Join (SU)W = Expense of (SU) + Expense of( (SU) W)**

= 5000+ (5( 300000/4) + 5(250)) = **381250**

**Determining the results of each query:**

**1. For RWS:**

(RS)W: This can be computed as [T(R S) \* T(W)] / max( V(R, B), V(S, B), V(W, B)), which results in (200 \* 2000) / 200, leading to **4000**.

**2. For RSU:**

(RS)U: The output is computed as [T(R S) \* T(U)] / max( V(R, A), V(U, A)), which equals (200 \* 1000) / 100, yielding **2000**.

**3. For RWU:**

(RW)U: This is computed as [(T(R W) \* T(U)) / max( V(R, A), V(U, A))] \* max( V(W, D), V(U, D)). Substituting values, which equals **8000**.

**4. For SWU:**

(SW)U: The output is computed as [T(S W) \* T(U)] / max( V(W, D), V(U, D)), which equals (60000 \* 1000) / 100, yielding **600000**.

**Filling the information in the table**

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **Plan type** | **Query** | **Output Size** | **Cost** | **Plan** |
| **2-way plans** | RS | 50 | **8750** | S ⋈ R |
| RW | 10000 | **7500** | W ⋈ R |
| RU | 10000 | **6250** | U ⋈ R |
| SW | 15000 | **6250** | W ⋈ S |
| SU | 750000 | Cartesian product | U ⋈ S |
| WU | 5000 | **3750** | U ⋈ W |
| **3-way plans** | RSW | 4000 | **11350** | W ⋈ (S ⋈ R) |
| RSU | 2000 | **10250** | U ⋈ (S ⋈ R) |
| RWU | 8000 | **33750** | R ⋈ (U ⋈ W) |
| SWU | 600000 | **32500** | S ⋈ (U ⋈ W) |

For a 4-way plan, we consider each of the 3-table query and consider the plan which has the lowest cost and join it with the 4th remaining table

Consider RSW, (RS)W plan as it has the lowest cost. Now we join this with the leftover table U So the join will be R,S,U,W

**Cost of (((RS)W)U)= 15250**

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **Plan type** | **Query** | **Output Size** | **Cost** | **Plan** |
| **2-way plans** | RS | 50 | **8750** | S ⋈ R |
| RW | 10000 | **7500** | W ⋈ R |
| RU | 10000 | **6250** | U ⋈ R |
| SW | 15000 | **6250** | W ⋈ S |
| SU | 750000 | Cartesian product | U ⋈ S |
| WU | 5000 | **3750** | U ⋈ W |
| **3-way plans** | RSW | 4000 | **11350** | W ⋈ (S ⋈ R) |
| RSU | 2000 | **10250** | U ⋈ (S ⋈ R) |
| RWU | 8000 | **33750** | R ⋈ (U ⋈ W) |
| SWU | 600000 | **32500** | S ⋈ (U ⋈ W) |
| **4-way plans** | RSWU | 40000 | **15250** | W ⋈ (U ⋈ (S ⋈ R)) |

Therefore, the final and the fastest join would be:

W**⋈ (**U**⋈ (**S**⋈**R))