1. {a, b, c} is a relation that contains the tuples a, b, and c. In the following cases the tuples have arity of 1. Calculate the following:

a.({1, 2, 3} ᴜ {5, 7, 11}) ∩ {2, 4, 6, 8, 10}

**Answer: 2**

b. ({1, 2, 3} ∩ {2, 4, 6, 8, 10}) ᴜ ({5, 7, 11} ∩ {2, 4, 6, 8, 10})

**Answer: 2**

2. A relation exists with 4 columns, named Column1, Column2, Column3, and Column4. Column1 is of type text. Column2, Column3, and Column4 are of type int:

a.Use relational algebra to fulfill the intent of the following SQL.

•**SELECT Column1, Column3 FROM MyTable WHERE Column2 = Column3**

**Answer:** πColumn1,Column3(σColumn2=Column3(MyTable))

b.Reverse the order of projection and selection in your algebraic formulation from item 2a. What is the result of the new algebraic expression?

**Answer:** πColumn2,Column3(σColumn1=Column3(MyTable))

3.πc1, c2(σϕ1(σϕ2(πc1, c2, c3, c5(R))))

Where

•ϕ1: C1 = C5;

•ϕ2: C5 = “Test”;

•R: MyTable;

a.Write a SQL statement that declares the intent of the algebraic notation

**Answer:**

select c1,c2 from (select c1,c2,c3,c5 from MyTable where c5 = 'Test') AA

where c1 = AA.c5

b.Simplify the algebraic statement. Simplification means minimize the number of parentheses and terms.25

**Answer:** πc1,c2(σ c1=c5(MyTable))

^c5=’Test’

4.SELECT \* FROM T1 JOIN T2 ON T1.C1 = T2.C1

a.Write out an equivalent in relational algebra using the join operator

**Answer:** T1 ⋈T1.C1=T2.C1 T2

b.Write out an equivalent in relational algebra without using the join operator

**Answer:** **σ**T1.c1=T2.c1**(T1XT2)**

5.πS.C1, R.C2(σϕ1(R) ⋈ϕ2 S)

where

•ϕ1 = (R.C2 = ‘A’)

•ϕ2 = (R.C1 = S.C2)

•Write out equivalent SQL and test this SQL using relations R and S that you create for this example. The relations R and S in RelationalAlgebraAndSQL.pdf and RelationalAlgebraAndSQL.sql don’t quite work because their column types do not match for this assignment.

**Answer:** SELECT S.C1, RR.C2 FROM (select \* from R where c2 = 'A') RR JOIN S ON RR.C1 = S.C2

6.Write a function in R that uses sqldf package to create a full outer join. The function is called OuterJoin and the function declaration is: ***OuterJoin <- function(tableA, tableB, keyA, keyB)***. tableA and tableB are data frames that will be joined. keyA and keyB are of type character and contain the names of the column that you join on. The function returns a data frame that is the outer join of the two input data frames. The SQL statement in sqldf is available on the web. In order to point out the difference between tables and sets, do not use the UNION keyword from sqldf, instead use rbind from {base}. rbind, by itself, doesn’t do exactly what UNION does and you may need more code.

Answer: Please see the attached R files in canvas.

1. Function : Assignment#5\_Function.R
2. Test Script : Assignment#5.R