

Section 5: Relational Calculus, SQL and Datalog - Solution

Relational calculus:

$$q = \left\{ x \left| \begin{array}{l} A(x) \wedge B(x) \wedge C(x) \\ \vee A(x) \wedge \neg B(x) \wedge \neg C(x) \\ \vee \neg A(x) \wedge B(x) \wedge \neg C(x) \\ \vee \neg A(x) \wedge \neg B(x) \wedge C(x) \end{array} \right. \right\}$$

$$C(x) = \left\{ x \left| \exists y T(x, y) \wedge A(y) \wedge B(y) \wedge \neg \exists z T(x, z) \wedge T(z, y) \wedge \neg A(z) \wedge \neg B(z) \right. \right\}$$

SQL query (executable in sqlite with version $\geq 3.8.3$):

```
with C(b) as (
  select T.h b from T,A,B
  where T.l = A.b and T.l = B.b
  and not exists
    (
      select * from
        (
          select distinct T1.h h,T2.l l
          from T T1, T T2, A,B
          where T1.l = T2.h and T1.l not in A and T1.l not in B
        ) as S1 where S1.h=T.h and S1.l=T.l
    )
)
select A.b x from A,B,C where A.b=B.b and A.b=C.b
union
select A.b x from A,B,C where A.b not in B and A.b not in C
union
select B.b x from A,B,C where B.b not in A and B.b not in C
union
select C.b x from A,B,C where C.b not in A and C.b not in B;
```

Datalog (using pyDatalog syntax, executable in pyDatalog):

```
#adjacent positions
D(H,L) <= T(H,L) & ~(T(H,Z) & T(Z,L))

#carray bit
C(H) <= D(H,L) & A(L) & B(L)
C(H) <= D(H,L) & C(L) & A(L)
C(H) <= D(H,L) & C(L) & B(L)

#result
R(X) <= A(X) & B(X) & C(X)
R(X) <= A(X) & ~B(X) & ~C(X)
R(X) <= B(X) & ~A(X) & ~C(X)
R(X) <= C(X) & ~A(X) & ~B(X)
```