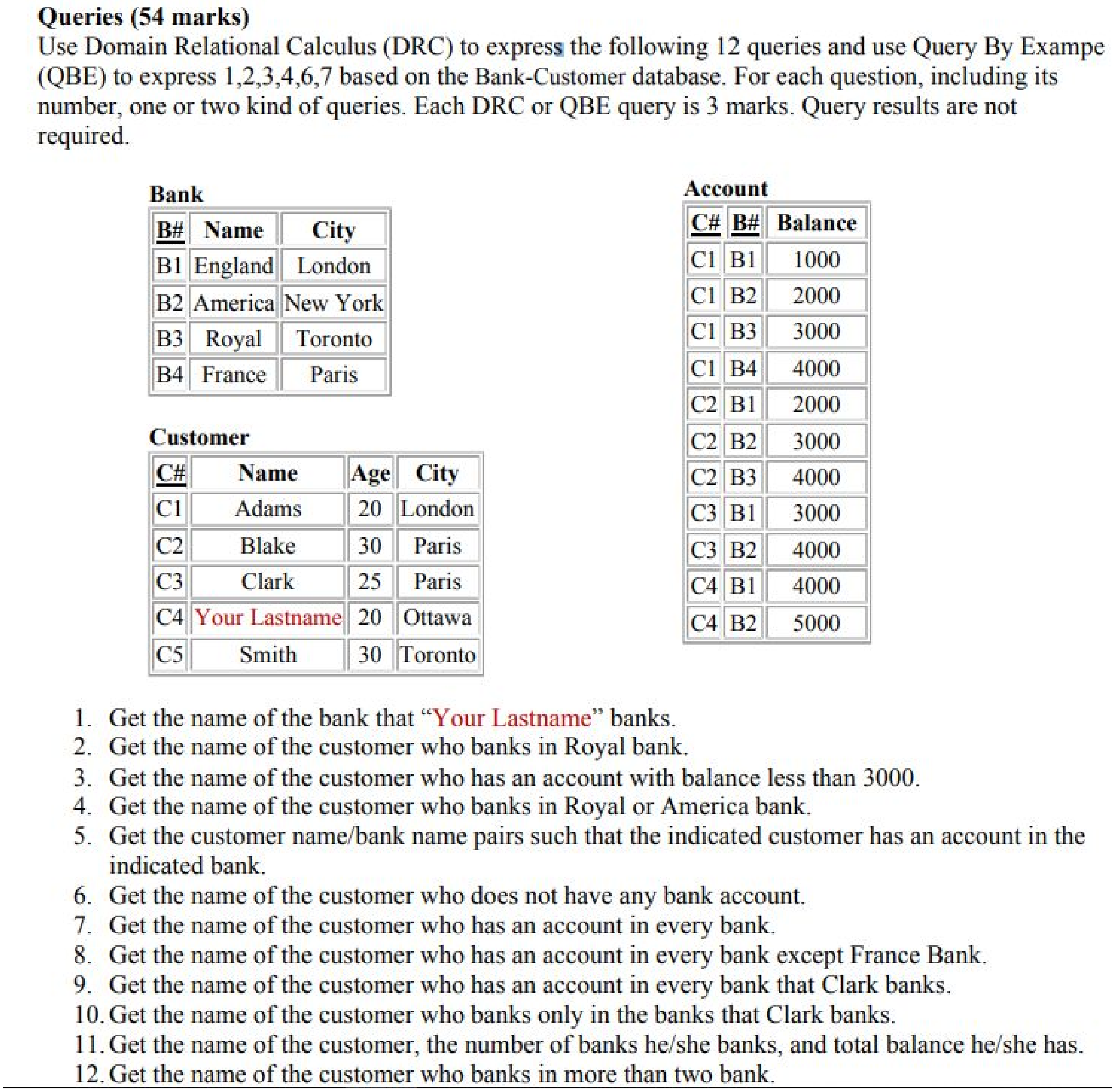
COMP 3005 A3
Krystian Wojcicki
101001444



Queries (54 marks)

Use Domain Relational Calculus (DRC) to express the following 12 queries and use Query By Exampe (QBE) to express 1,2,3,4,6,7 based on the Bank-Customer database. For each question, including its number, one or two kind of queries. Each DRC or QBE query is 3 marks. Query results are not required.

Bank

B#	Name	City
B1	England	London
B2	America	New York
B3	Royal	Toronto
B4	France	Paris

Customer

C#	Name	Age	City
C1	Adams	20	London
C2	Blake	30	Paris
C3	Clark	25	Paris
C4	Your Lastname	20	Ottawa
C5	Smith	30	Toronto

Account

C#	B#	Balance
C1	B1	1000
C1	B2	2000
C1	B3	3000
C1	B4	4000
C2	B1	2000
C2	B2	3000
C2	B3	4000
C3	B1	3000
C3	B2	4000
C4	B1	4000
C4	B2	5000

1. Get the name of the bank that “Your Lastname” banks.
2. Get the name of the customer who banks in Royal bank.
3. Get the name of the customer who has an account with balance less than 3000.
4. Get the name of the customer who banks in Royal or America bank.
5. Get the customer name/bank name pairs such that the indicated customer has an account in the indicated bank.
6. Get the name of the customer who does not have any bank account.
7. Get the name of the customer who has an account in every bank.
8. Get the name of the customer who has an account in every bank except France Bank.
9. Get the name of the customer who has an account in every bank that Clark banks.
10. Get the name of the customer who banks only in the banks that Clark banks.
11. Get the name of the customer, the number of banks he/she banks, and total balance he/she has.
12. Get the name of the customer who banks in more than two bank.

1) DRC> { BNAME | (exists CNAME, C, B) (Customer (C, CNAME, _, _) and Bank(B, BNAME, _) and Account(C, B,_) and CNAME=’Wojcicki’)};

Customer Account Bank

C#	Name	Age	City
_C	Wojcicki		

C#	B#	Balance
_C	_B	

B#	Name	City
_B	P.	

2) DRC> { CNAME | (exists C,B,) (Bank(B,’Royal’,_) and Account(C,B,_) and Customer(C, CNAME, _, _))};

Customer

C#	Name	Age	City
_C	P.		

Account

C#	B#	Balance
_C	_B	

Bank

B#	Name	City
_B	Royal	

3) DRC> { CNAME | (exists C, BALANCE)(Account(C, _, BALANCE) and Customer(C,CNAME,_,_) and BALANCE < 3000)};

Customer

C#	Name	Age	City
_C	P.		

Account

C#	B#	Balance
_C		< 3000

4) DRC> { CNAME | (exists B, C, BNAME) (Bank(B,BNAME,_) and Account(C,B,_) and Customer(C, CNAME, _, _) and (BNAME=’Royal’ or BNAME=‘America’)) };

Customer

C#	Name	Age	City
_C	P.		

Account

C#	B#	Balance
_C	_B	

Bank

B#	Name	City
_B	Royal	
_B	America	

```
5) DRC> { CNAME, BNAME | (exists C,B) ( Bank(B, BNAME, _) and Account(C,B,_) and
Customer(C, CNAME, _,_))};
```

```
6) DRC> { CNAME | (exists C) (Customer(C,CNAME,_,_) and not Account(C,_,_) ) };
```

Customer

C#	Name	Age	City
_C	P.		

¬

Account

C#	B#	Balance
_C		

```
7) DRC> { CNAME | (exists C) (Customer(C,CNAME,_,_) and (forall B)( Bank(B,_,_) and
Account(C,B,_))) };
```

Customer

C#	Name	Age	City
_C	P.		

¬

Bank

B#	Name	City
_B		

¬

Account

C#	B#	Balance
_C	_B	

```
8) DRC> { CNAME | (exists C)( Customer(C,CNAME,_,_) and
(forall B, BNAME)
        (Bank(B,BNAME,_) and
        ( (BNAME = ‘France’ and not Account(C,B,_)) or
        (BNAME != ‘France’ and Account(C,B,_) )
        )
) };
```

```
9) DRC> { CNAME | (exists C1,C) (Customer(C1,CNAME,_,_) and CNAME != ‘Clark’ and
Customer(C,’Clark’,_,_) and
(forall B)
```

```
          (Bank(B,_,_) and
                    ((Account(C1,B,_) and Account(C,B,_))
                    or not Account(C,B,_) )
          )};
```

```
10) DRC> {   CNAME | (exists C1, C) (Customer(C1,CNAME,_,_) and CNAME != ‘Clark’ and
Customer(C, ‘Clark’,_,_) and (forall B)
( Bank(B,_,_) and
          ( (Account(C1,B,_) and Account(C,B,_))
           or ( not Account(C,B,_) and not Account(C1,B,_) ) )
 )    };
```

```
11) { CNAME, count(B), sum(BALANCE) | (exists C, B) (Customer(C, CNAME,_,_) and
Account(C,B,BALANCE))};
```

12)

```
DRC> T1(CNAME, COUNT) := { CNAME, count(*) | (exists C) (Customer(C,CNAME,_,_) and
Account(C,_,_)) };
 { CNAME | (exists C)( T1(CNAME,C) and C > 2  )  };
```