Q1:

1. [-3] Each incorrect answer
2. [-3] Each incorrect answer
3. [+2] Each correct answer
4. [+3] correct answer
5. [+3] correct answer
6. [+1] Each statement (true or false)
7. [+1] Each statement (true or false)
8. [+1] Each statement (true or false)

Q2: SQL

1. [-4] Wrong answer.
   [-2] Calculate sum of B or 1's in T with A= 3
   [-2] Compute 1 is 3 is in T, 0 if not

2. [-6] Wrong query
   [-1] Miss DISTINCT
   [-2] GROUP BY the wrong attribute (e.g., name (pk is passID, not the name))
   [-1] Group By passID (need the specify passenger.passID)
   [-1] Syntax Error
   [-4] Wrong groupby and having
   [-1] HAVING COUNT (DISTINCT (*))
   [-2] miss passenger.passid = r2.passid or additional r1.passid <> r2.passid
   [-1] select the wrong attribute (e.g., passid)

ALTERNATIVE ANSWERs:

```
SELECT name
FROM passenger, resvn
WHERE passenger.passID = resvn.passID
GROUP By passenger.passID
HAVING COUNT(DISTINCT date) = 2;
```

```
SELECT name
FROm Passenger
WHERE passID IN (
        SELECT R1.passID FROM Resvn AS R1, Resvn R2 where R1.passID = R2.passID
AND R1.date <> R2.date);
```

```
select name from passenger as P, resvn as r1, resvn as r2 where p.passID = r1.passId and
p.passID = r2.passID and r1.date <> r2.date group by name having count(*) = 2;
```

```
select distinct passenger.name from passenger, resvn r1, resvn r2 where passenger.passid =
r1.passid and passenger.passid = r2.passid and r1.date <> r2.date;
```

3. [-1] For each wrong answer

Q3
1.
[2] pushing down the selection predicate
[2] Build hash table on Course after selection and Student, and store in memory at the same time.
[2] Pipeline the tuples from Took; no materialization for the intermediate result

Remark: The size of Course after selection is 200, while the size of Student is 5K. Since 200 + 5K < 6K, we can hold them simultaneously in memory. Please refer to slide 20 in part 2 of Query Optimization from lecture note (pdf).

2.
[2] pushing down the selection predicate
[2] Nested-loop join for both join
[2] First join Course after selection and Took after selection, and materialize the result.

Remark: Student has no common attribute with Course, thus if we first join Student with Course after selection, the join size is 200*5K = 1000K. While if we join Course after selection and Took after selection, the join size if 0.01*200*6k=12K, which is much smaller than 1000K. Thus, we should first join Course after selection and Took after selection.

Q4
(1)
(a)
[1] insert 4 correctly
[3] insert 8 correctly
[1] insert 6 correctly
[1] misc, like format (if point deducted here, will not be deducted again for format in next section)
(b) [1] delete 8 correctly
[4] delete 7 correctly
[1] delete 6 correctly

(2)
(a) [-1] for each misplaced key, up to [-3]
[-0.5] for each wrong bit, up to [-1]
[-1] for other things like wrong arrow

(b) [-1] for each misplaced key, up to [-4]
[-1] for other things like wrong arrow

Q5
(1)
Correct answer for initial value: +2 points
Correct answer for final value: +2 points

(2)
+2 for each transaction (T4, T5, and T2)

-1 for having one wrong transaction included