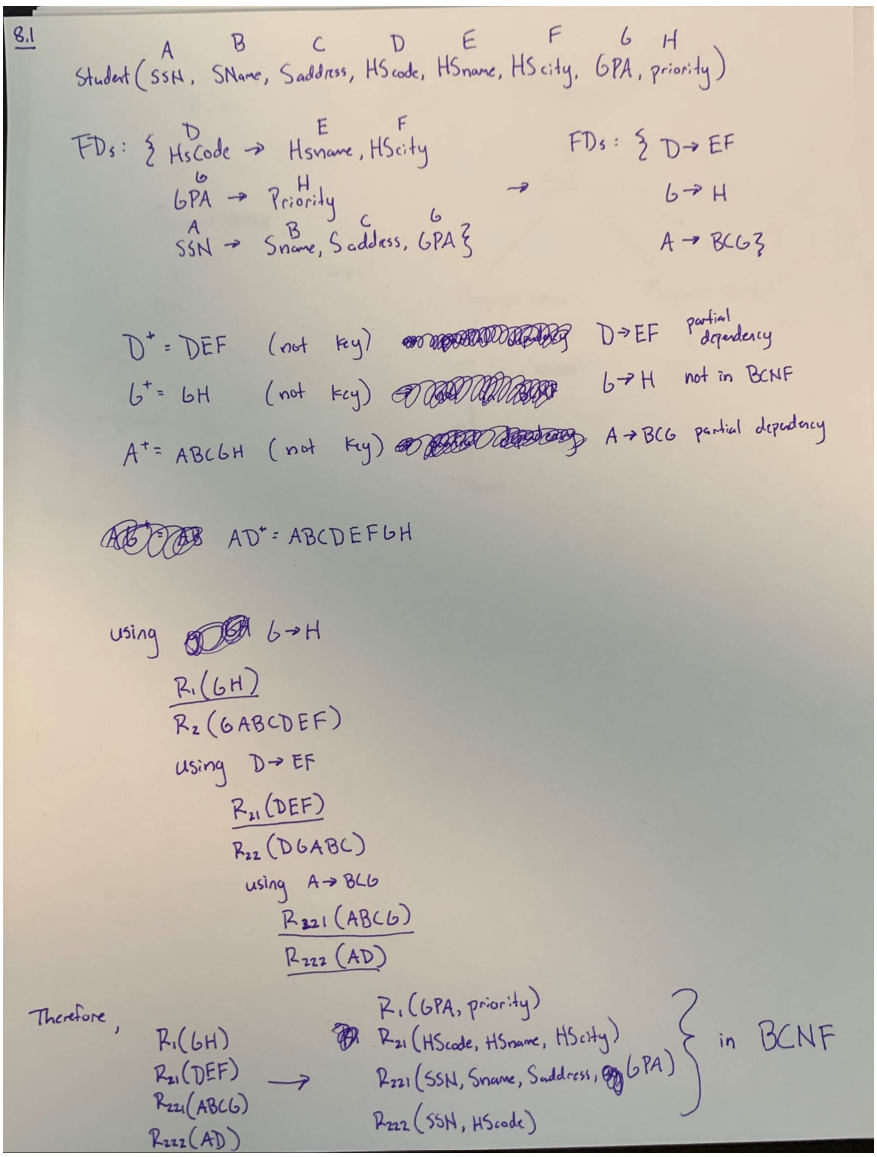
6.1

```
SELECT c1.sid, c2.sid
FROM catalog c1, catalog c2
WHERE c1.sid =c2.sid and c1.cost > c2.cost;
```

6.2

```
SELECT s.sid
FROM Suppliers s, Catalog c
WHERE exists (  select *
                from parts p
                where p.color='red' and s.sid=c.sid and c.pid=p.pid) or s.address='Ruston City';
```

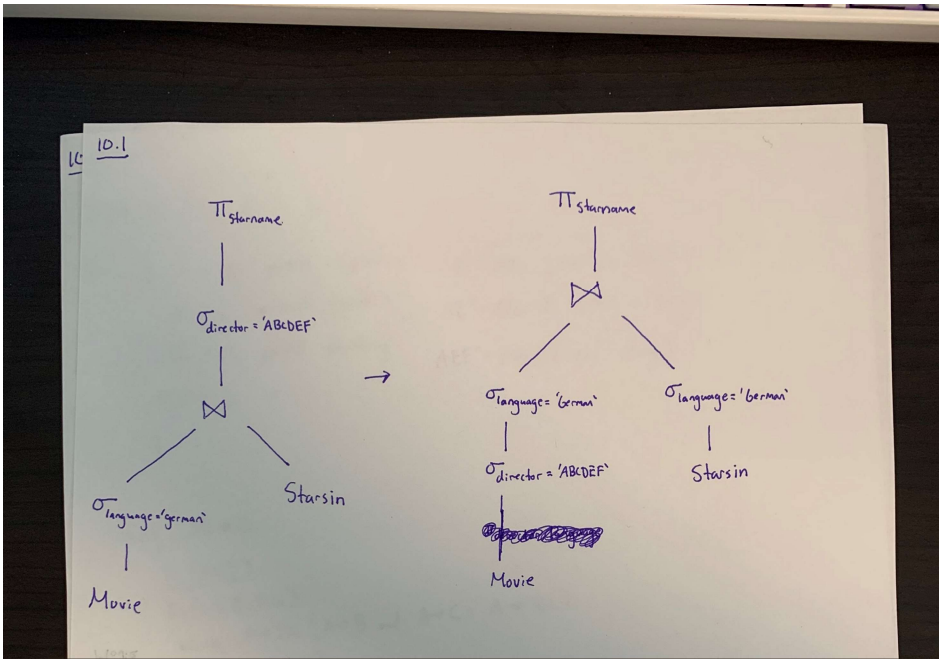
8.1



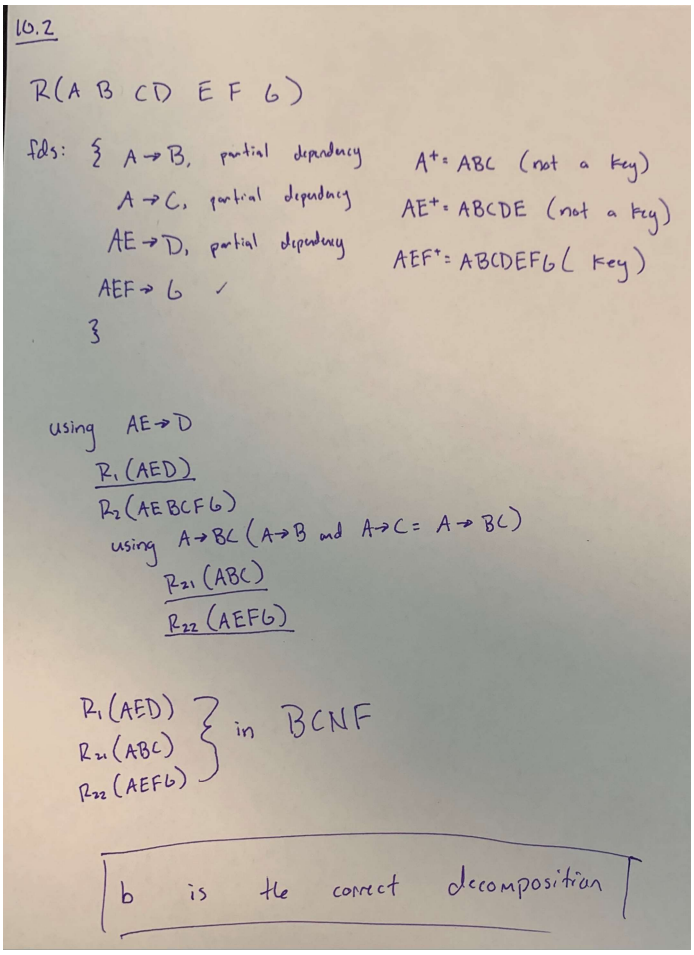
8.2

The two primary rules used when optimizing a parse tree are pushing selects and inserting projects. The purpose of inserting projects is to insert them near the leaves of the parse tree to reduce the size of tuples going up the tree. The point of pushing selects is to reduce the size of any given relation. There are also some exceptions for selects such as cascading selects, pushing selects within views, and pushing selects over joins.

10.1



10.2



10.3

Given the set difference binary operation R - S assuming S is the larger relation the strategy is to read S into memory completely and make it accessible through an in memory index structure. Then for each tuple of R search if it already exists in S and delete it from S if it exists. Lastly, output all remaining tuples.