Homework
Query Processing, part 2

The problems in this homework will use the following database (same as in the previous homework). Recall our abbreviations: the table names, `T`, `C`, `P`, `G` and `S` refer respectively to `Teams`, `Coaches`, `Players`, `Games` and `Stats`; `Position` is abbreviated as `Pos` and `HomeTeam`, `HomeTeamScore`, `Awa yTeam`, `AwayTeamScore` — as `HT`, `HTS`, `AT` and `ATS`.

```
CREATE TABLE Teams (
   Id     INT PRIMARY KEY,
   Name   CHAR(30),
   Coach  INT REFERENCES Coaches,
   Wins   INT,
   Losses INT,
   Place  INT
 );

CREATE TABLE Players (
   Id       INT PRIMARY KEY,
   Name     CHAR(30),
   Position CHAR(2),
   Height   INT,   /* in inches */
   Team     INT REFERENCES Teams
);


CREATE TABLE Stats (
  Player  INT REFERENCES Players,
  Game    INT REFERENCES Games,
  PTS     INT, /* points scored */
  AST     INT, /* assists */
  RB      INT, /* rebounds */
  BLK     INT, /* blocks */
  STL     INT, /* steals */
  TO      INT, /* turnovers */
  PF      INT, /* personal fouls */
  TF      INT, /* technical/flagrant fouls */
```

```
CREATE TABLE Coaches (
   Id    INT PRIMARY KEY,
   Team  INT REFERENCES Teams,
   Name  CHAR(30)
);

CREATE TABLE Games (
   Id            INT  PRIMARY KEY,
   HomeTeam      INT  REFERENCES Teams,
   HomeTeamScore INT,
   AwayTeam      INT  REFERENCES Teams,
   AwayTeamScore INT
 );
```

```
  PRIMARY KEY (Player, Game)
)
```

Consider the following table and column statistics:

Teams	Coaches	Players	Games	Stats
$T(T) = 32$	$T(C) = 32$	$T(P) = 512$	$T(G) = 1500$	$T(S) = 30,000$
$V(T, Id) = 32$ $V(T, Coach) = 32$ $V(T, Wins) = 90$ $V(T, Losses) = 90$ $V(T, Name) = 32$	$V(C, .) = 32$	$V(P, Pos) = 5$ $V(P, Height) = 20$ $V(P, Team) = 32$	$V(G, HT) = V(G, AT) = 32$ $V(G, HTS) = V(G, ATS) = 50$	$V(S, Player) = 512$ $V(S, Game) = 1500$ $V(S, PTS) = 50$ $V(S, AST) = V(S, RB) = 20$ $V(S, BLK) = V(S, STL) = 15$ $V(S, TO) = 10$ $V(S, PF) = 6$ $V(S, TF) = 3$

(note: these statistics roughly reflect the NBA 32-team, ∼12-players/team, 92-game season plus some play-off games. some other considerations: player height goes from 5'10" to 7'6" (20 values), game scores go from 70 to 120 - 50 different possibilities;)

Problem 1 *Estimate the sizes of the results in terms of the number of tuples of the following operations (expressed either in relational algebra or in SQL). For operations represented as SQL, also provide the appropriate relational algebra translation (expression or tree).*

Indicate, which specific estimation formulas you have used.

For joins, assume *value sets containment* and *preservation of value sets*, **unless otherwise noted**.

(note, $\rho_S(R)$ is a renaming operation, which is used in relational algebra expressions to disabmiguates self-joins.)

1. $\sigma_{Position='PG'}(P)$;
2. $\sigma_{HTS>90 \wedge ATS<80}(G)$;
3. $\sigma_{HT<>2}(G)$;
4. $\sigma_{PTS>15}(S) - \sigma_{AST=5}(S)$;
5. $\sigma_{PTS>15}(S) \cap \sigma_{AST=5}(S)$;
6. $\sigma_{PTS>15}(S) \cup \sigma_{AST=5}(S)$;
7. $\rho_{P1}(P) \bowtie_{P1.Pos=P2.Pos} \rho_{P2}(P)$;
8. $\rho_{S1}(S) \bowtie_{S1.AST>S2.RB} \rho_{S2}(S)$;
9. $\rho_{S1}(S) \bowtie_{S1.AST=S2.RB \wedge S1.RB=S2.AST} \rho_{S2}(S)$;
10. $sigma_{Game=150}(\rho_{S1}(S)) \bowtie_{S1.PF=S.PF} \sigma_{Player=150}(S)$;
11. $\gamma_{Pos,COUNT(*)}(P)$;

12. $\gamma_{Player,SUM(PTS)}(S)$;

13. $\gamma_{BLK,COUNT(*)}(S)$;

14. $\delta(\pi_{PTS,AST}(S))$;

15. $\delta(\pi_{AST,RB,BLK}(S))$;

16.
```
SELECT *
FROM Players
WHERE Position <> 'PG' AND Height > 78;
```

17.
```
SELECT *
FROM Players P, Teams T, Coaches C
WHERE P.Team = T.Id AND T.Coach = C.Id;
```

18.
```
SELECT *
FROM Players P, Stats S
WHERE P.Id = S.Player AND
      P.Position = 'PF' AND
      S.BLK = 10;
```

19. (pairs of players from different teams in the same game who scores same number of points. Note, that since you only need number of tuples, you can ignore the projection)

```
SELECT P1.Name, P2.Name
FROM  Stats S1, Stats S2, Players P1, Players P2
WHERE S1.Game = S2.Game AND
      P1.Id = S1.Player AND P2.Id = S2.Player AND
      P1.Team <> P2.Team AND
      S1.PTS = S2.PTS
```

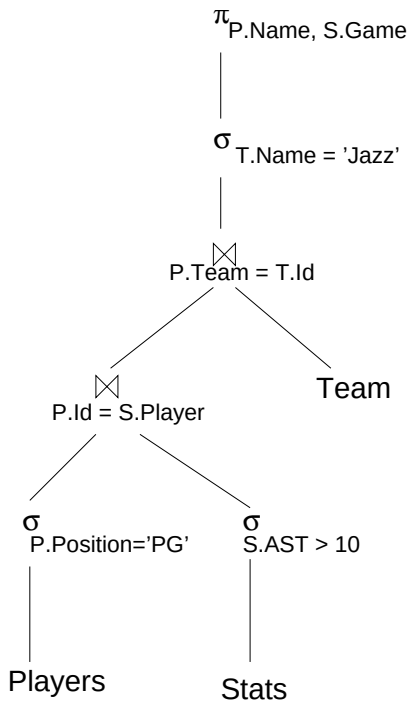
20.
```
SELECT P.Height, P.Position, COUNT(*), SUM(PTS)
FROM Players P, Stats S
WHERE P.Id = S.Player
GROUP BY P.Height, P.Position
HAVING COUNT(*) > 10;
```

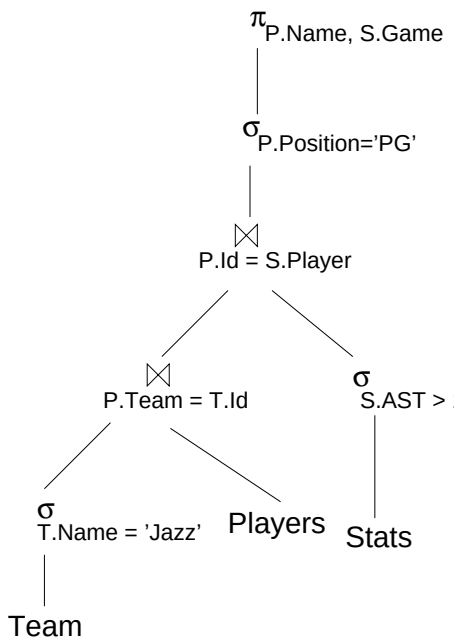
Problem 2 *For each query below, consider the two logical query plans shown. Determine which query plan is better. Show all work (i.e., all cost computations). Propose a* ***reasonable*** *physical query plan for the better logical query plan, compute estimated I/O cost for it.*[1]

Feel free to restrict your computations to the number of tuples (i.e., ignore the effects of projection).

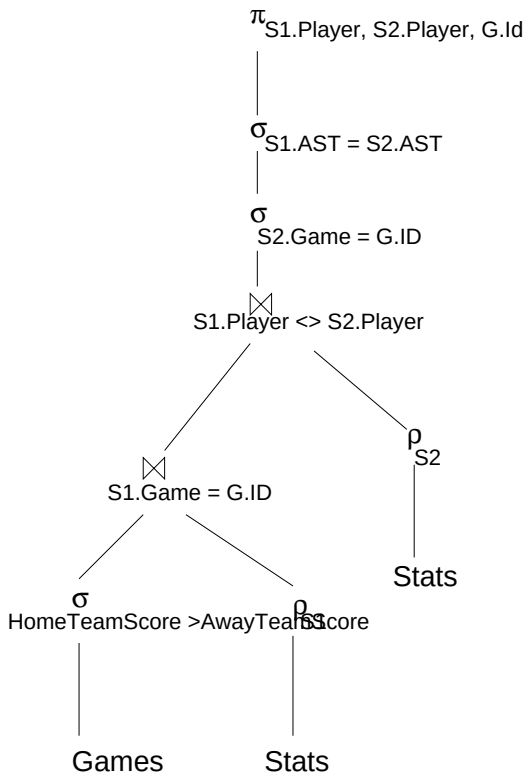
[1]This problem combines together a number of questions. In the final exam some of these questions may be split, to exclude the chance/effects of "carryover errors".

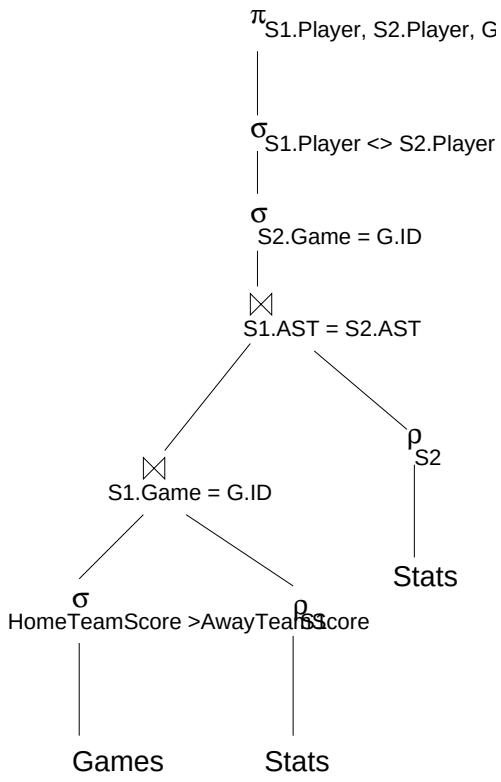
1.
```
SELECT P.Name, S.Game
FROM Players P, Stats S, Teams T
WHERE P.Id = S.Player AND P.Team = T.Id AND
      T.Name = 'Jazz' AND
      P.Position = 'PG' AND
      S.AST > 10;
```





2.
```
SELECT S1.Player, S2.Player, G.ID
FROM  Games G, Stats S1, Stats S2
WHERE S1.Game = G.Id AND
      S2.Game = G.Id AND
      S1.Player <> S2.Player AND
      S1.AST = S2.AST AND
      G.HomeTeamScore > G.AwayTeamScore;
```





3. (same query as in previous question)

```
SELECT S1.Player, S2.Player, G.ID
FROM  Games G, Stats S1, Stats S2
WHERE S1.Game = G.Id AND
      S2.Game = G.Id AND
      S1.Player <> S2.Player AND
      S1.AST = S2.AST AND
      G.HomeTeamScore > G.AwayTeamScore;
```