Student Worksheet

Result size estimation:

Result cardinality = Max # tuples * $\prod RF_i$

- col = value [example: $\boldsymbol{\sigma}_{r.bid = 100}$]
  - RF = $^1/_{\mathrm{NDistinct(T1)}}$
- col1 = col2 [example: $\boldsymbol{\sigma}_{r.bid = s.bid}$]
  - RF = $^1/_{\mathrm{Max(NDistinct(T1), NDistinct(T2))}}$
- col > value [example: $\boldsymbol{\sigma}_{s.rating > 5}$]
  - RF = $\frac{High(T1) - value}{High(T1) - Low(T1)}$

Cost estimation (# I/O) for single-relation plans:

- Sequential scan of file: Cost = NPages(R)
- B+ tree index I on key for equality search:
  Cost = Height(I) + 1
- Clustered index I for multiple select predicates:
  Cost = (NPages(I)+N**Pages**(R)) * $\prod RF_{matching}$
- Non-clustered index I matching one or more selects: Cost = (NPages(I)+N**Tuples**(R)) * $\prod RF_{matching}$

Tables:

Kitties: (kid [int], cuteness [1-10], owner [10 distinct]): 100 pages, 400 tuples

Puppies (pid [int], yappiness [1-10], owner [5 distinct]): 50 pages, 200 tuples

Humans: (hid [int], age [1-100]): 1,000 pages, 50,000 tuples

Indexes:

1. B+ tree (unclustered) on Kitties.cuteness [5 pages]
2. B+ tree (unclustered) on Puppies.yappiness [5 pages]
3. B+ tree (clustered) on (Puppies.owner, Puppies.yappiness) [15 pages]
4. B+ tree (unclustered) on Humans.hid [20 pages]

Query:

SELECT * FROM Kitties K, Puppies P, Humans H
WHERE K.owner = P.owner AND P.owner = H.hid
AND P.yappiness = K.cuteness
AND H.hid < 1200 AND P.yappiness = 7;

Note: it was suggested in section today that we could use the fact that K.cuteness = P.yappiness = 7 in the query to use the Kitties.cuteness index. This would result in (5 + 400) * 1/10 IOs for the Kitties table. This isn't reflected in the answers below.

1. **What are the best single-table plans (i.e., Phase 1)?**
   Kitties: File scan, no appropriate B+ trees
   Puppies: We will pick the B+ tree on (yappiness) [File scan = 50, B+ tree = (5 + 200) * 1/10]
   Humans: We will pick the file scan [File scan = 1000, B+ tree = (20 + 50,000) * 1,200/50,000]

2. **List the pairs of tables the optimizer will consider for 2-way joins (i.e., Phase 2)?**
   Kitties[File scan] x Puppies
   Puppies[unclustered B+] x Kitties
   ~~Humans[File scan] x Kitties~~
   ~~Kittes[File scan] x Humans~~
   Puppies[unclustered B+] x Humans
   Humans[File scan] x Puppies

3. **Which plans will be avoided?**
   Kittes and Humans don't have a join predicate, so those plans will be avoided.

Let's consider a possible sub-join of this expression, which in SQL would be

```
SELECT * FROM Kitties K, Puppies P
WHERE K.owner = P.owner
    AND P.yappiness = K.cuteness
    AND P.yappiness = 7;
```

4. **What would be the IO cost of doing ~~Index~~ Page-oriented Nested Loops join using Puppies as the outer, with the optimal single table selection methods (see part 1)?**

   Index Nested Loop Join:
   Puppies: Index Scan: ~21 IOs to select puppies
   20*(5+400)*1/10 = 810 IO
   831 I/Os

   We assume we're getting 20 tuples back from the index lookup on puppies from above (based on the selectivity). For each of those tuples, we do an index lookup on all the kitties that have K.cuteness = P.yappiness, and access it with the equation above.

   Page-oriented Nested loop join:
   Puppies: Index Scan: ~21 IOs to select puppies
   5 * 100 = 500 IOs
   521 I/Os

   If we access kitties with a page-oriented nested loop join, we end up with 5 pages of puppies (20 puppies at 4/page) and 100 pages of kitties.

   Note: In this page-oriented nested-loop join, the formula is N*M, not M + (N*M) as before, because the tuples of the outer relation (i.e., the results of select P.yappiness > 7) arrive on-the-fly, you don't have to read them from the disks.

5. **Now with Kitties as the outer.**
   We do 100 IOs to select the kitties. We can now use the clustered B+ tree on Puppies to its fullest.
   A clustered lookup for (owner = val and yappiness = 7):
   Cost = (15 + 50)*1/10*1/10 = 1 IO.
   100 + 400*1 = 500 IOs

   This comes from our equation above, but is bogus because we need more than 1 IO to get through the B+Tree. But, it's just an estimate!