Reserves(*sid:* integer, *bid:* integer, *day:* dates, *rname:* string)

```
SELECT DISTINCT R.sid, R.bid
FROM    Reserves R
```

Reserves has 1000 pages, Tuple size: 40bytes

Main memory reserves 20 buffer pages.

How many I/O operations needed for the "Projection based on Sorting"

Step 1: Scan the entire file and create temporary relation with required attributes

Step 2: Sort the temporary relation

Step 3: Scan the sorted relation, compare the adjacent tuples, remove duplicates and return results to the user

Step 1: Scan the entire file and create temporary relation with required attributes

Size of the tuples in the temporary relation: ¼ of the original (i.e. 10 bytes).

1000 (read) + 250 (temp write) = 1250 I/Os

Step 2: Sort the temporary relation

Assume External Merge Sort:

a. Create sorted run: 250/20 = 13 runs
b. Number of pass = (ceil(log $_{(20\text{-}1)}$ 13) +1) pass
c. I/O = (ceil(log $_{(20\text{-}1)}$ 13) +1) pass * 2 r/w * 250 = 1000 I/Os

Step 3: Scan the sorted relation, compare the adjacent tuples, remove duplicates and return results to the user

Total: 1250+1000+250=2500 I/Os

Exercise 14.3 Consider processing the following SQL projection query:

SELECT DISTINCT E.title, E.ename FROM Executives E

You are given the following information:

Executives has attributes *ename*, *title*, *dname*, and *address*; all are string
fields of the same length.
The *ename* attribute is a candidate key.
The relation contains 10,000 pages.
There are 10 buffer pages.

1. How many sorted runs are produced in the first pass? What is the average length of
these runs? (Assume that memory is utilized well and any available optimization
to increase run size is used.) What is the I/O cost of this sorting pass?

Size of the temporary relation: 10,000/4 = 2500 Pages

Number of sorted runs: 2500/ 10 = 250 Sorted runs

Average number of pages per run = 10 Pages

I/Os = 2500 (read) + 2500 (write) = 5000 I/Os

2. How many additional merge passes are required to compute the final result of the
projection query? What is the I/O cost of these additional passes?

ceil(log $_{(10-1)}$ 250) passes

2 * ceil(log $_{(10-1)}$ 250) passes = I/Os

3. (a) Suppose that a clustered B+ tree index on *title* is available. Is this index
likely to offer a cheaper alternative to sorting? Would your answer change if
the index were unclustered?

Clustered B+ tree

10000 (original file read) + 2500 (B+ leaf nodes) = 12500 I/Os

Unclustered

(10000 * tuples per page) + 2500 (B+ leaf nodes)