SOLUTION

Query 1 – Enumerate & Cost Pass 1 and Enumerate Pass 2

```
SELECT   *
FROM     employee E, department D
WHERE    E.dept = D.dept AND D.budget > $10000
```

1) Pass 1 (Find cheapest and interesting access methods)
    a. Employee (10,000 employees)
        i. File Scan, 1,000 I/Os
        ii. Index Scan, no selection, 2 I/Os to traverse index + 99 I/Os to traverse leaves + 10,000 I/Os [interesting order]
    b. Department (4/5 RF, 40 matching departments)
        i. File Scan, 5 I/Os
        ii. Index Scan, no selection, 1 I/O for index + 50 I/Os [interesting order]
2) Pass 2 (two way joins)
   [for each, nested loops (or index nested loops), hash, and sort-merge must be considered]
    a. Employee (File Scan) join Department (File Scan) – 8,000 results
    b. Employee (File Scan) join Department (Index) – 8,000 results
    c. Employee (Index) join Department (File Scan) – 8,000 results
    d. Employee (Index) join Department (Index) – 8,000 results
    e. Department (File Scan) join Employee (File Scan) – 8,000 results
    f. Department (File Scan) join Employee (Index) – 8,000 results
    g. Department (Index) join Employee (File Scan) – 8,000 results
    h. Department (Index) join Employee (Index) – 8,000 results

Query 2 – Enumerate & Cost Pass 1, Enumerate Pass 2

```
SELECT   *
FROM     employee E, department D, contracts C
WHERE    E.dept = D.dept AND D.area = C.area AND
         D.budget > $10000 AND C.state = CA AND C.rating > 5
```

1) Pass 1 (Find cheapest and interesting access methods)
    a. Employee (10,000 employees)
        i. File Scan, 1,000 I/Os
        ii. Index Scan, no selection, 2 I/Os to traverse index + 99 I/Os to traverse leaves + 10,000 I/Os [interesting order]
    b. Department (4/5 RF, 40 matching departments)
        i. File Scan, 5 I/Os
        ii. Index Scan, no selection, 1 I/O for index + 50 I/Os [interesting order]
    c. Contract (1/50 * 1/2 = 1/100 RF, 10,000 matching contracts)
        i. ~~File Scan, 10,000 I/Os~~
        ii. ~~Index Scan on C.company, no selection, 20 I/Os for index directory + 2000*5 I/Os to traverse buckets & overflow pages + 1,000,000 I/Os [non-interesting order]~~
        iii. Index Scan on C.state,rating, 3 I/Os to traverse index + 100 I/Os

2) Pass 2 (two way joins)
   [for each, nested loops (or index nested loops), hash, and sort-merge must be considered]
   a. Employee (File Scan) join Department (File Scan) – 8,000 results
   b. Employee (Index) join Department (File Scan) – 8,000 results
   c. Employee (File Scan) join Department (Index) – 8,000 results
   d. Employee (Index) join Department (Index) – 8,000 results
   e. ~~Employee (File Scan) join Contract – not considered, cross product~~
   f. ~~Employee (Index) join Contract – not considered, cross product~~
   g. Department (File Scan) join Employee (File Scan) – 8,000 results
   h. Department (File Scan) join Employee (Index) – 8,000 results
   i. Department (File Scan) join Contract – 8,000 results
   j. Department (Index) join Employee (File Scan) – 8,000 results
   k. Department (Index) join Employee (Index) – 8,000 results
   l. Department (Index) join Contract – 8,000 results
   m. ~~Contract join Employee (File Scan) – not considered, cross product~~
   n. ~~Contract join Employee (Index) – not considered, cross product~~
   o. Contract join Department (File Scan) – 8,000 results
   p. Contract join Department (Index) – 8,000 results