



Figure 15.1 Steps in query processing.

Given a query, there are generally a variety of methods for computing the answer. For example, we have seen that, in SQL, a query could be expressed in several different ways. Each SQL query can itself be translated into a relational-algebra expression in one of several ways. Furthermore, the relational-algebra representation of a query specifies only partially how to evaluate a query; there are usually several ways to evaluate relational-algebra expressions. As an illustration, consider the query:

```

select salary
from instructor
where salary < 75000;
  
```

This query can be translated into either of the following relational-algebra expressions:

- $\sigma_{\text{salary} < 75000} (\Pi_{\text{salary}} (\text{instructor}))$
- $\Pi_{\text{salary}} (\sigma_{\text{salary} < 75000} (\text{instructor}))$

Further, we can execute each relational-algebra operation by one of several different algorithms. For example, to implement the preceding selection, we can search every tuple in *instructor* to find tuples with salary less than 75000. If a B⁺-tree index is available on the attribute *salary*, we can use the index instead to locate the tuples.

To specify fully how to evaluate a query, we need not only to provide the relational-algebra expression, but also to annotate it with instructions specifying how to evaluate each operation. Annotations may state the algorithm to be used for a specific operation or the particular index or indices to use. A relational-algebra operation annotated