Data Structures in The Real World:
Databases & Expert Systems

- Databases are the lifeblood of E-Commerce & Medical Info Systems
  - **Examples:**
    - Given a key (credit card number) find the account record.
    - Given a patient ID, find the most recent X-rays.
  - **Problem:** **Mismatch** of disk access and main memory:
    - Main Memory access < 10 nanosec (10^{-8} sec)
    - Disk access < 10 millisec (10^{-2} sec)
  - **Solution**: Use **data structures** and algorithms to overcome mismatch:
    - **B-Tree** is a **balanced** multi-way *search* tree of order N: <= N children
    - Keys in left child are **less than** parent's key, keys in right child are **>=**
    - Within a node: keys are in *sorted* order ⇒ use binary search.
- Result: only 3 **disk** accesses to find any one of 100 million records:
  - 100 children per node; 4 level tree; 10^{+8} = 100 million.
  - All leaves are at same level (3 below root); keep root in memory.

Banks Merge: combine common accounts

- Combine accounts from Bank-A and Bank-B for matching Social Security Numbers:
  - Create two **Hash Table indices** of SS# and account info for each bank, using common hash function.
  - Merge hash tables and combine entries - **is database “Hash-Join”**
  - Also could do with one hash table.

Expert Systems and Decision Making

- Decision Trees and Game Trees - actually graphs.
- *Node* for each **decision** (or game position).
- *Branch*/edge for each **choice** (or game move).
  - Weight on edge is "*cost*" of that choice.
  - Value in leaf is profit/*benefit* of reaching that goal (win/loose).
- **Path** from root to a leaf is the **plan** or **solution**.
  - **Optimize** the cost or likelihood of success.
- Overall graph is the **'decision space'** or **'plan space'**.
- Another example:
  - Rule-based Expert Systems use a 'Rete net' **graph** to represent the interdependent rules, and to determine which rules can be executed when.