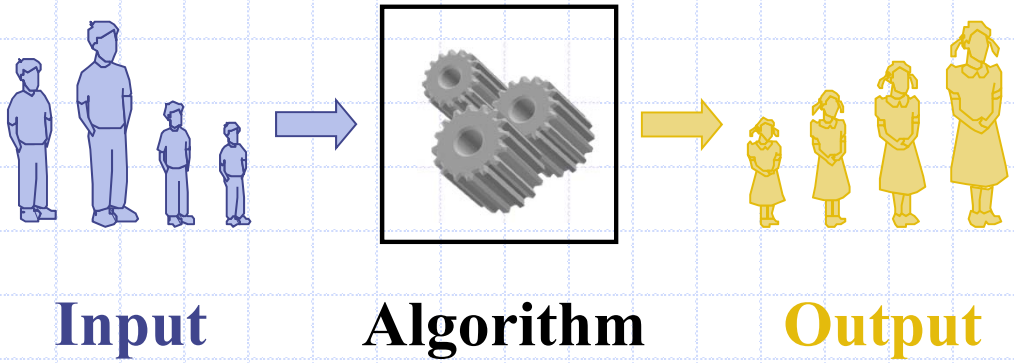
Chapter 14

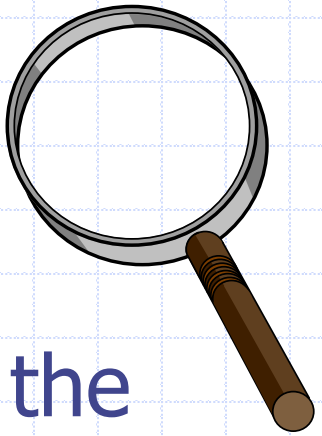
Algorithmic Frameworks



Outline and Reading

- External Memory Algorithms (14.1)
- Parallel Algorithms (14.2) - Optional
- Online Algorithms (14.3)

Framework



- Our computational framework in the book so far has been "unrealistic":
  - Single Processor
  - Unlimited Memory with equal access time
  - Single input, single output

Memory Hierarchy

- Caches, Main Memory, Blocking, disk blocks, and pages
- Principle of Locality
- Virtual Memory
- Large data size
- These require a new model for External Searching

(a,b) Trees and B-Trees

- (a, b) trees are generalization of (2,4) trees
  - Multi-way search tree
  - Each node stores between a-1 and b-1 items, where 2<=a<=(b+1)/2
  - Supports searching, inserting, and removing
  - Each internal node has at least a children and at most b children
  - All external nodes have the same depth
- A B-tree of order d is an (a,b) tree with a= ⌈d/2⌉, and b=d
  - See Figure 14.4

Competitive Ratio

Let A be an online algorithm and OPT an offline optimal algorithm for a sequence of services P.

Then A is c-competitive if

Cost(A,P)<=c x cost(OPT,P)+b for some b>=0. c is the competitive ratio of A.

Online Algorithms, Caching

- Locality of reference dictates to store copies of web-pages in cache, say in maximum of m slots
  - Fully associative
  - Replacement Policy: FIFO, LRU, Random
    - Worst case for FIFO and LRU is a loop repeatedly requesting m+1 pages in cyclic order
    - But in reality both are good with competitive ratio of m (see Theorem 14.10)