

Agenda

- ANALYSIS OF ALGORITHMS:**
- ONLINE PROBLEMS AND AMORTIZED ANALYSIS**
 - COMPETITIVE ANALYSIS**
 - REVIEW: PAGING PROBLEMS**
 - REVIEW: DICTIONARY DATA STRUCTURE**
 - INPUT DISTRIBUTIONS**

Amortized Analysis

- Usually, algorithms are analyzed for
 - (i) worst case behavior and (ii) average case behavior
- Average case behavior is aggregated (to compute the average) over a sequence of inputs:
 - But usually a specific input distribution is assumed – that is often uniform:
 - e.g. analysis of sorted list vs. unsorted list
 - Real workloads often behave differently: *uniform distribution is not the common case!*

Amortized Analysis

- Recall the competitive analysis of page replacement algorithms:
 - Worst case behavior was measured but it was averaged over a sequence of inputs!
 - This is referred to as *amortized analysis*.

Dictionary Data Structure

- Consider the dictionary data structure with its typical operations:
 - find, insert, and delete
- Usually analysis is done on an individual operation:
 - e.g. what is the worst case time complexity of a find operation in a list?
- Or it is averaged over a sequence of operations:
 - e.g. what is the average case time complexity of a find operation in a list?
 - The answer to this depends on input distribution.
- In fact the way the list can be best arranged will depend on the input distribution.

Amortized Analysis

- Consider a hashtable with separate chaining on collision:
 - What is the cost of find operation?
 - What does it depend on?
 - How do you adapt when collisions increase?
 - What is the cost?