

## Amortized vs. Average

Amortized time is a time-averaged running time.

- based on a worst-case analysis of the running time of an arbitrary sequence of operations
  - worst-case running time of any sequence of  $n$  operations /  $n$
- gives the average worst-case performance of each operation
  - but any particular instance of the operation may be (far) worse
- useful when expensive cases exist but occur infrequently
  - e.g. dynamic array resizing
  - e.g. union-find with path compression
  - e.g. splay trees

## Amortized vs. Average

Amortized time is a **time-averaged** running time.

- worst-case analysis of the running time of an arbitrary sequence of operations
  - worst-case running time of any sequence of  $n$  operations /  $n$
- average worst-case performance of each operation
  - any single operation may be (far) worse
  - total for the sequence will not exceed  $n \times$  operation time

Average time is an **instance-averaged** running time.

- average-case analysis of the running time of an operation
  - based on the probability of each input instance occurring
- expected performance of each operation
  - any single operation may be (far) worse
  - low (but non-zero) probability that total for a sequence will exceed  $n \times$  operation time