Abstract Data Type (ADT)

1. A description of some data
2. A list of operations to be performed on it.
String ADT

① A sequence of characters
② char at index

length
concatenation
delete char at index

Instances of Strings: “dog”, [contents of Hamlet]
Priority queue ADT

1. A collection of things, each of which has a "key" from an ordered set

2. Enqueue — add a (key, thing) to the PQ
   Dequeue — remove the (key, thing) with the largest (smallest?) key

Size/length

Dijkstra's Alg needs one of these
PQ, min, using a singly linked list, sorted by key (min at head)

Enqueue: $O(N)$

Dequeue min-priority: $O(1)$
PQ w/ a min-heap

- Binary tree
- Each node's key is ≥ its parent's key
- It's "full"
- each level has maximum # of nodes except bottom level, filled left-to-right

Enqueue:
Depends on implementation
Enqueue: 6

1. Stick 6 at the end
2. Fix the heap condition by walking the new node up the heap.
Dequeune:  
0. remove root  
① move bottom right item to the root  
③ fix the heap  
Deguene $O(\log N)$
K's children are at indexes $2k+1$ and $2k+2$.
Q: A B C D E F G H I J

\( \text{dist}: 2 7 2 3 3 1 1 1 1 1 1 \)

\( \text{dist}[u] = 0 \)

\( v = A \quad \text{alt} = 0 + 2 \quad \text{if alt} < \text{dist}[A] \quad \text{dist}[A] = 2 \)

1st iteration

\( u = J \)
Q: A B C D E F G H I

\text{dist}: 2 \infty \infty \infty \infty \infty \infty \infty \infty

\text{dist}[u] = 1 + 5
\text{dist}[u] = \text{alt}

\text{alt} < \text{dist}[u]
\text{dist}[u] = \text{alt}

u = I

v = C
\text{alt} = 1 + 5