Sequences

EECS 214

October 19, 2015

Doubly-linked lists

```
struct Node<T>:
    prev:  ref Node<T>
    value: T
    next:  ref Node<T>
```

Doubly-linked lists

```
struct Node<T>:
    prev:  ref Node<T>
    value: T
    next:  ref Node<T>
```

- Doubly-linked list:
  - `n.next.prev` = `n`
  - `n.prev.next` = `n`

Doubly-linked lists

```
struct Node<T>:
    prev:  ref Node<T>
    value: T
    next:  ref Node<T>
```

- Doubly-linked list:
  - If `n.next` $\neq$ `nil` then `n.next.prev` $=$ `n`
  - If `n.prev` $\neq$ `nil` then `n.prev.next` $=$ `n`

Doubly-linked lists

```
struct Node<T>:
    prev:  ref Node<T>
    value: T
    next:  ref Node<T>
```

- Doubly-linked list:
  - If `n.next` $\neq$ `nil` then `n.next.prev` $=$ `n`
  - If `n.prev` $\neq$ `nil` then `n.prev.next` $=$ `n`
  - There's some j such that `n.prev`^{j} $=$ `nil`
  - There's some k such that `n.next`^{k} $=$ `nil`

Doubly-linked lists

```
struct Node<T>:
    prev:  ref Node<T>
    value: T
    next:  ref Node<T>
```

- Doubly-linked list:
  - If `n.next` $\neq$ `nil` then `n.next.prev` $=$ `n`
  - If `n.prev` $\neq$ `nil` then `n.prev.next` $=$ `n`
  - There's some j such that `n.prev`^{j} $=$ `nil`
  - There's some k such that `n.next`^{k} $=$ `nil`
- Circular doubly-linked list:
  - `n.next.prev` $=$ `n`
  - `n.prev.next` $=$ `n`

DLL insertion

To insert `o` after `n` (DLL):

```
o.prev = n
o.next = n.next
o.prev.next = o
if o.next ≠ nil:
    o.next.prev = o
```

DLL insertion

To insert `o` after `n` (DLL):

```
o.prev = n
o.next = n.next
o.prev.next = o
if o.next ≠ nil:
    o.next.prev = o
```

To insert `o` after `n` (CDLL):

```
o.prev = n
o.next = n.next
o.prev.next = o
o.next.prev = o
```

Lists as sequences

How do we:

- lookup an element by position?
- insert an element?
- insert an element at position?
- delete an element at position?

Lists as sequences

How do we:

- lookup an element by position?
- insert an element?
- insert an element at position?
- delete an element at position?

How long does it take?

Arrays as sequences

Assume we leave extra space at the end:

```
struct Vector<T>:
    size: ℕ
    data: Array<T>
```

How do we:

- lookup an element by position?
- insert an element?
- insert an element at position?
- delete an element at position?

Arrays as sequences

Assume we leave extra space at the end:

```
struct Vector<T>:
    size: ℕ
    data: Array<T>
```

How do we:

- lookup an element by position?
- insert an element?
- insert an element at position?
- delete an element at position?

How long does it take?