

# DLLList: A Doubly-Linked List

Open Data Structures



# DLList

A doubly-linked list



# DLList

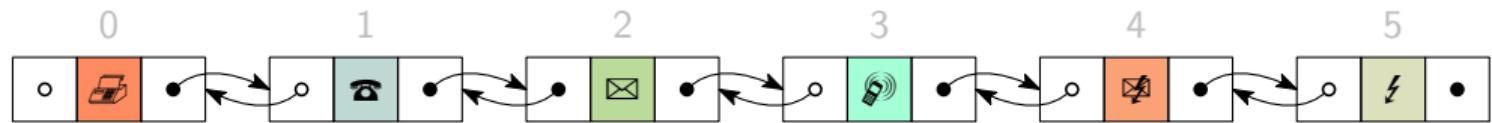
A doubly-linked list

prev x next



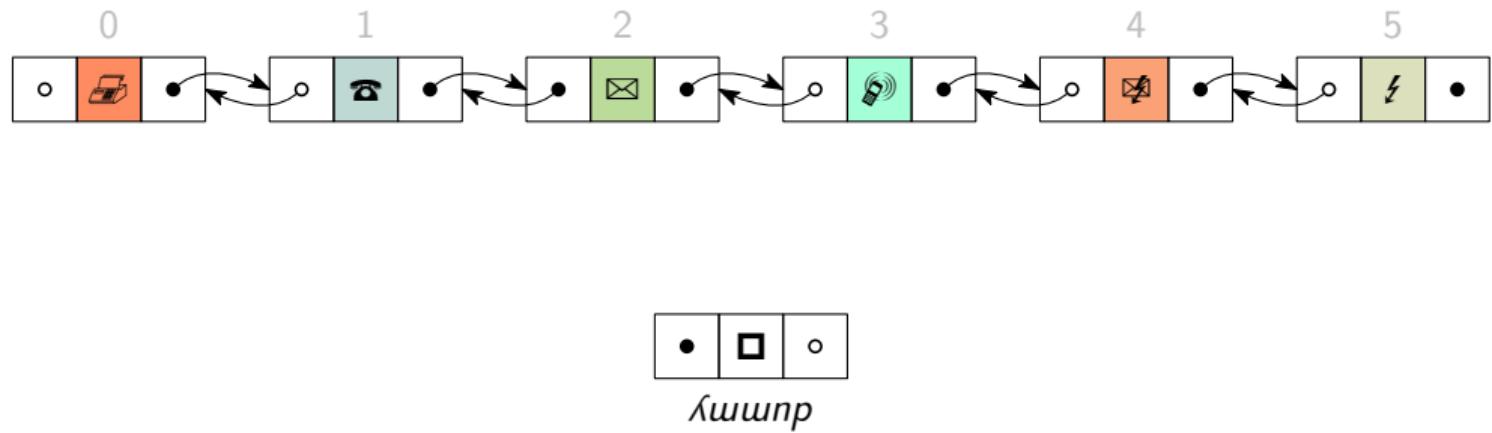
# DLLList

A doubly-linked list



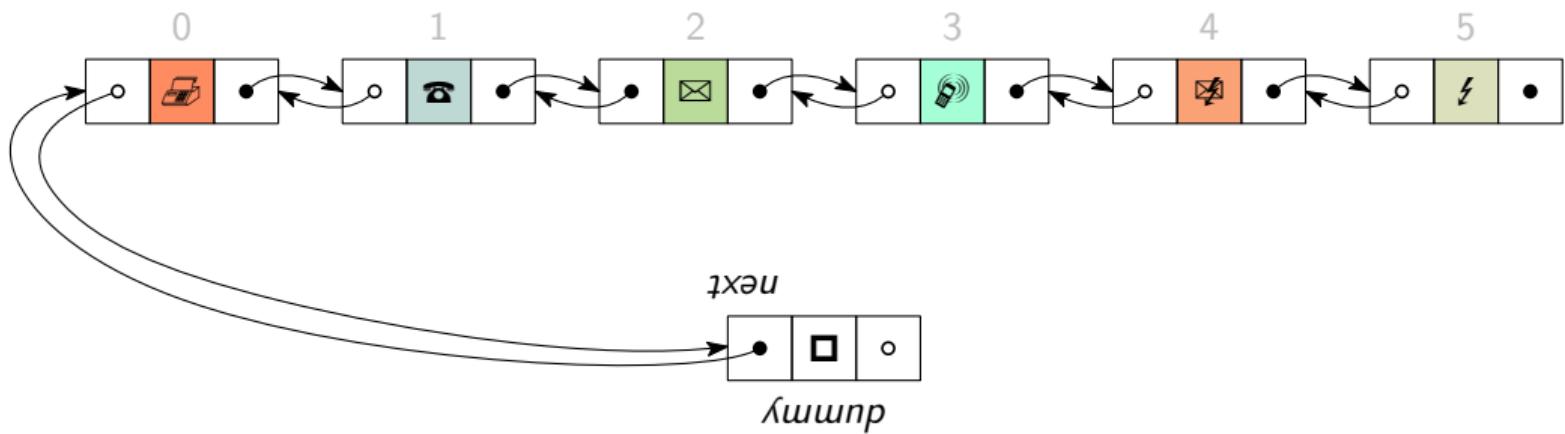
# DLLList

A doubly-linked list



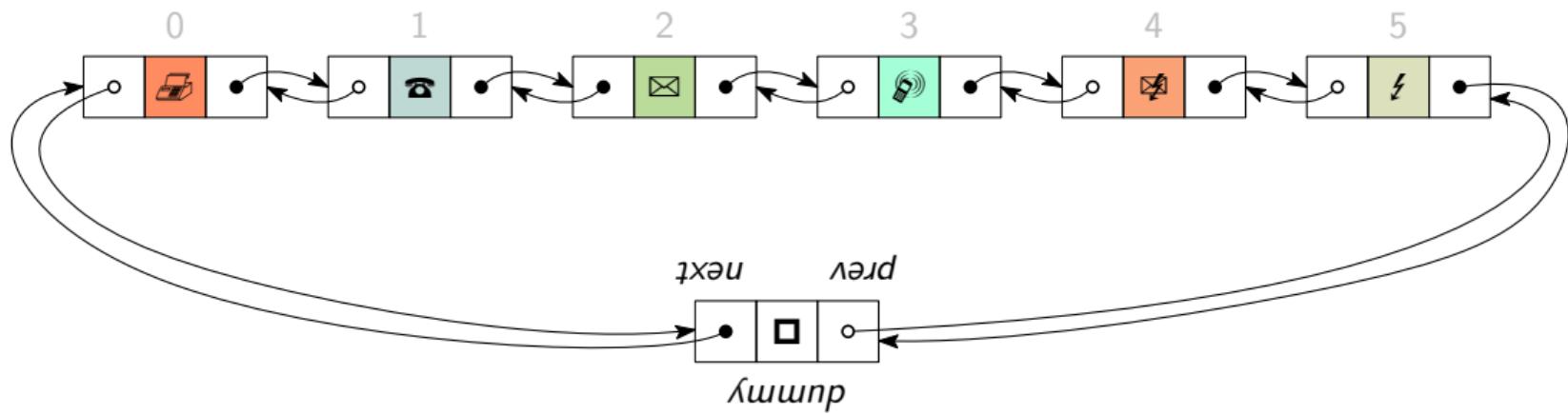
# DLLList

A doubly-linked list



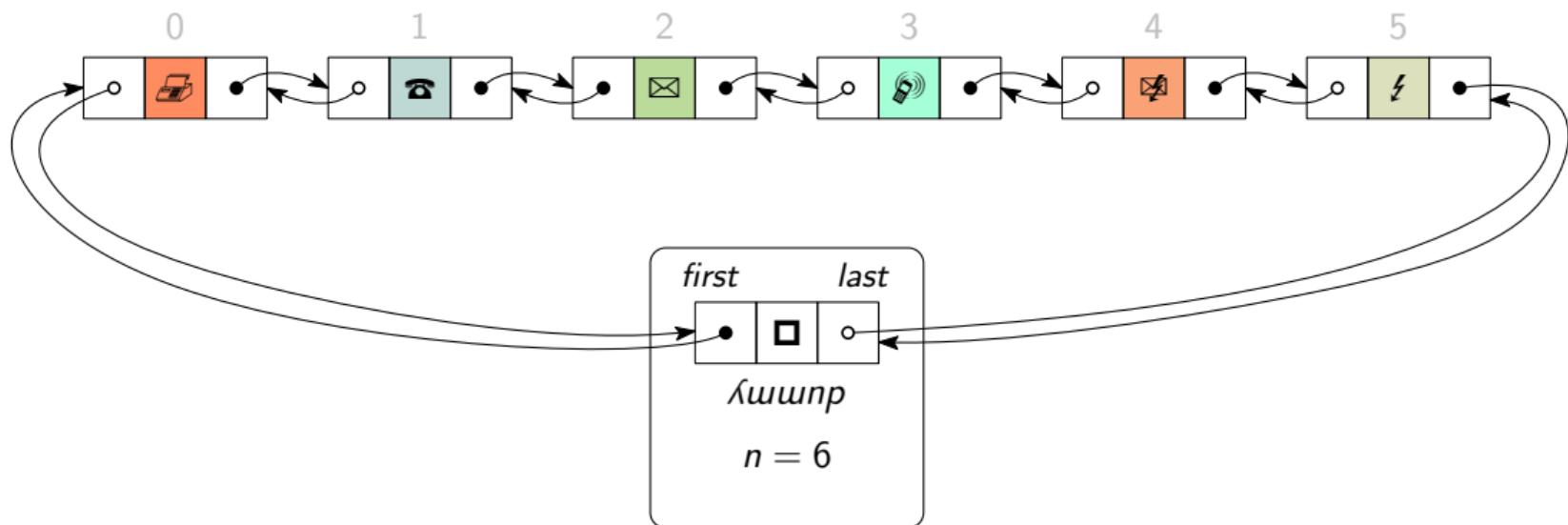
# DLLList

A doubly-linked list



# DLLList

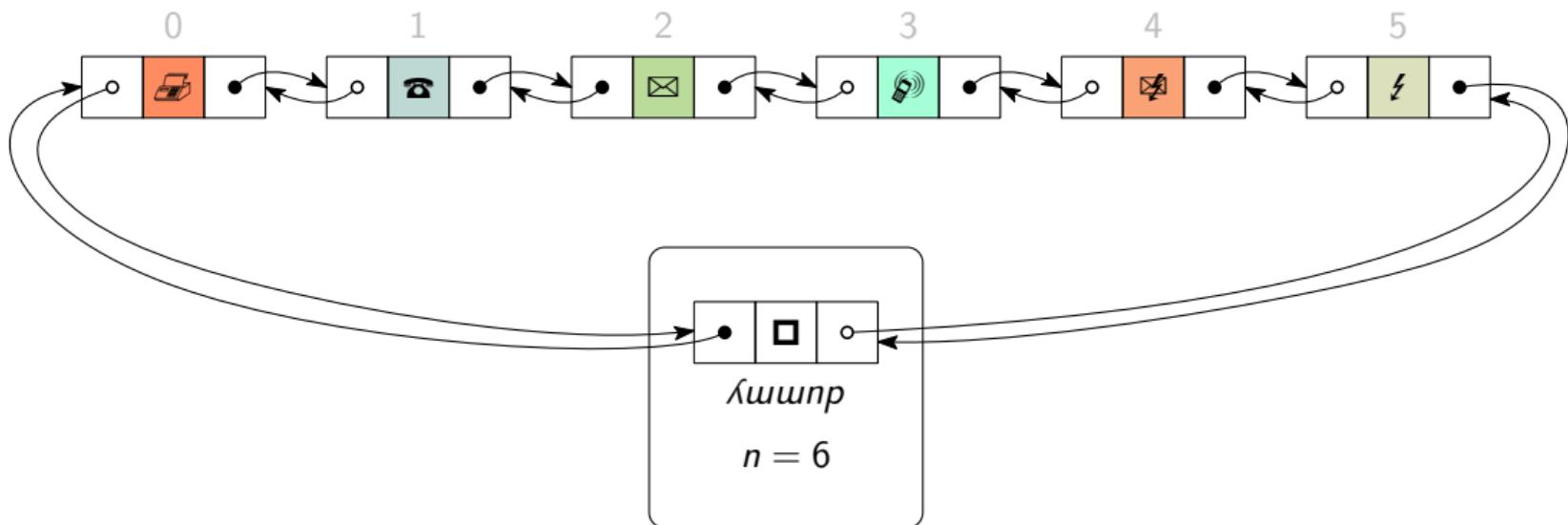
A doubly-linked list



# DLLList

A doubly-linked list

getNode(2)

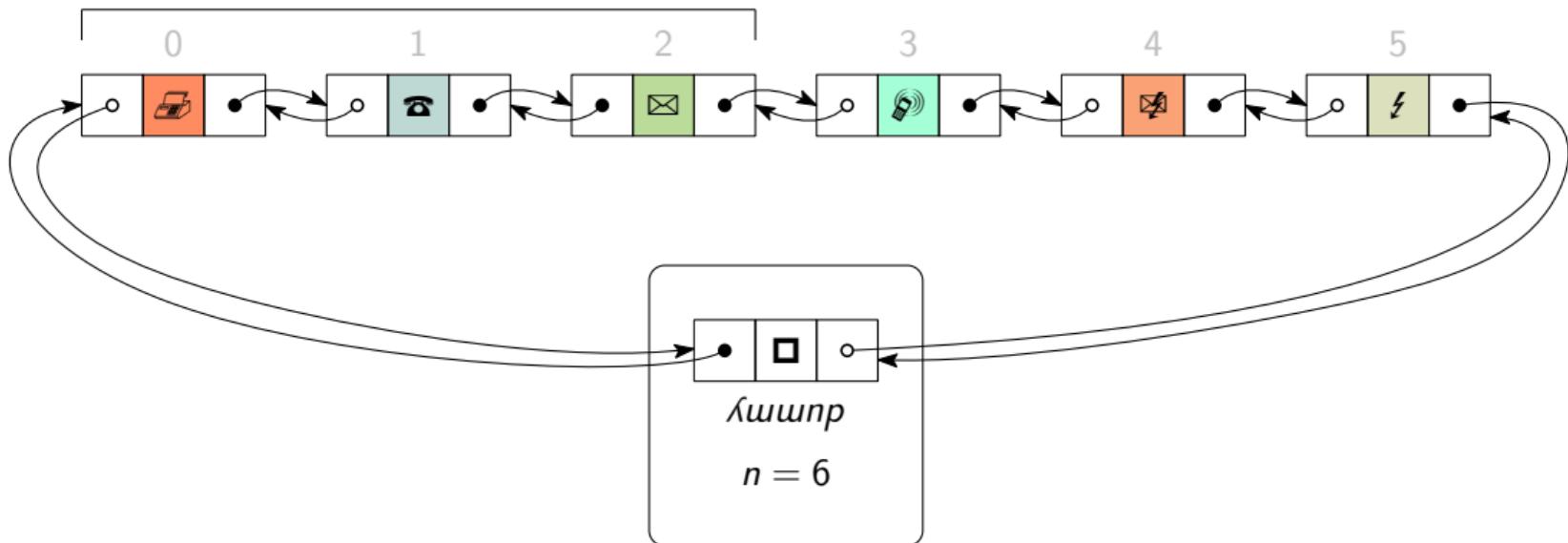


# DLLList

A doubly-linked list

getNode(2)

$2 < n/2$

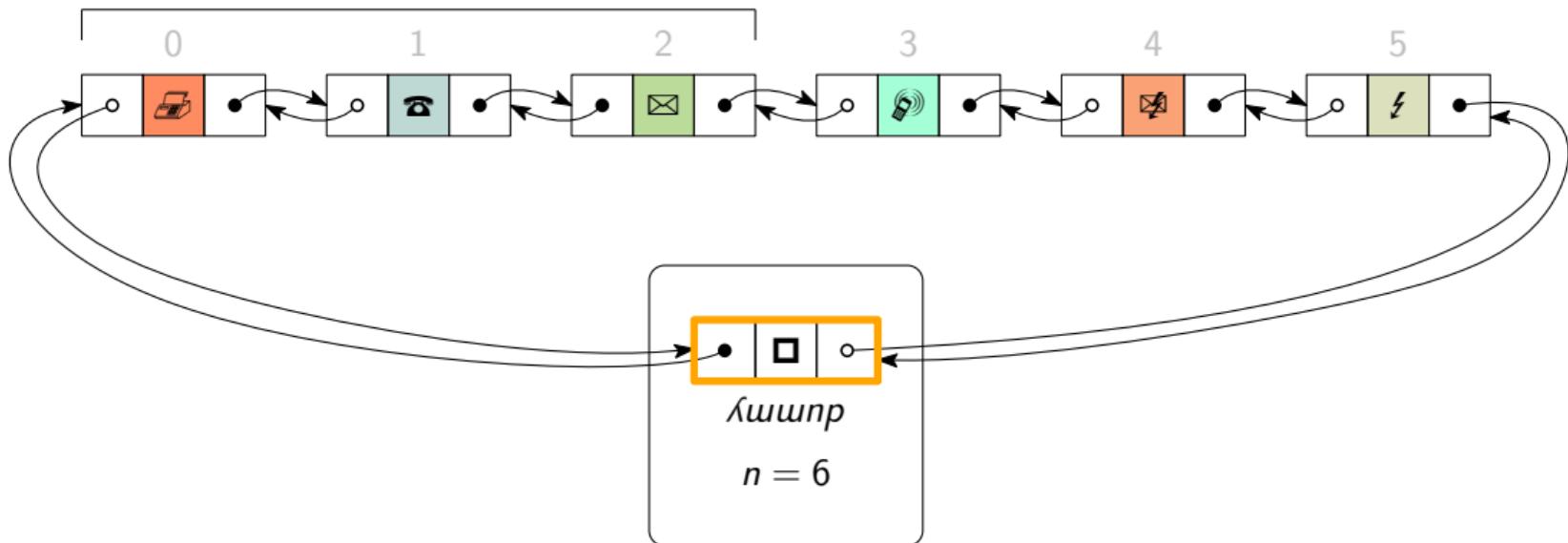


# DLLList

A doubly-linked list

getNode(2)

$2 < n/2$

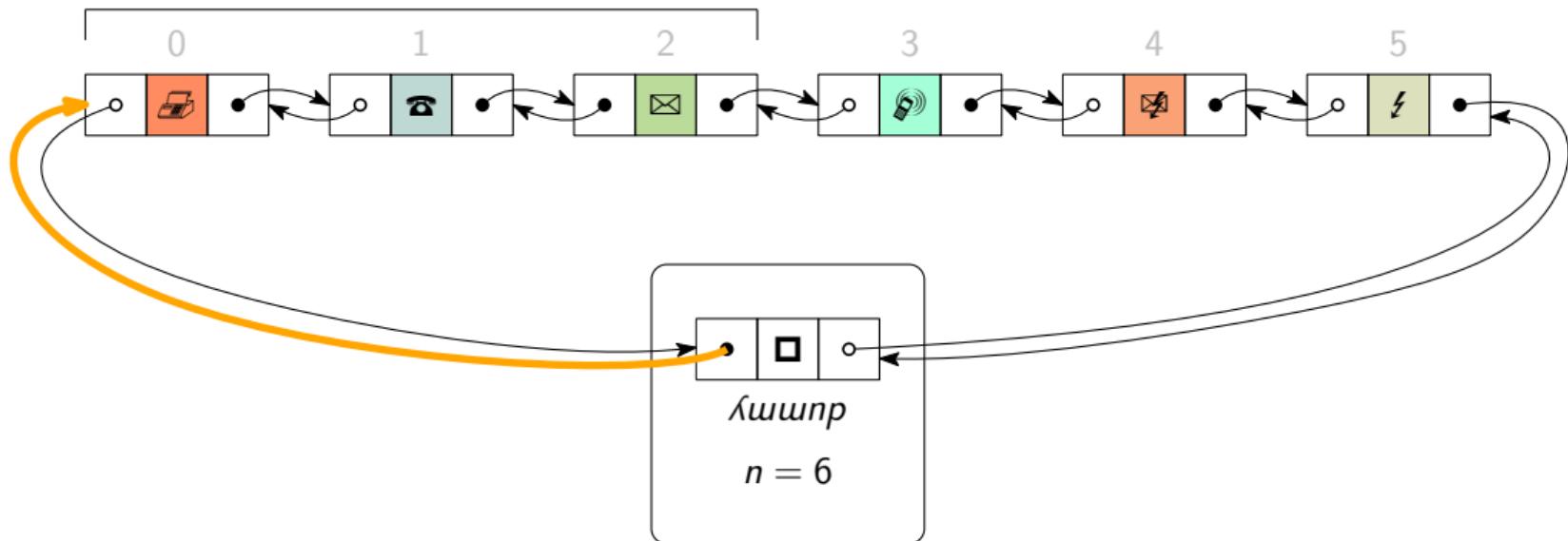


# DLLList

A doubly-linked list

getNode(2)

$2 < n/2$

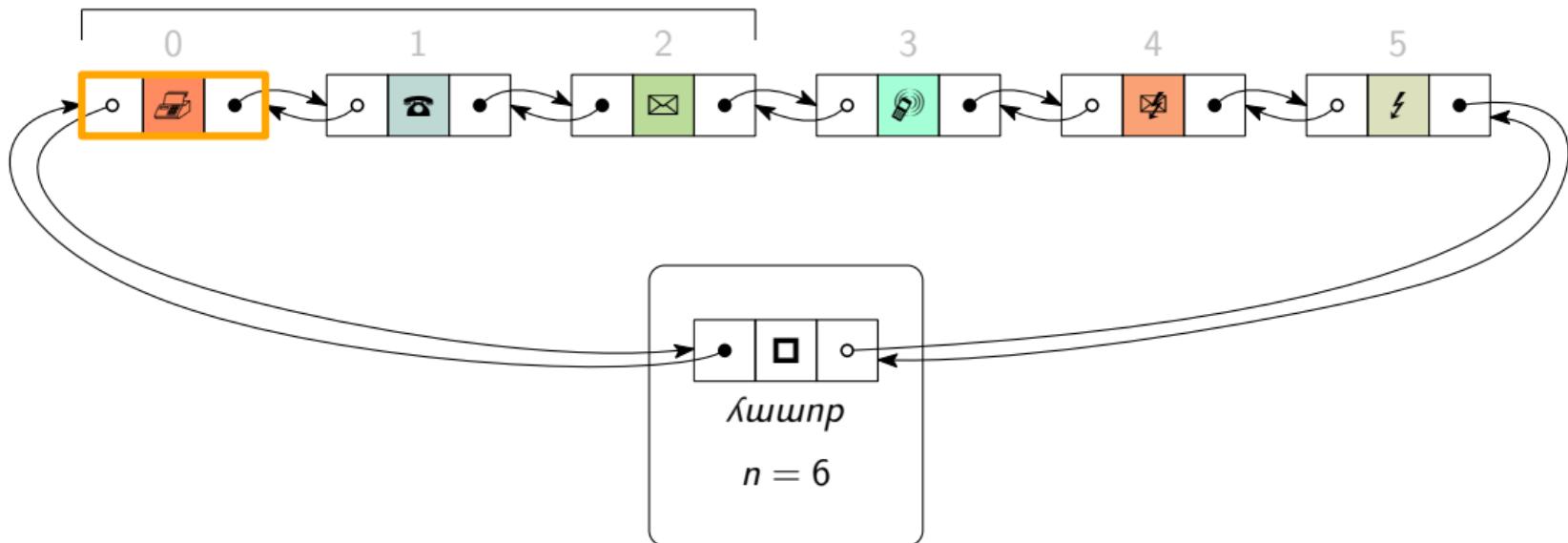


# DLLList

A doubly-linked list

getNode(2)

$2 < n/2$

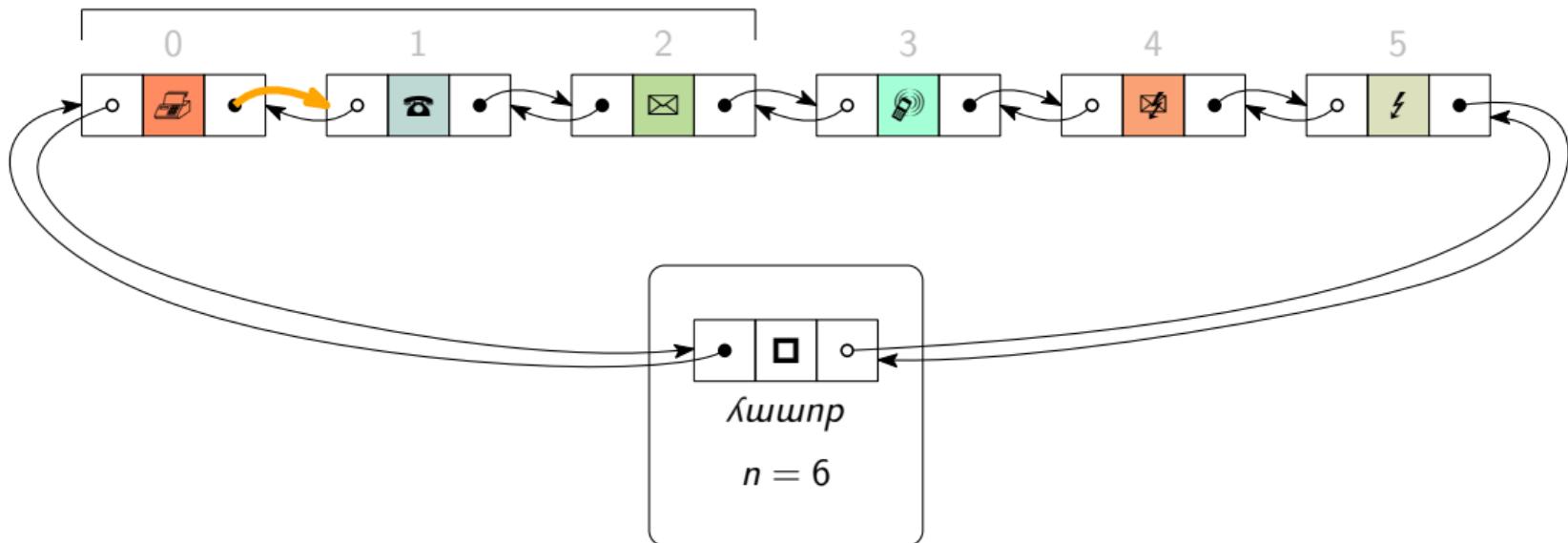


# DLLList

A doubly-linked list

getNode(2)

$2 < n/2$

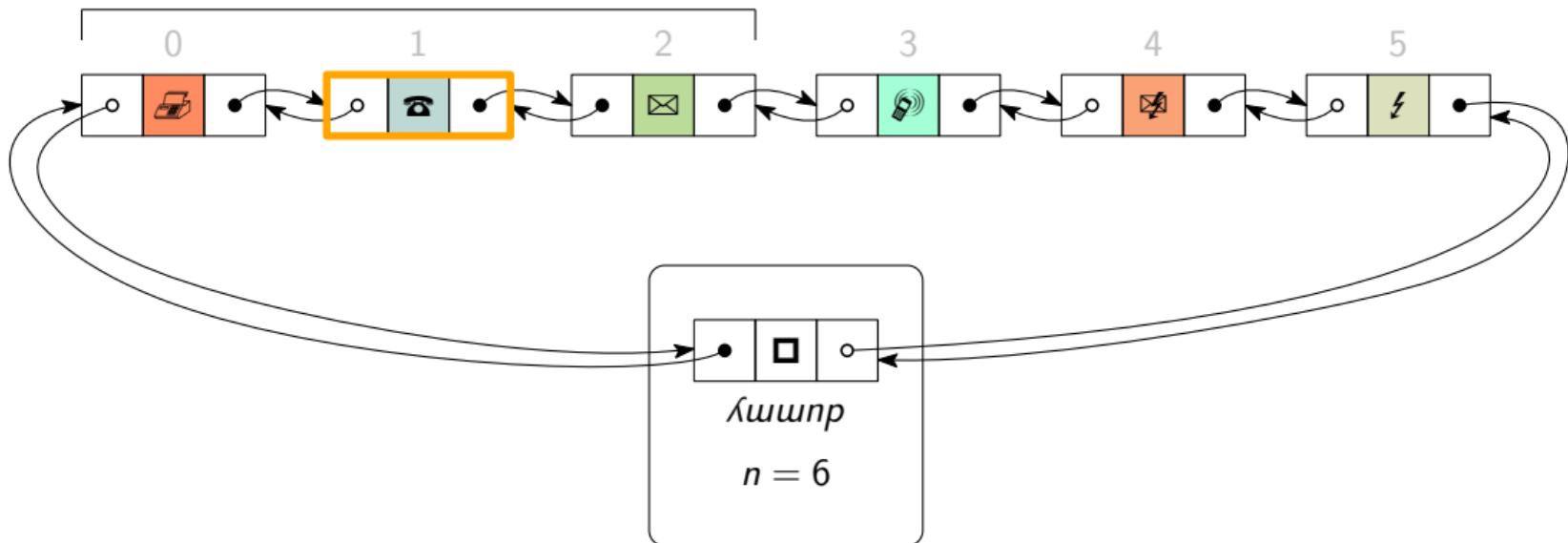


# DLLList

A doubly-linked list

getNode(2)

$2 < n/2$

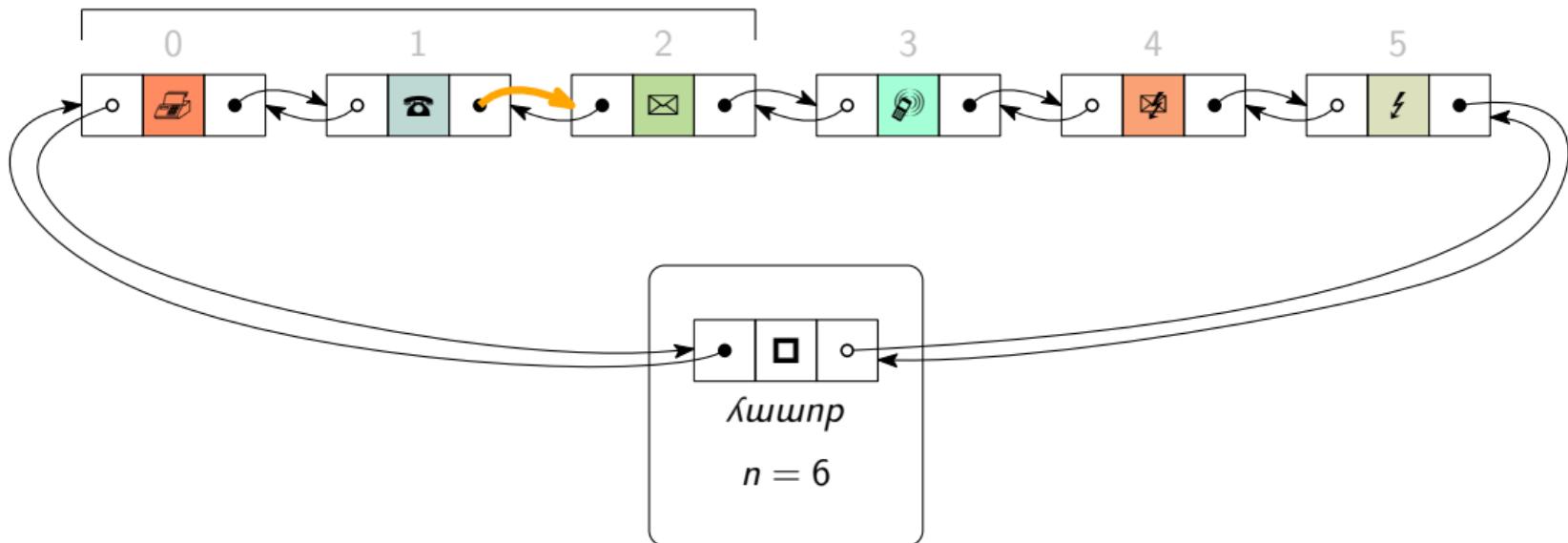


# DLLList

A doubly-linked list

getNode(2)

$2 < n/2$

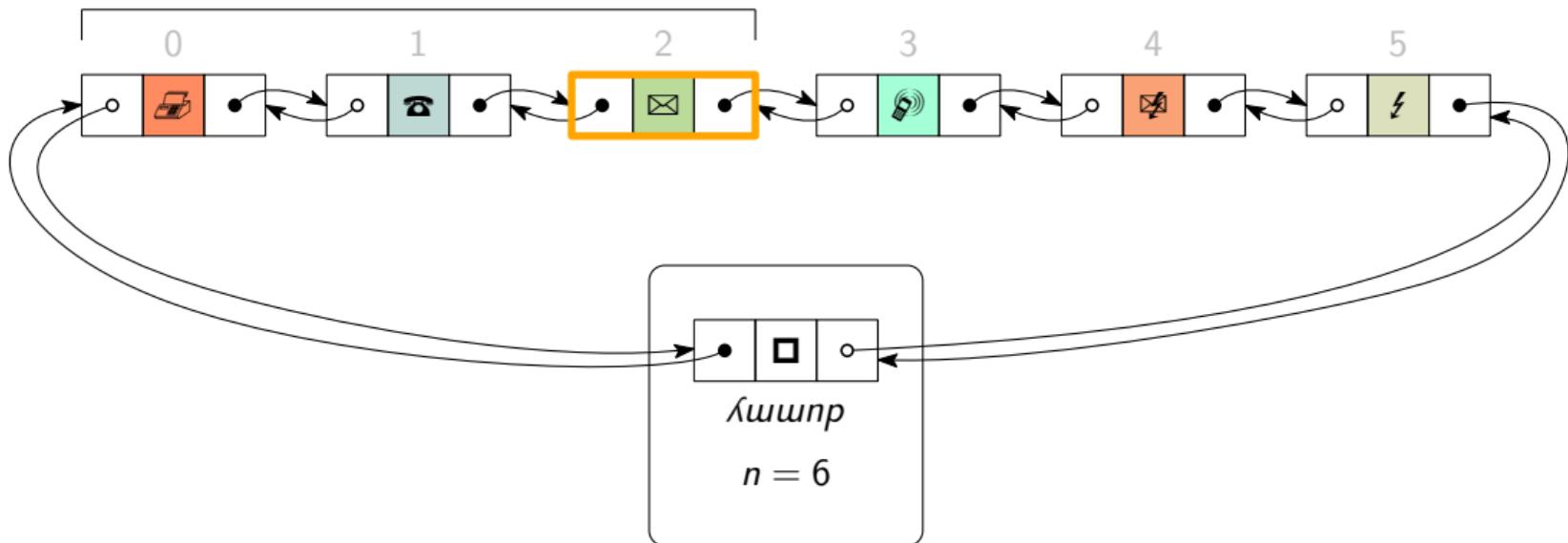


# DLLList

A doubly-linked list

getNode(2)

$2 < n/2$

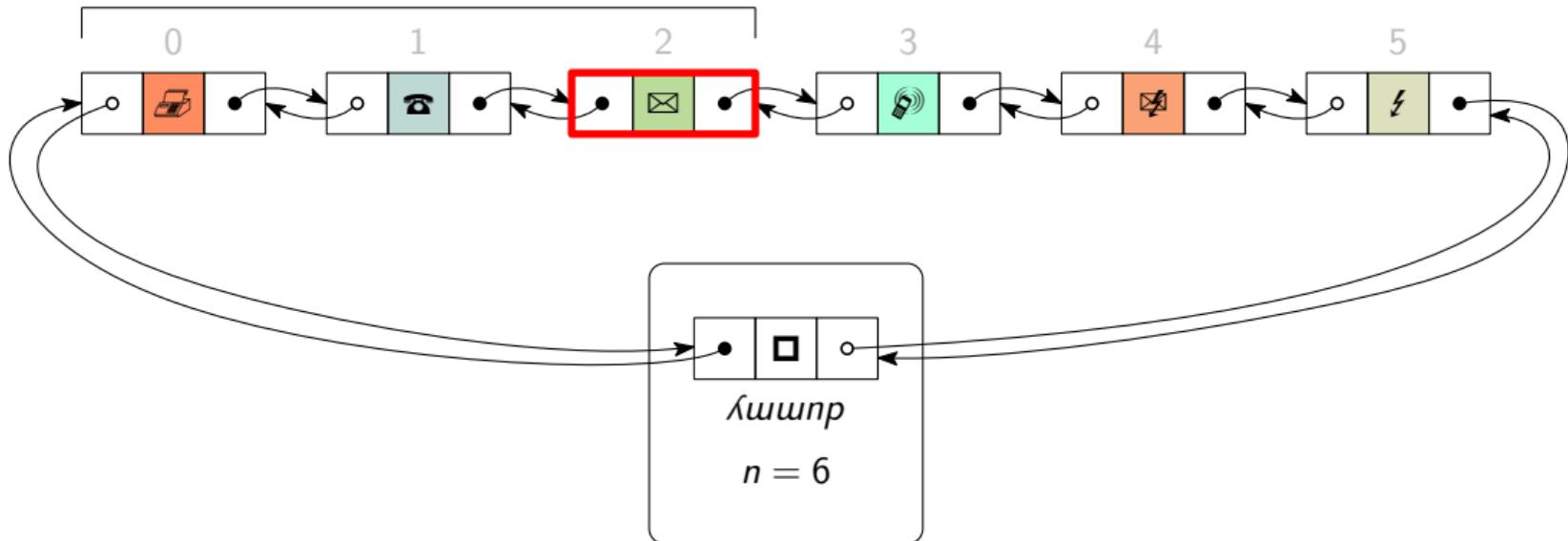


# DLLList

A doubly-linked list

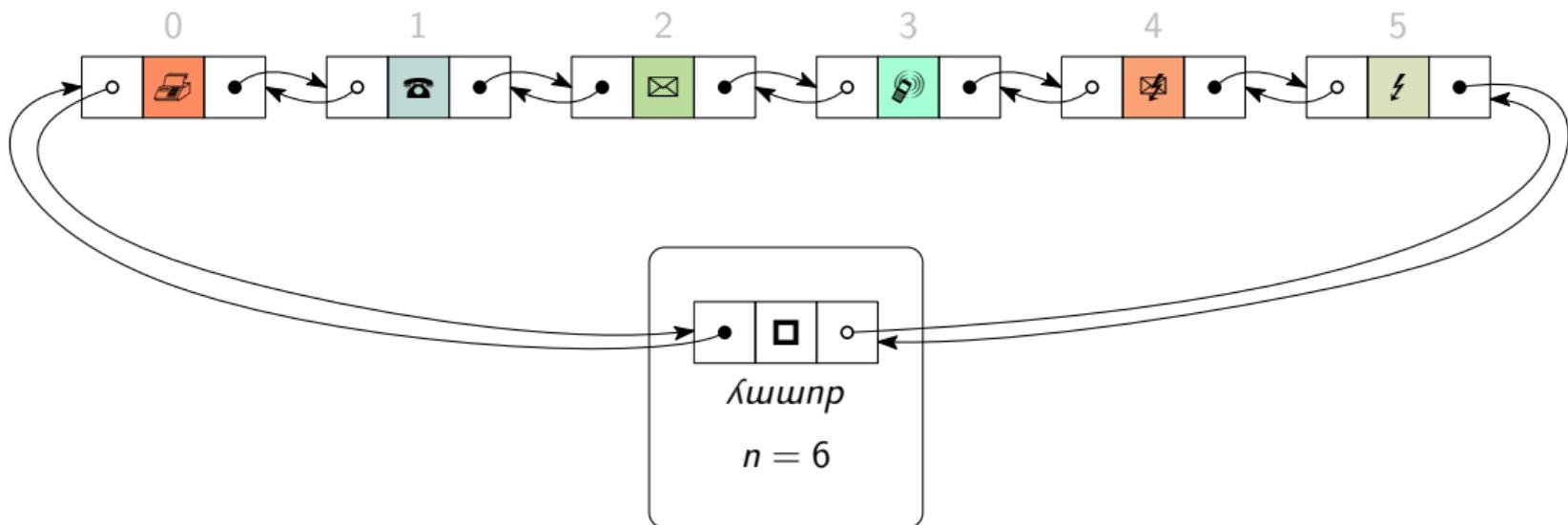
getNode(2)

$2 < n/2$



# DLLList

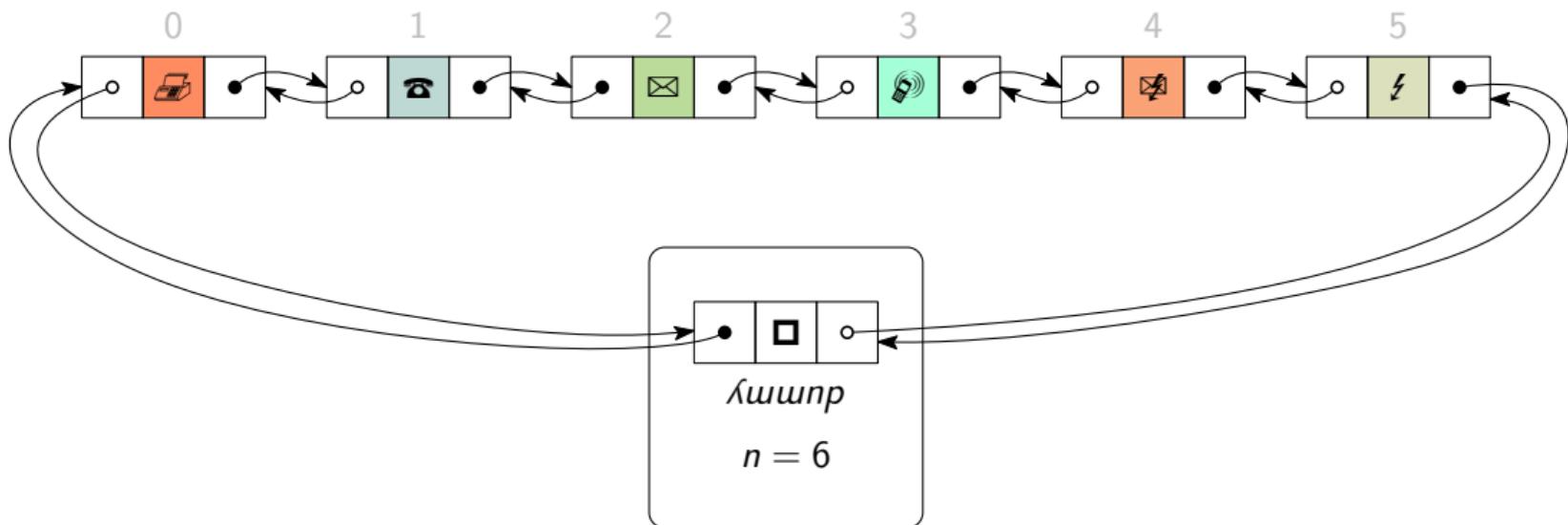
A doubly-linked list



# DLLList

A doubly-linked list

getNode(4)

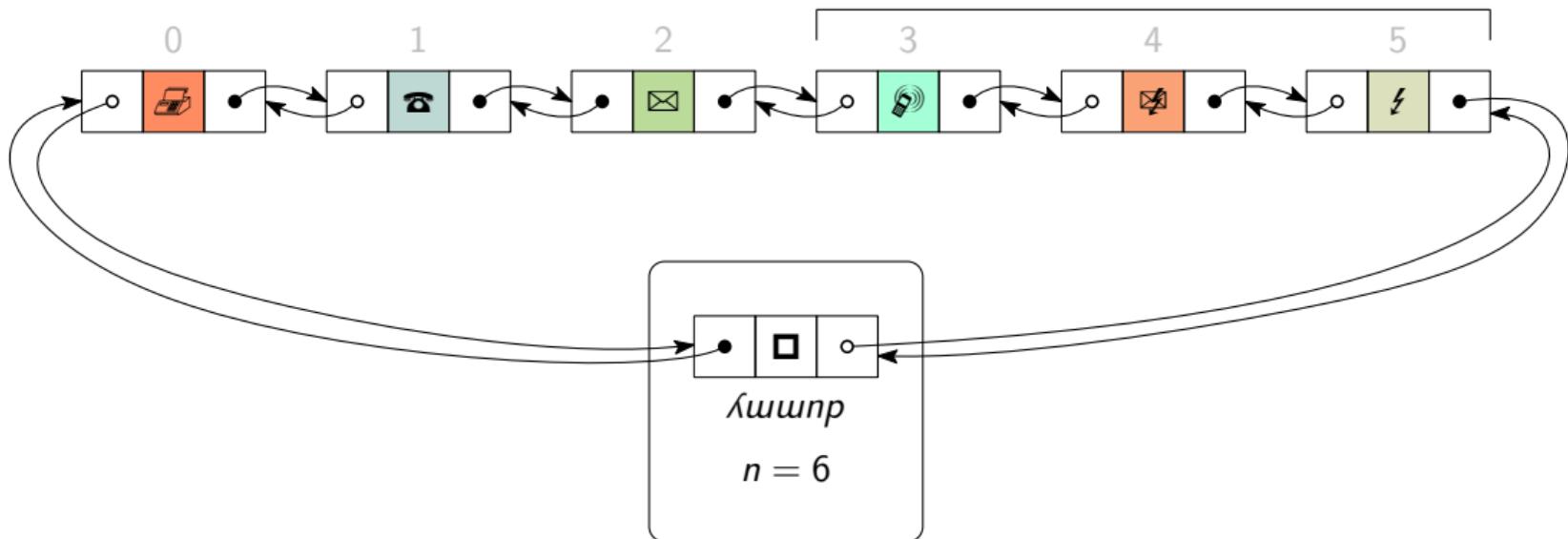


# DLLList

A doubly-linked list

getNode(4)

$4 \geq n/2$

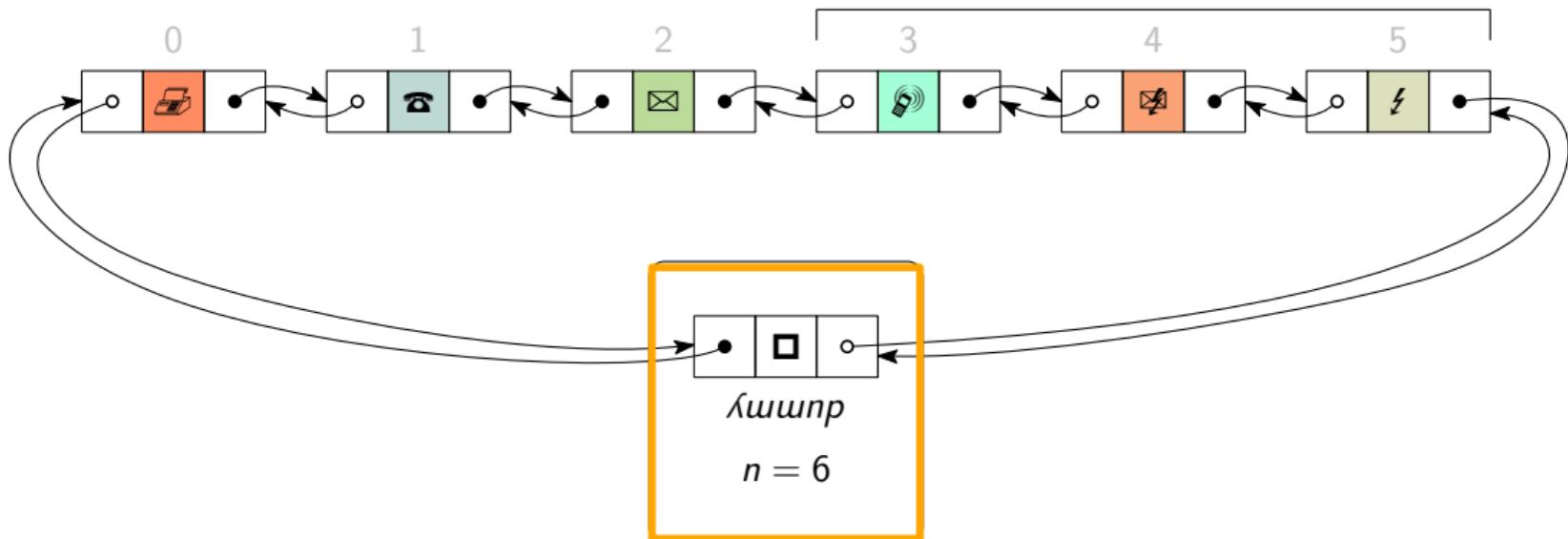


# DLLList

A doubly-linked list

getNode(4)

$4 \geq n/2$

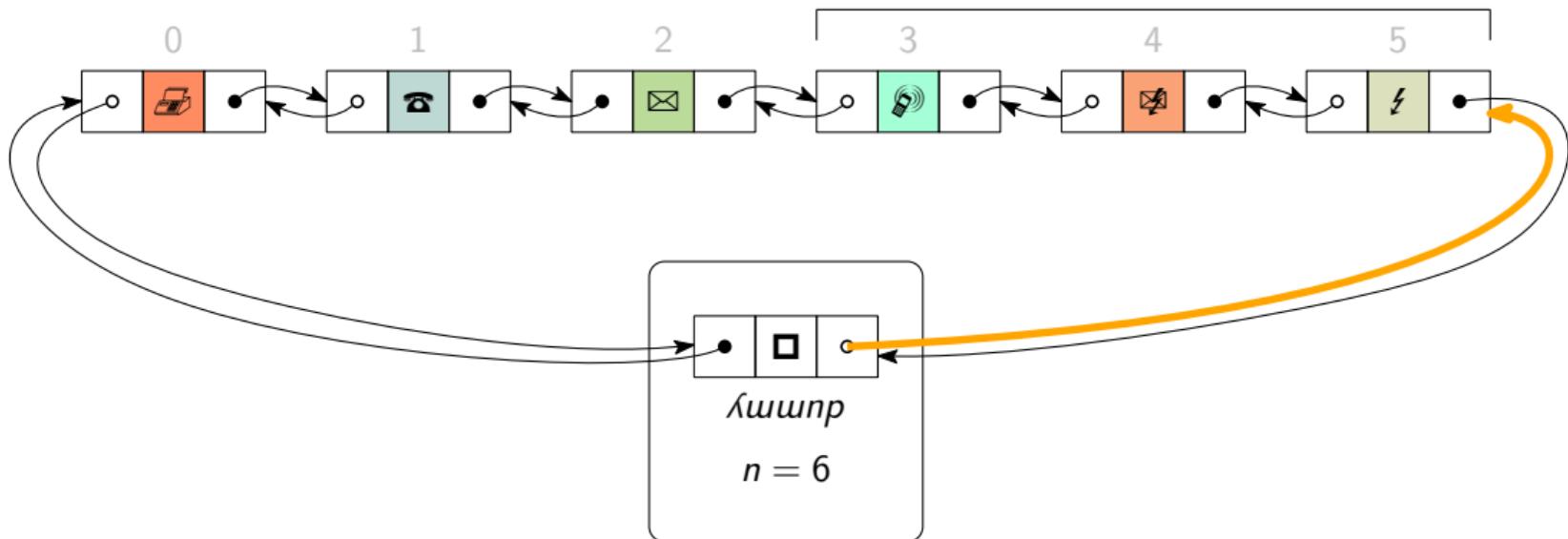


# DLLList

A doubly-linked list

getNode(4)

$4 \geq n/2$

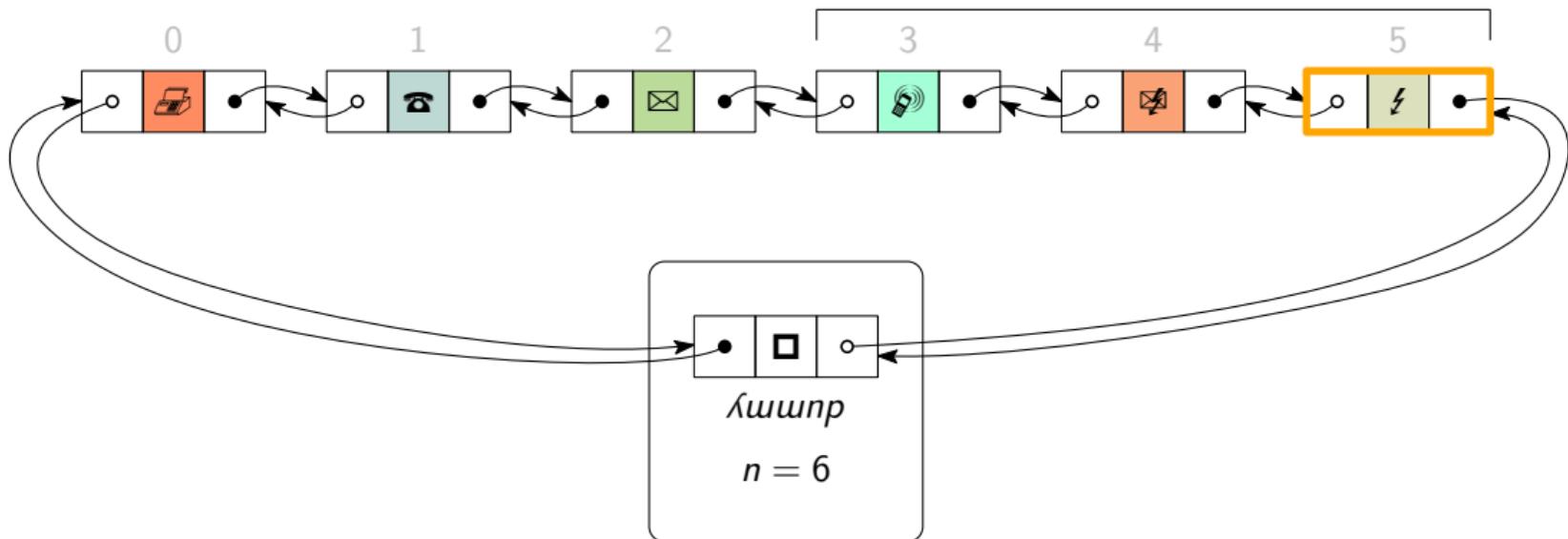


# DLLList

A doubly-linked list

getNode(4)

$4 \geq n/2$

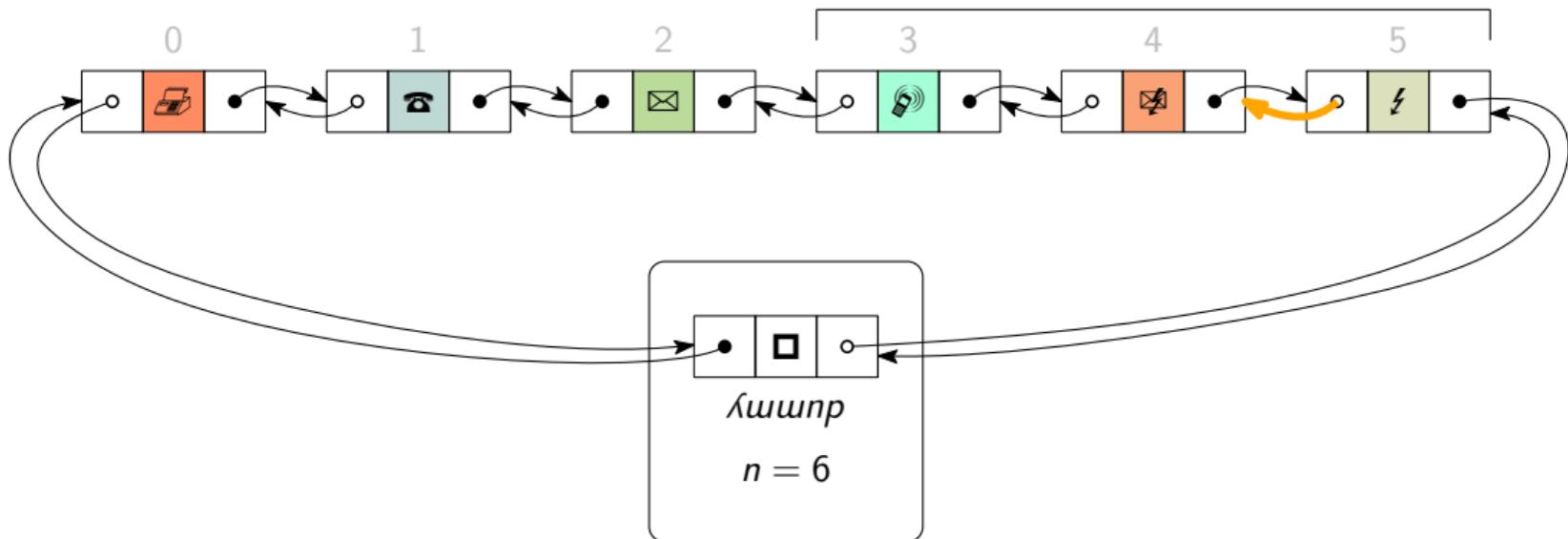


# DLLList

A doubly-linked list

getNode(4)

$4 \geq n/2$

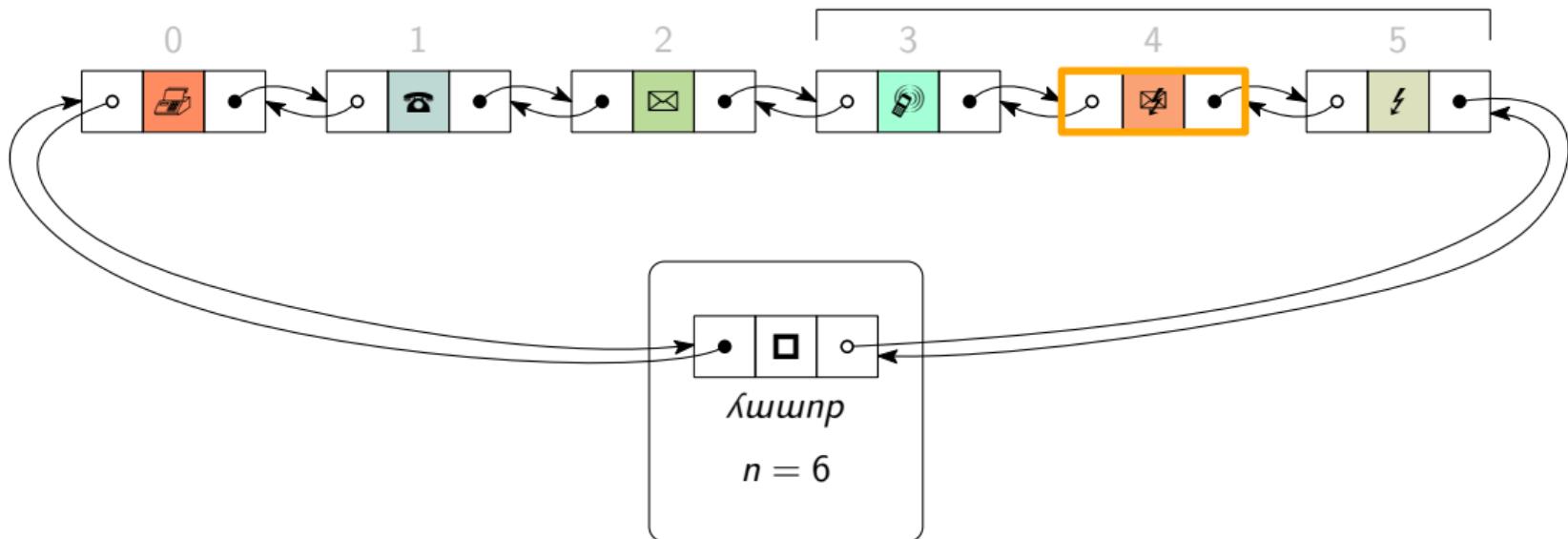


# DLLList

A doubly-linked list

getNode(4)

$4 \geq n/2$

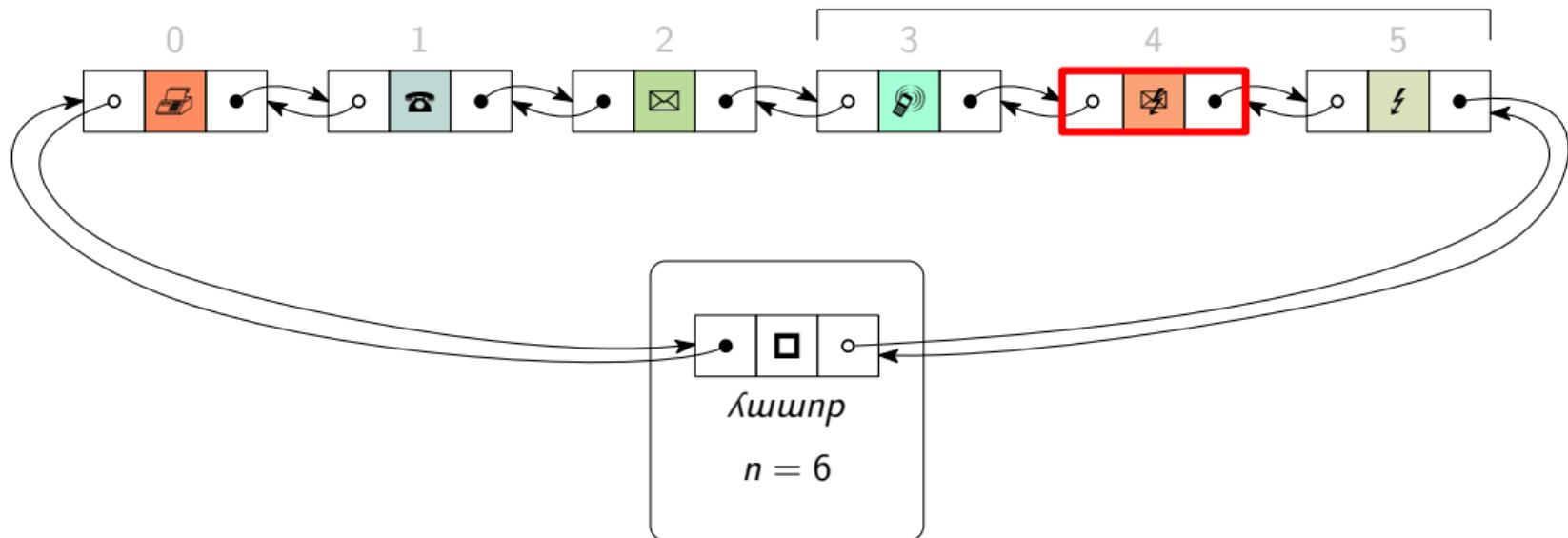


# DLLList

A doubly-linked list

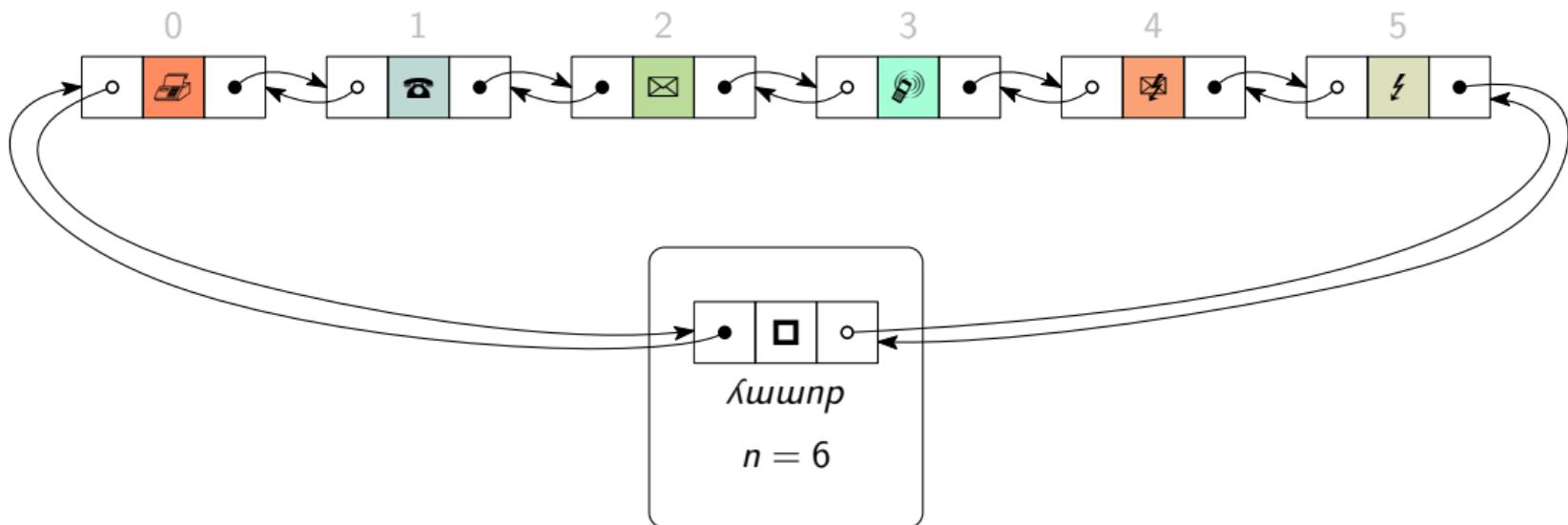
getNode(4)

$4 \geq n/2$



# DLLList

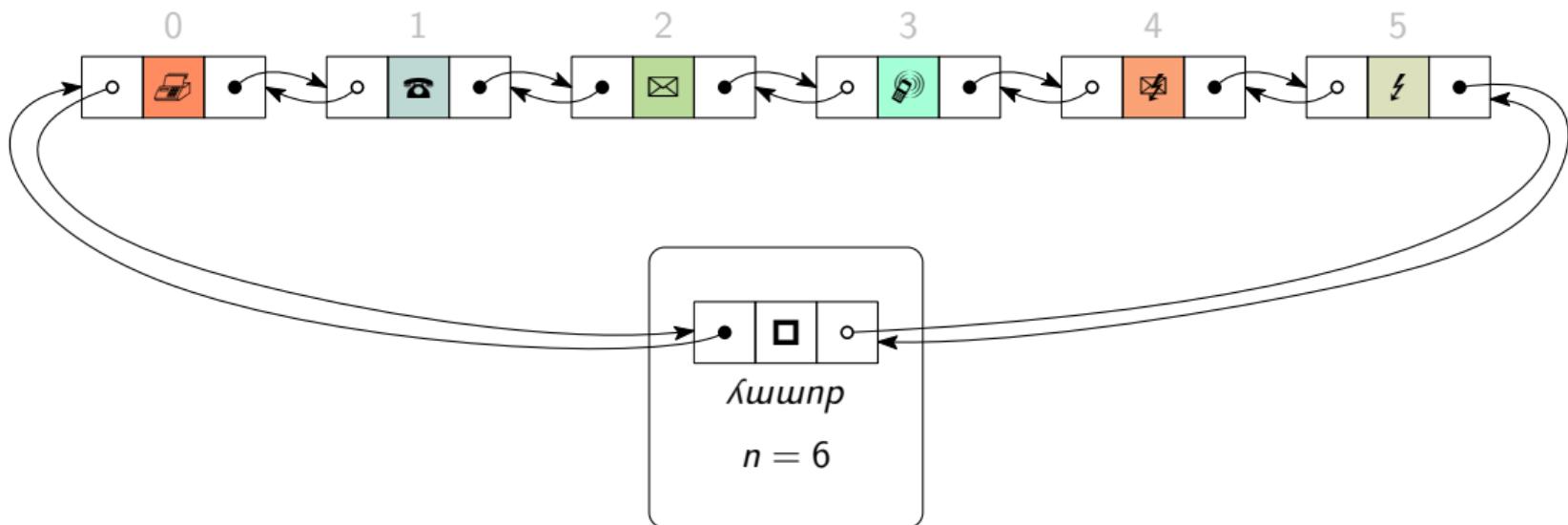
A doubly-linked list



# DLLList

A doubly-linked list

$\text{getNode}(i)$

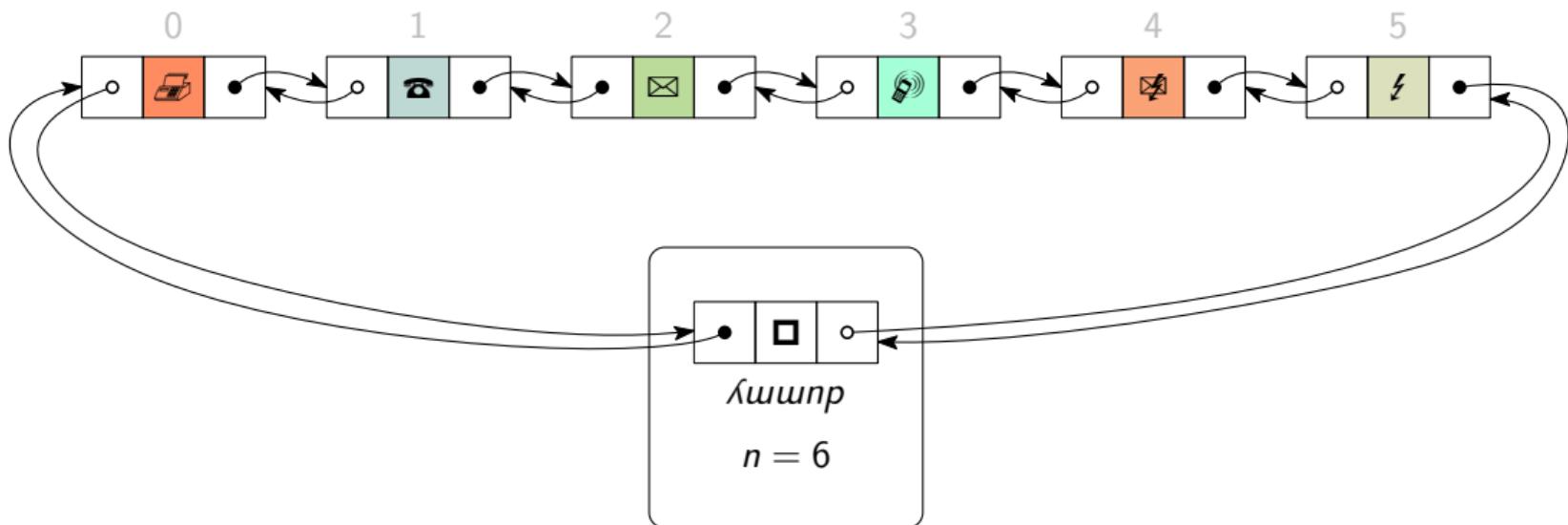


# DLLList

A doubly-linked list

`getNode( $i$ )`

$\min\{i + 1, n - i\}$

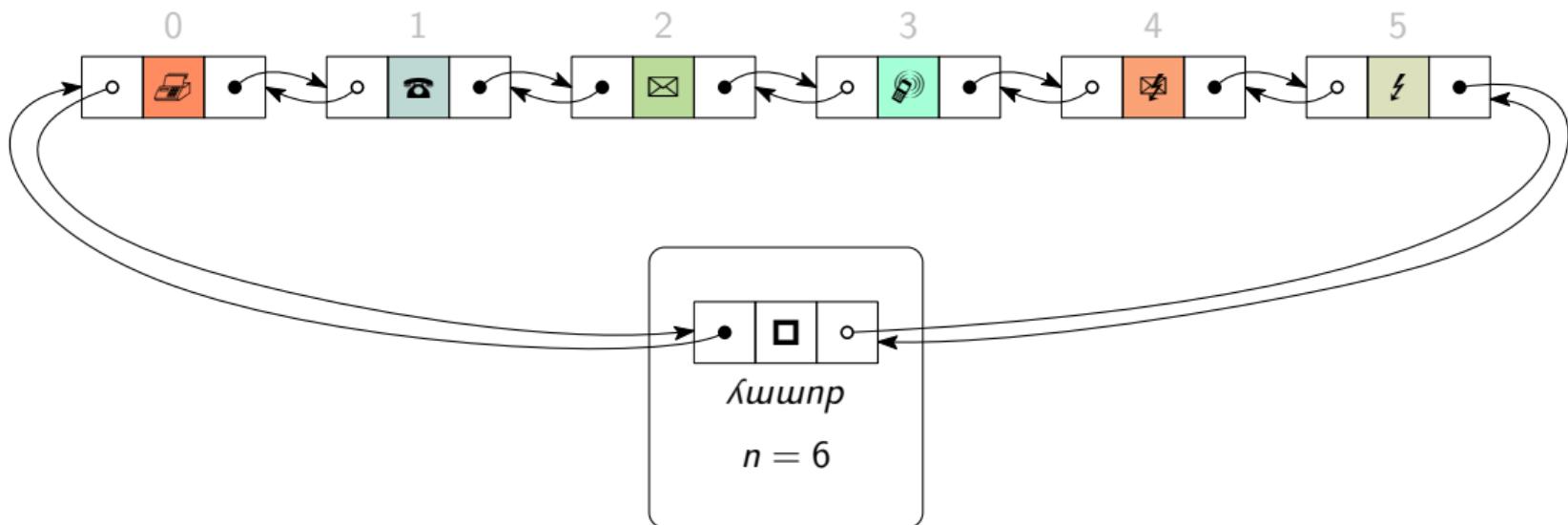


# DLLList

A doubly-linked list

`getNode( $i$ )`

$O(1 + \min\{i, n - i\})$

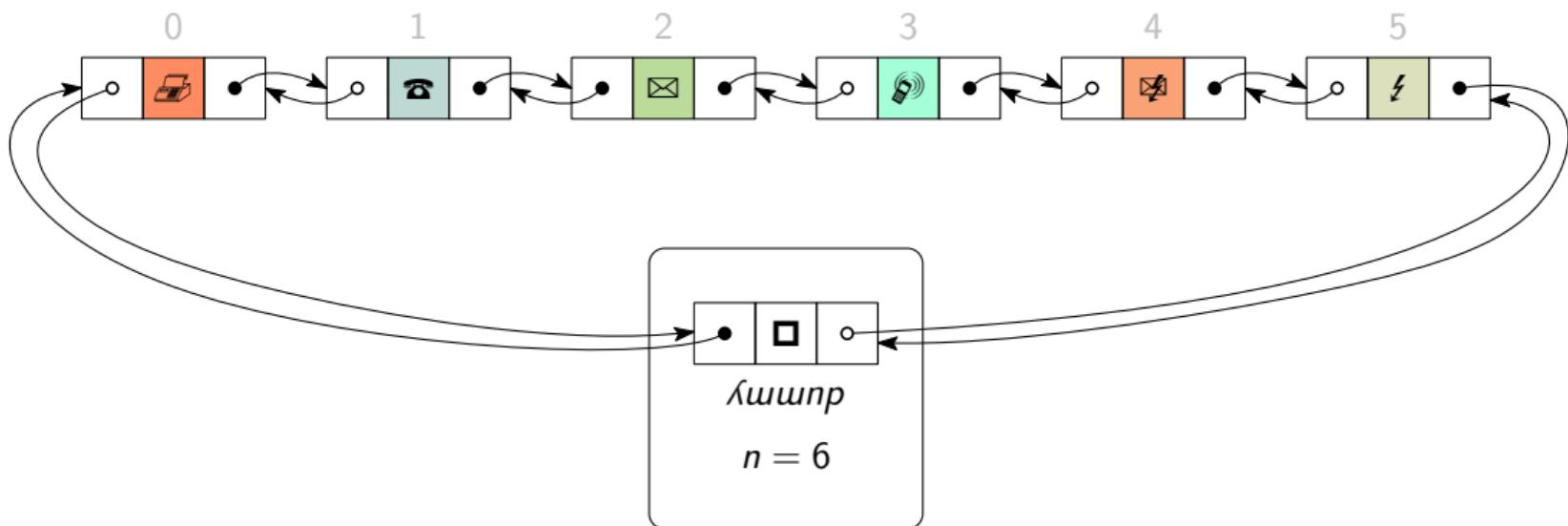


# DLLList

A doubly-linked list

`getNode( $i$ )`

$O(1 + \min\{i, n - i\})$

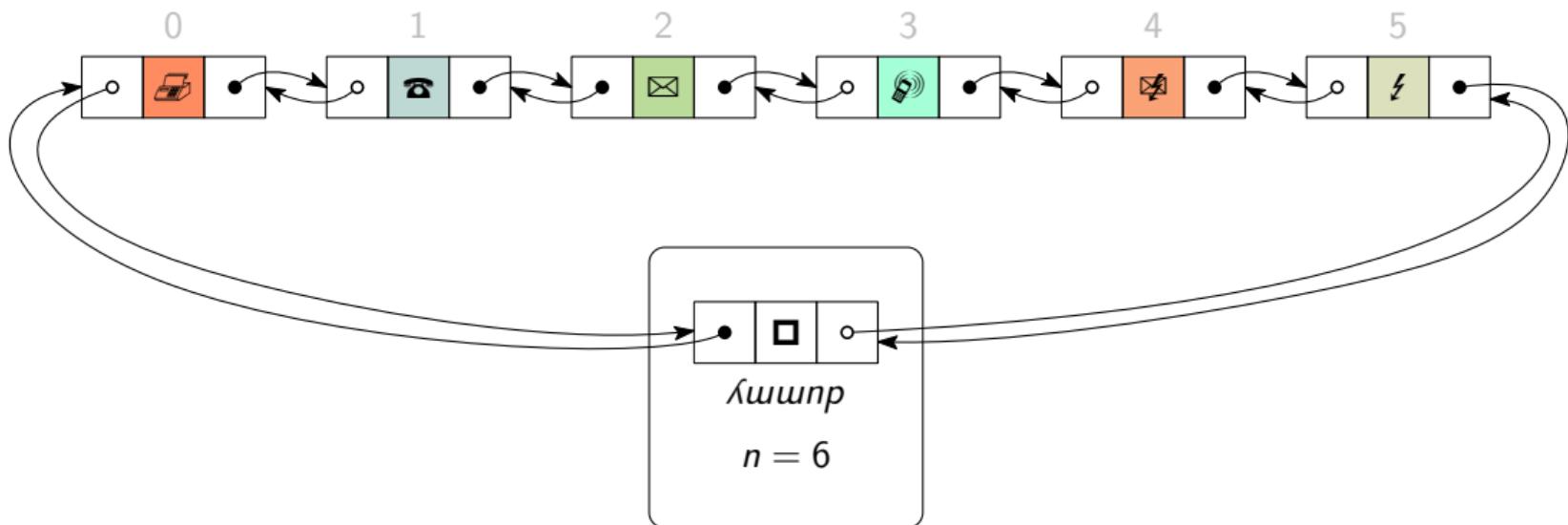


# DLLList

A doubly-linked list

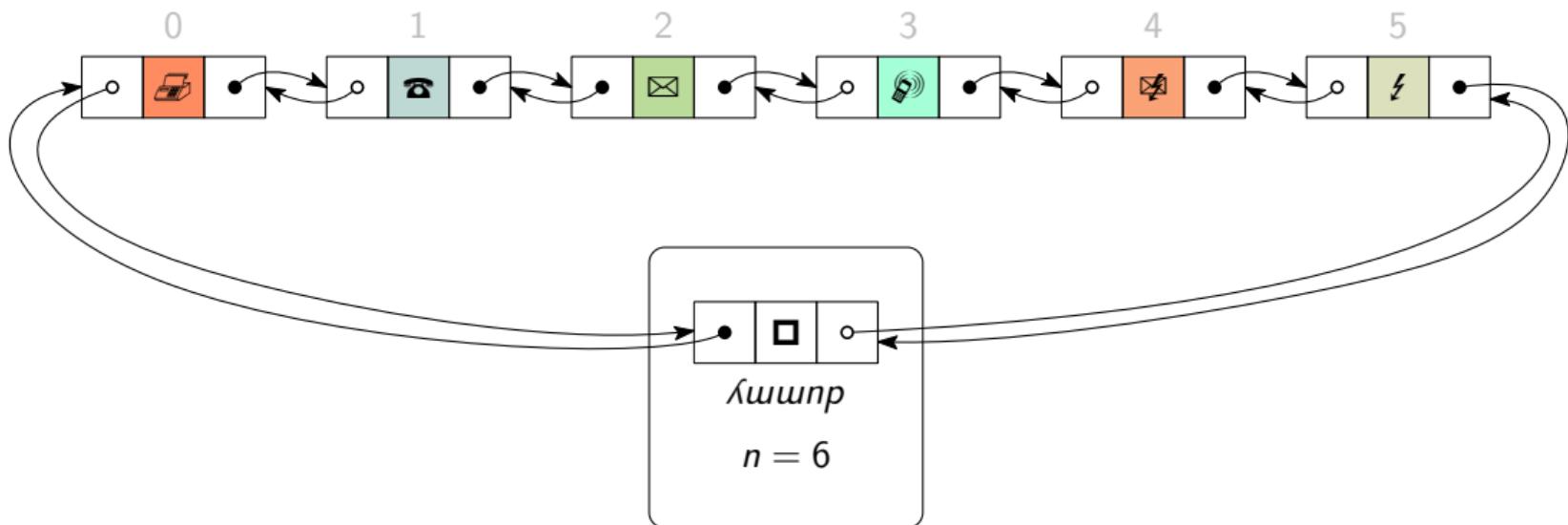
`getNode( $i$ )`

$O(1 + \min\{i, n - i\})$



# DLLList

A doubly-linked list

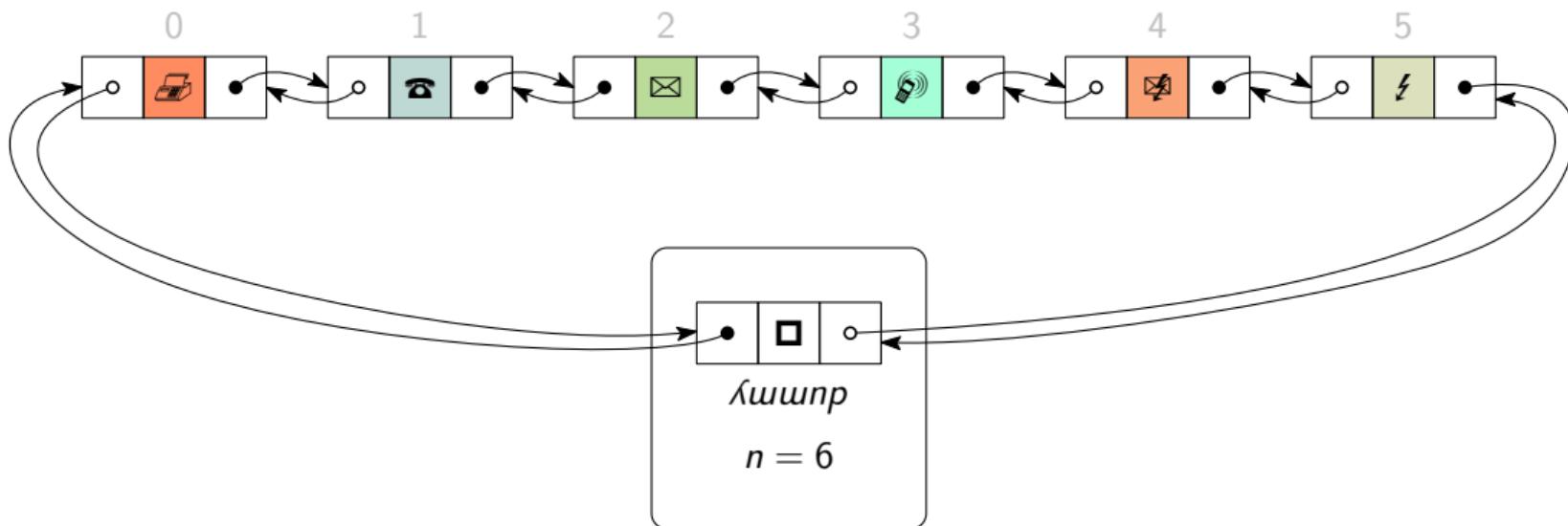


# DLLList

A doubly-linked list

`get(i)`

```
u ← getNode(i)
return u.x
```



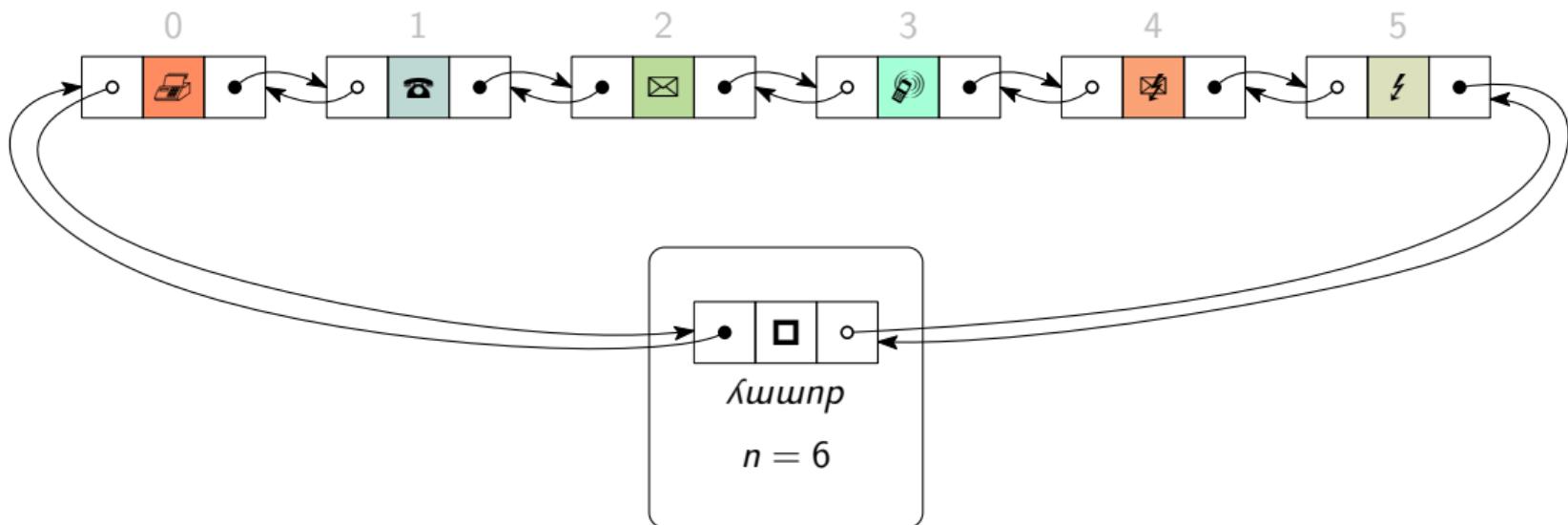
# DLLList

A doubly-linked list

`get(i)`

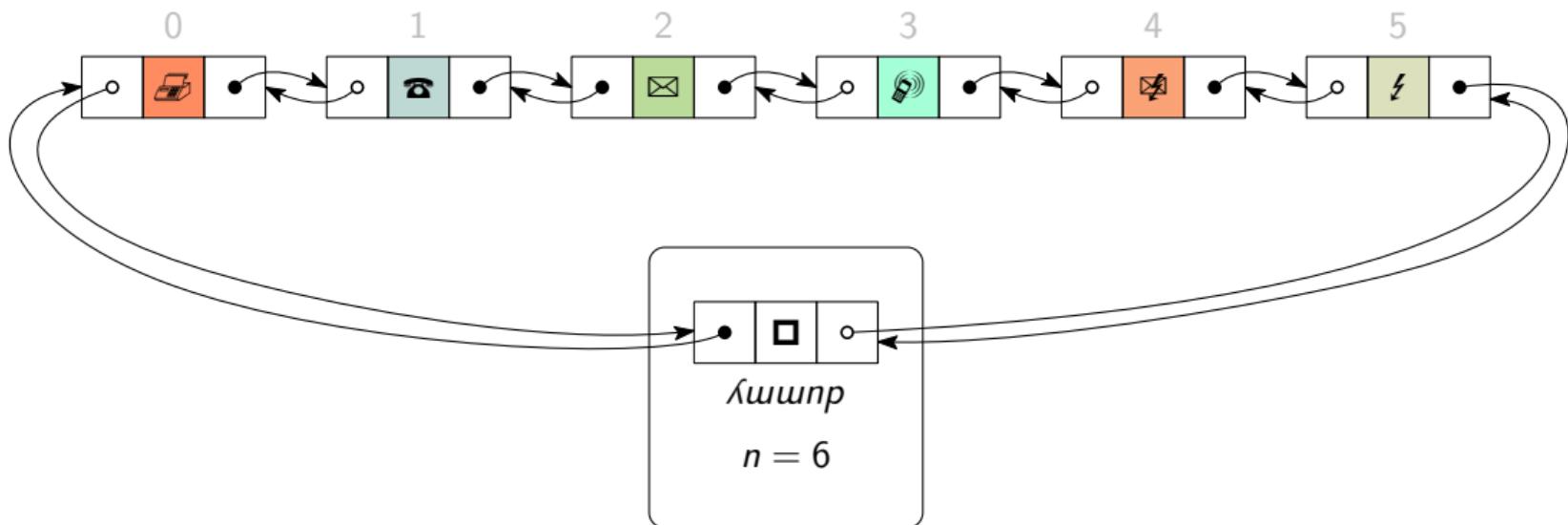
```
u ← getNode(i)
return u.x
```

$O(1 + \min\{i, n - 1\})$



# DLLList

A doubly-linked list

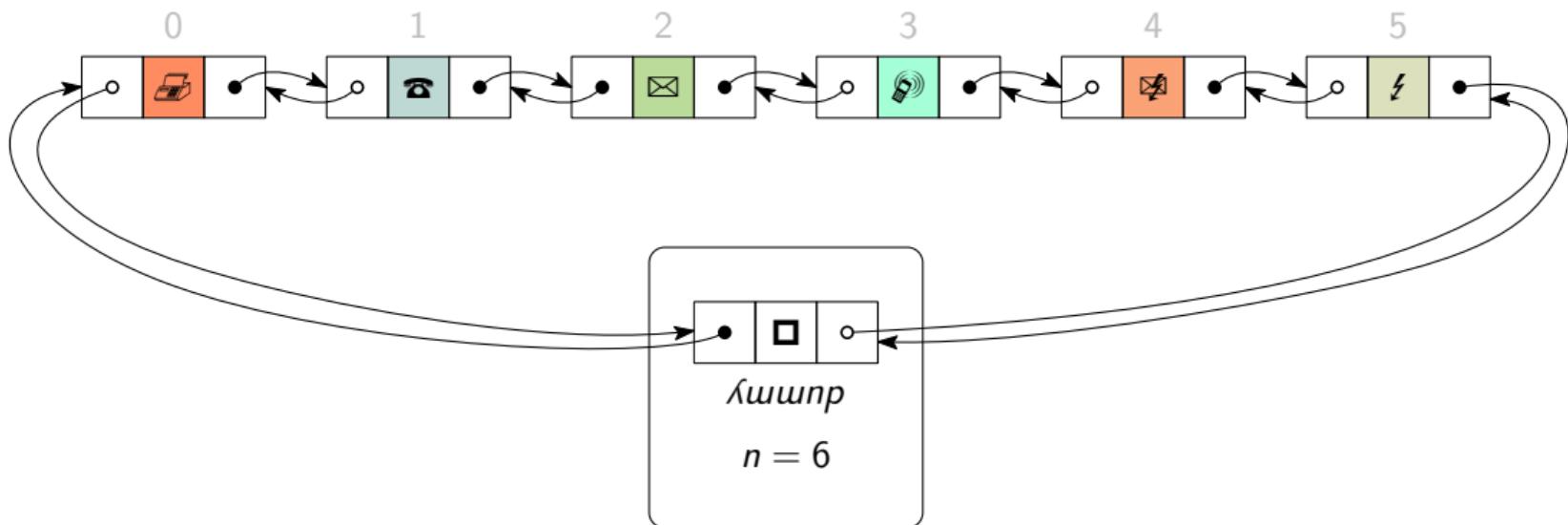


# DLLList

A doubly-linked list

`set( $i, x$ )`

```
 $u \leftarrow \text{getNode}(i)$ 
return  $u.x \leftarrow x$ 
```



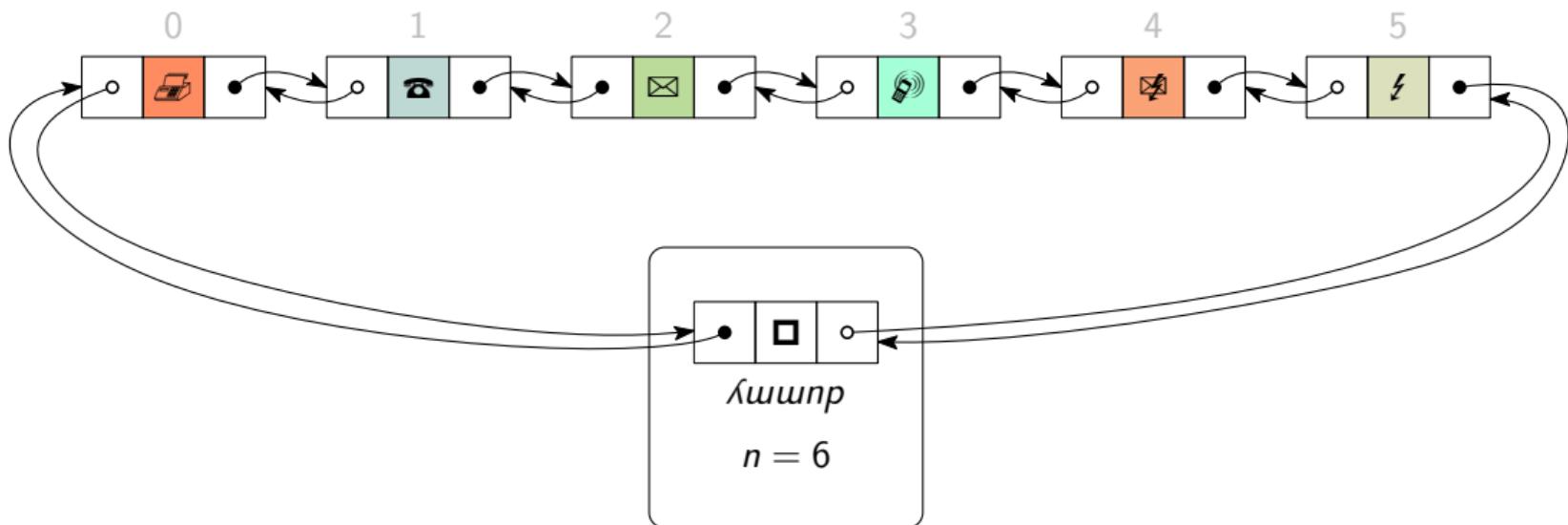
# DLLList

A doubly-linked list

`set( $i, x$ )`

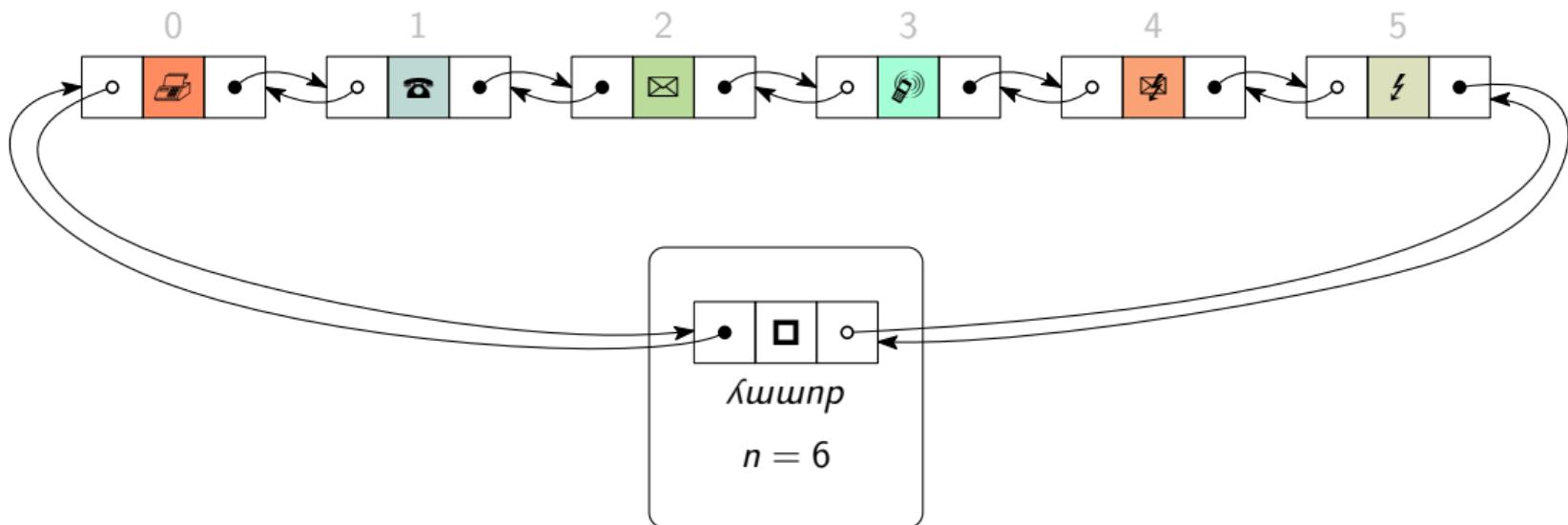
```
 $u \leftarrow \text{getNode}(i)$ 
return  $u.x \leftarrow x$ 
```

$O(1 + \min\{i, n - 1\})$



# DLLList

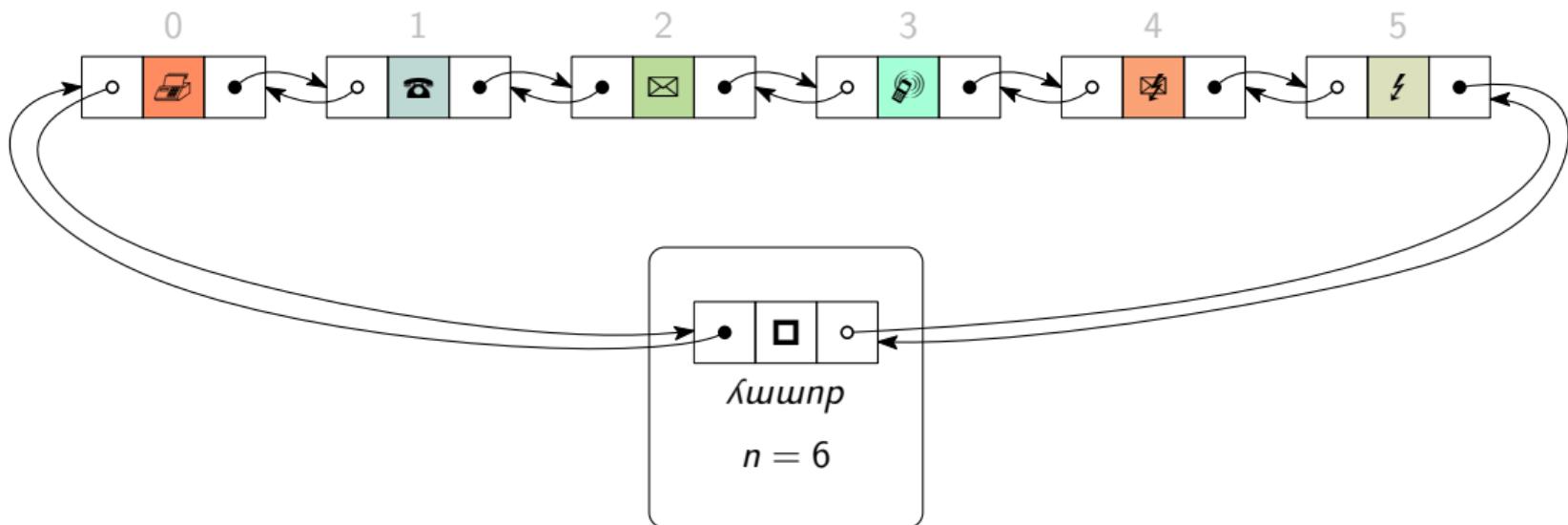
A doubly-linked list



# DLLList

A doubly-linked list

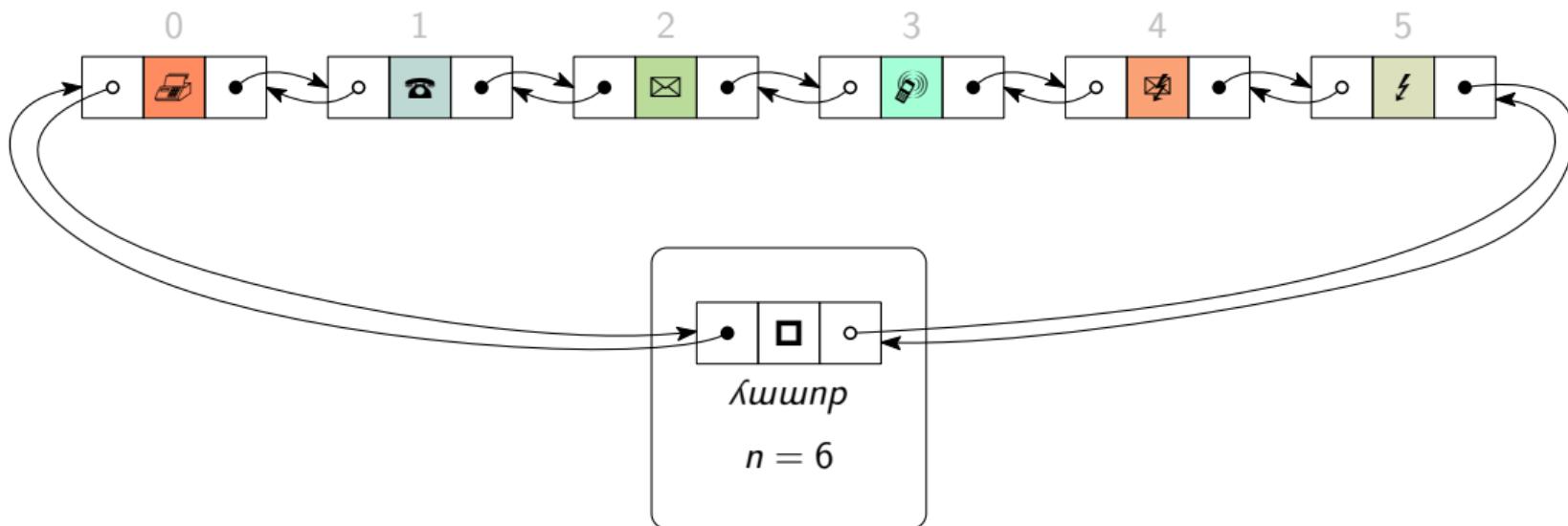
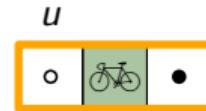
`add(4, )`



# DLLList

A doubly-linked list

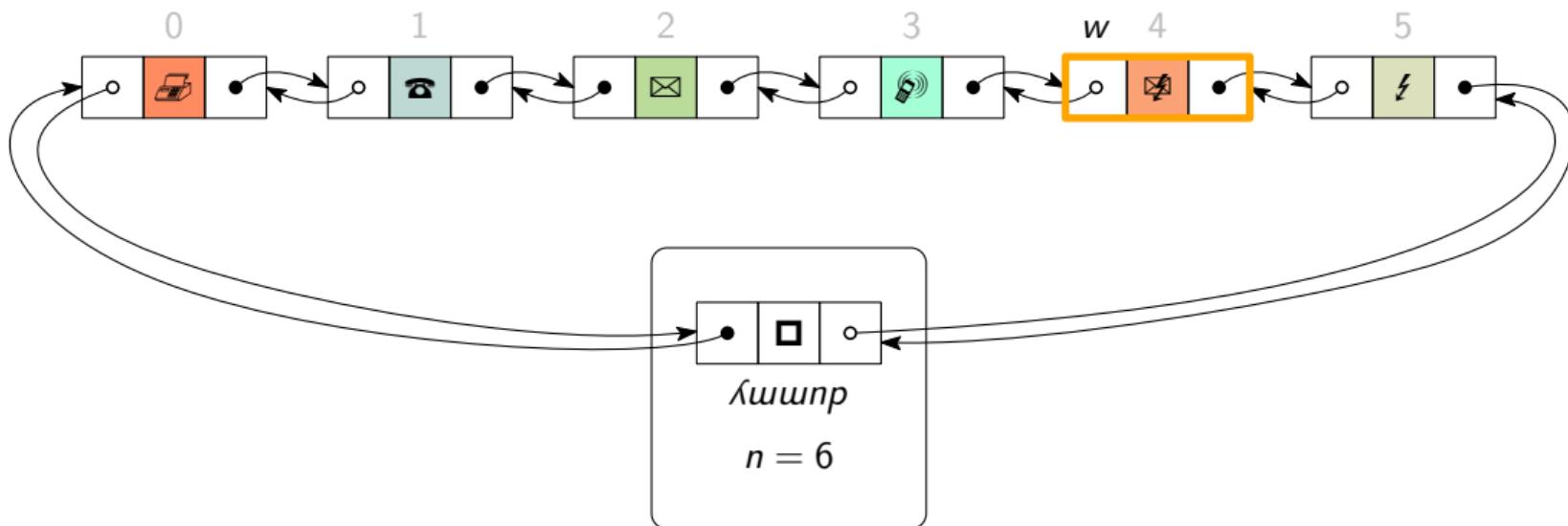
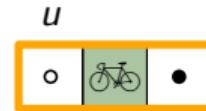
`add(4, )`



# DLLList

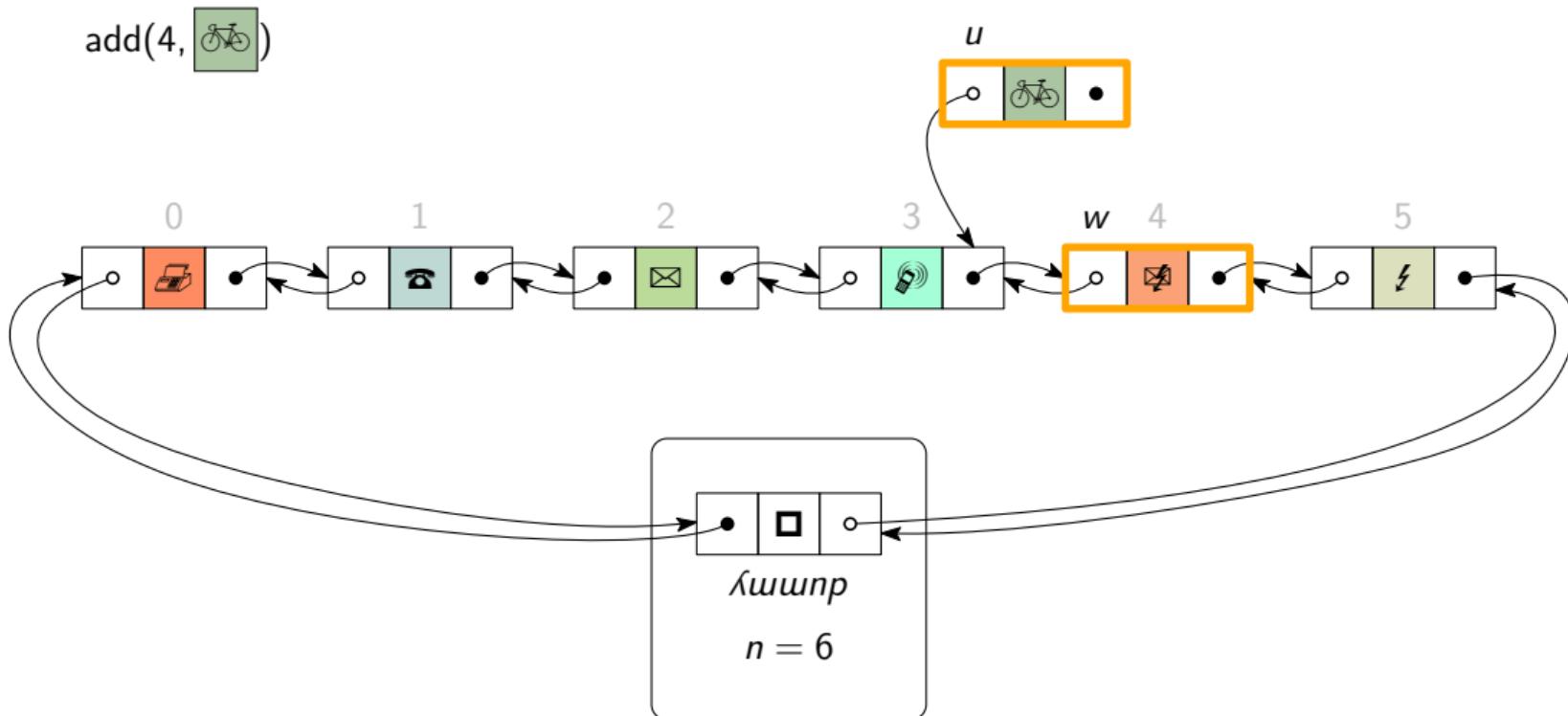
A doubly-linked list

`add(4, )`



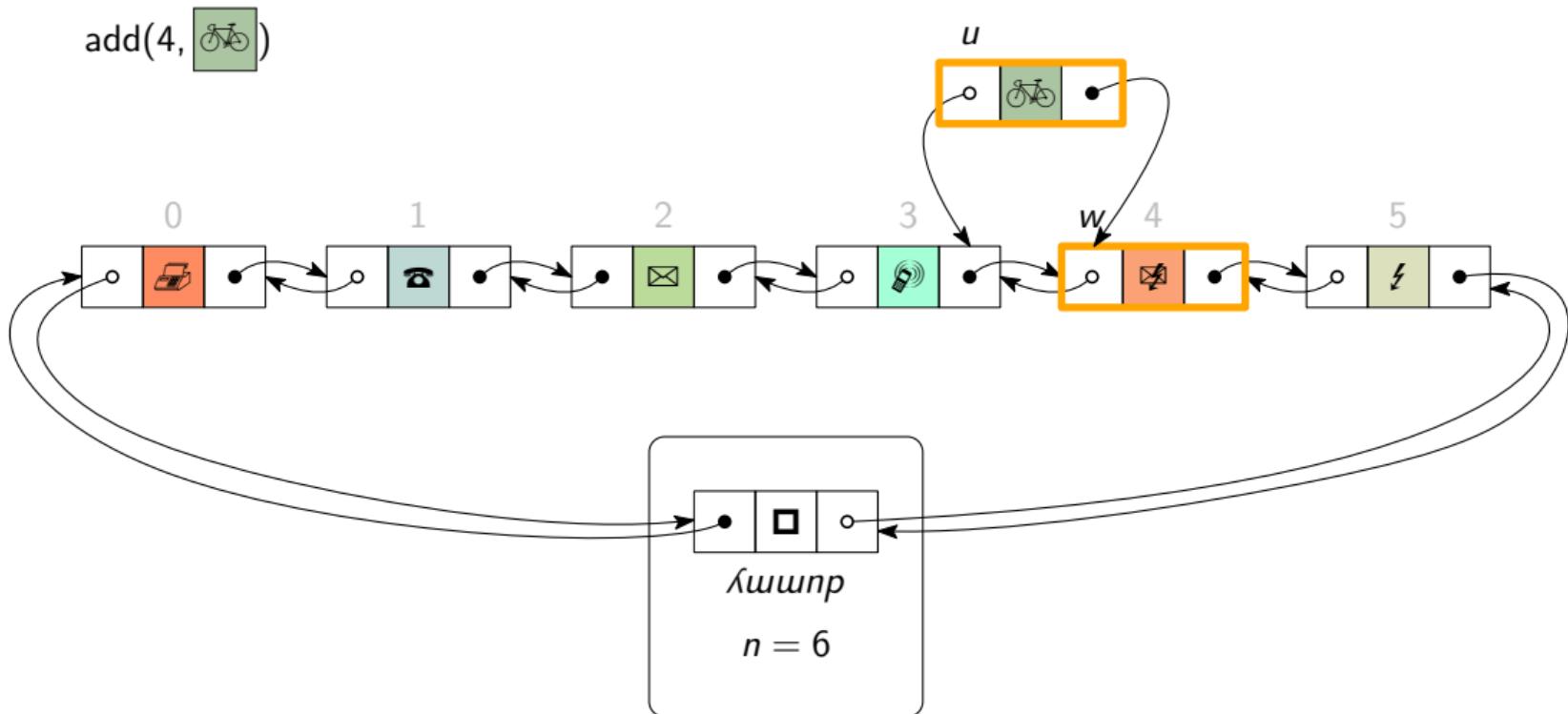
# DLLList

A doubly-linked list



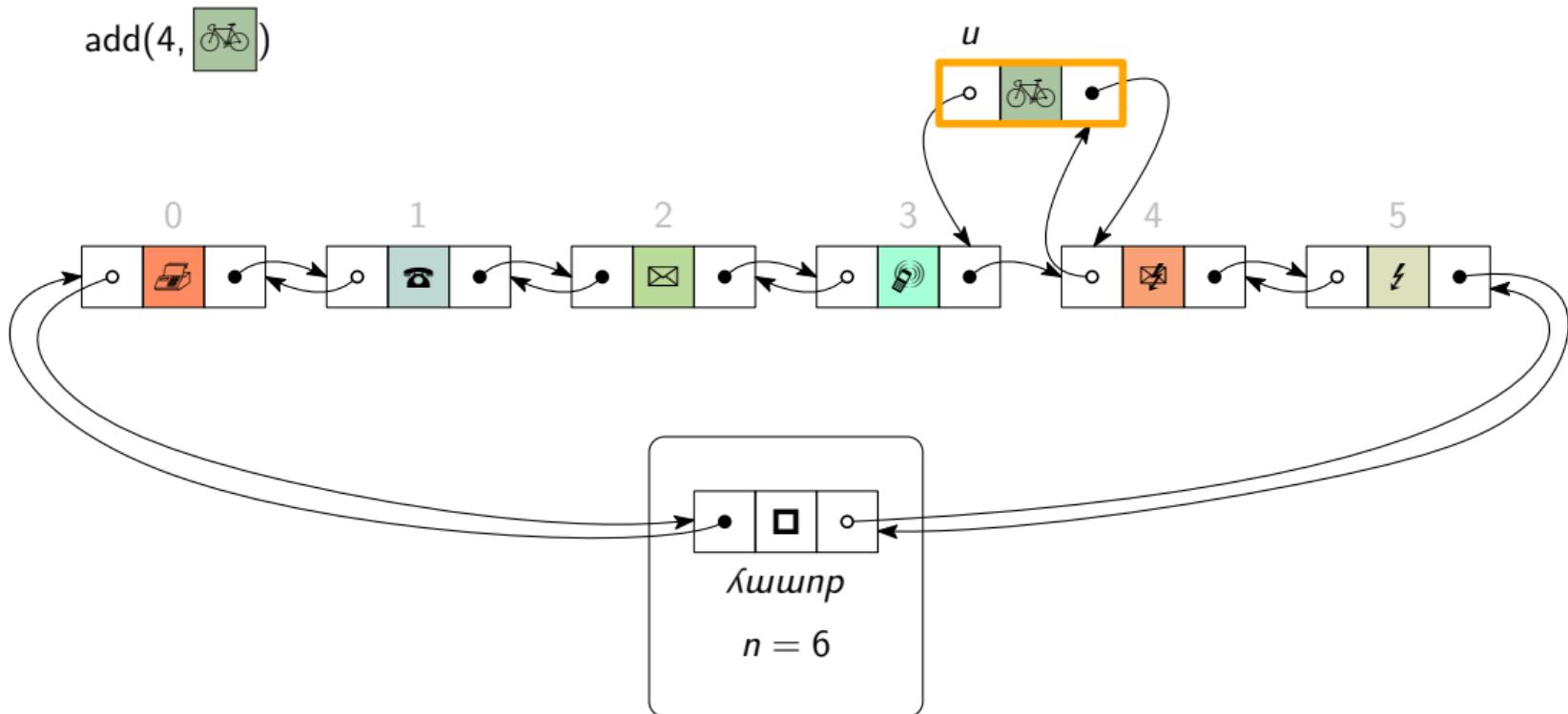
# DLLList

A doubly-linked list



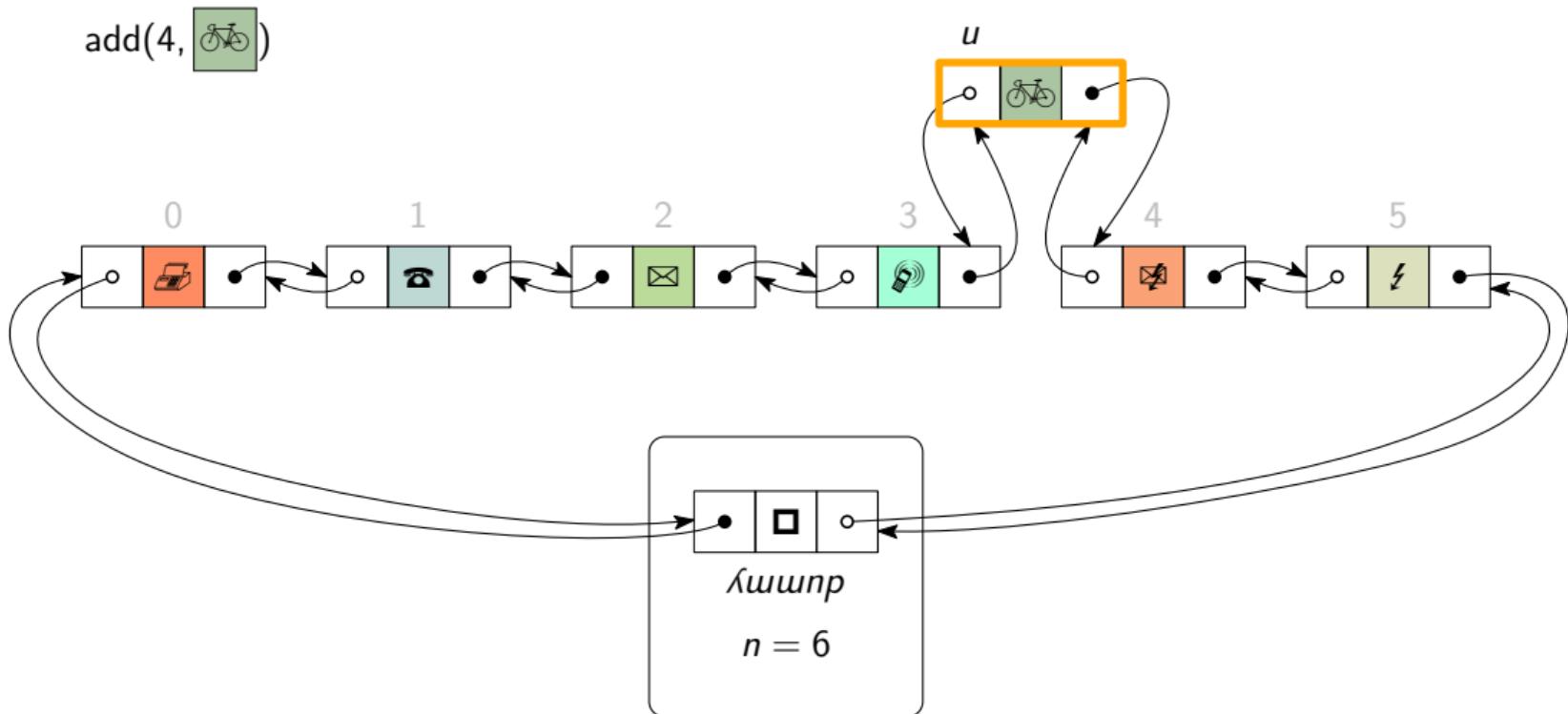
# DLLList

A doubly-linked list



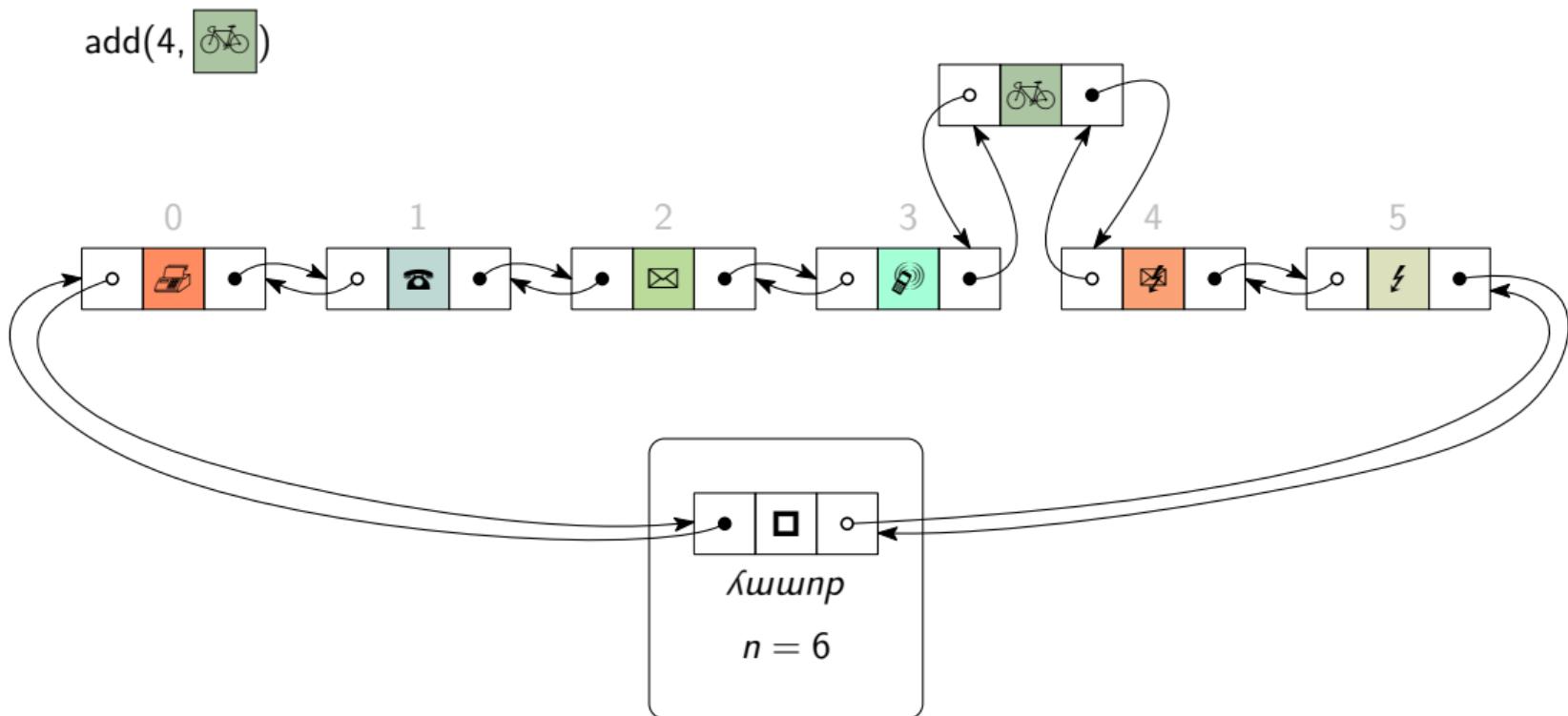
# DLLList

A doubly-linked list



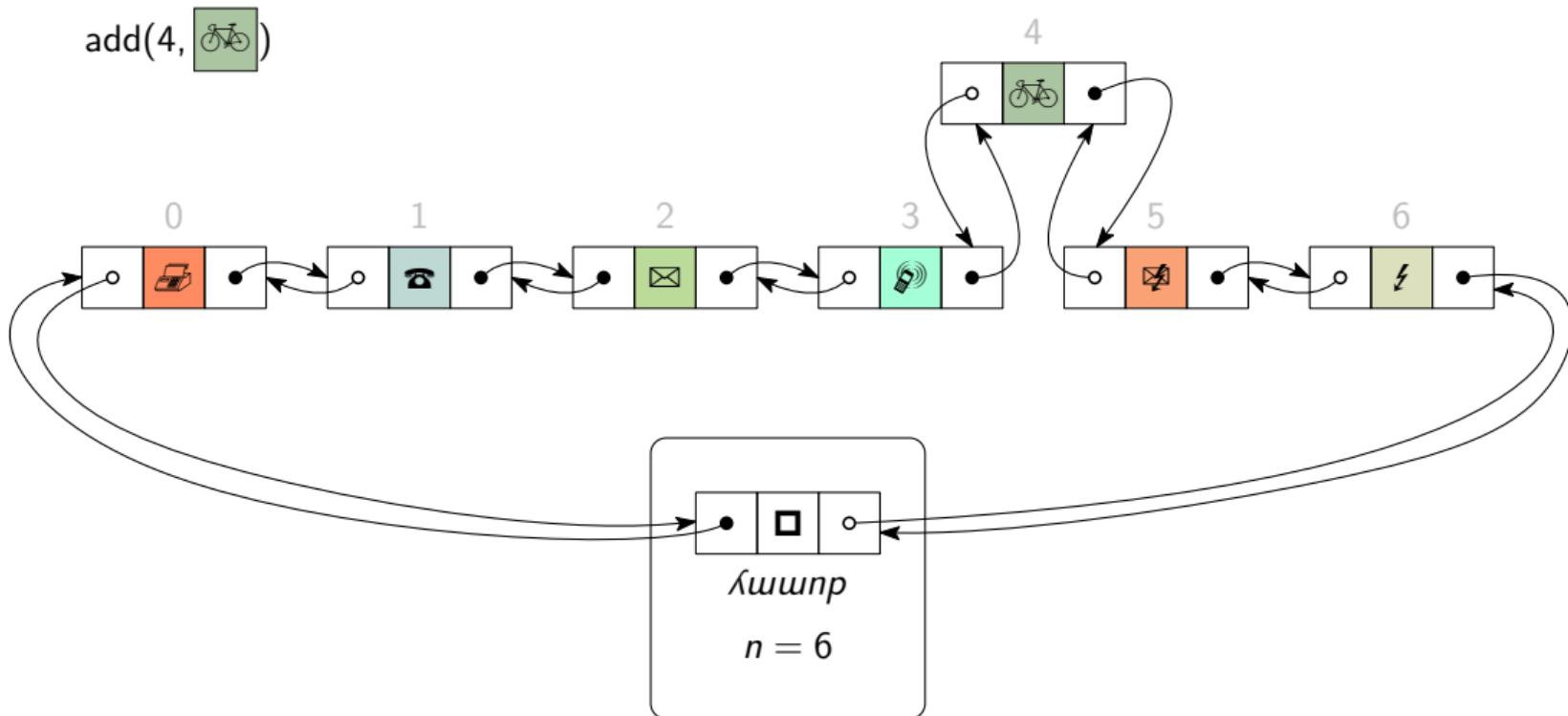
# DLLList

A doubly-linked list



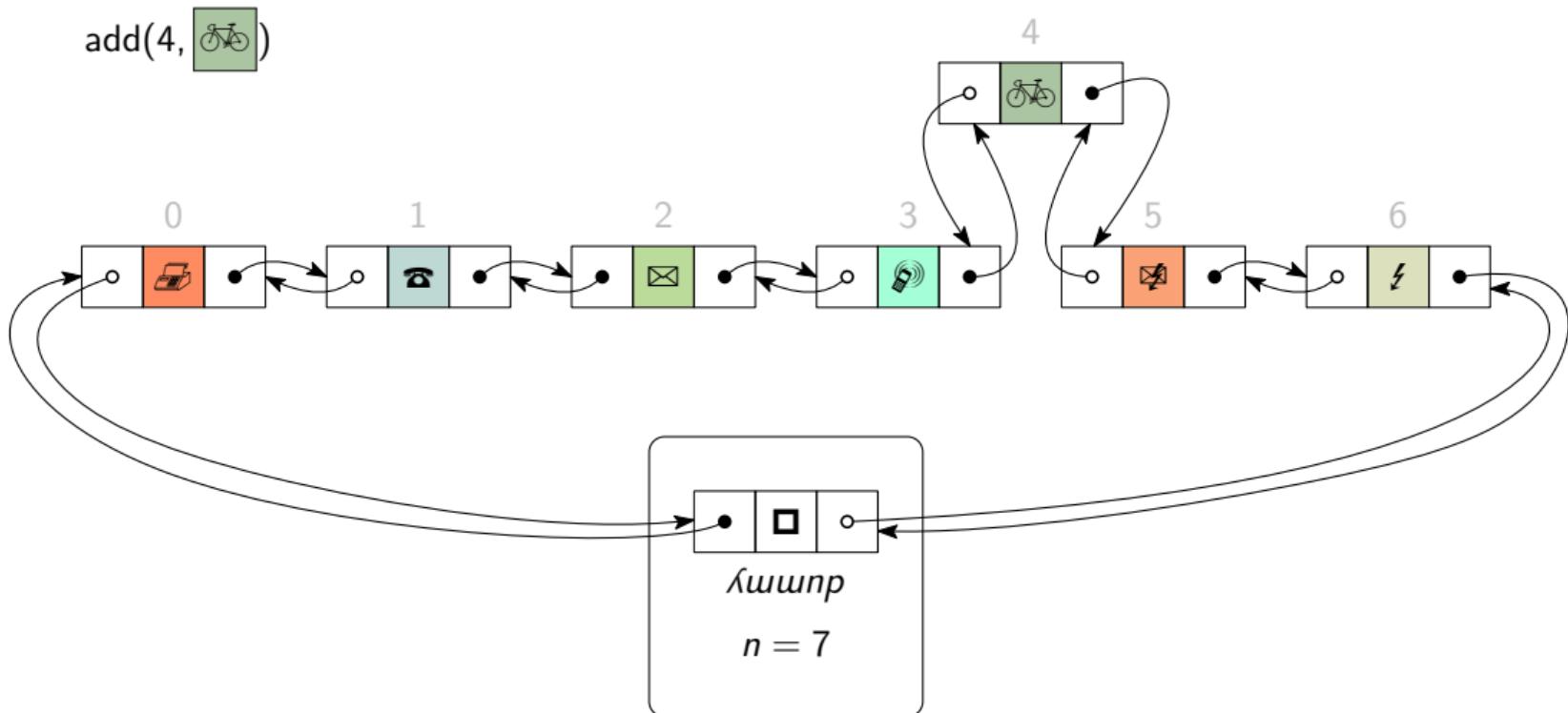
# DLLList

A doubly-linked list



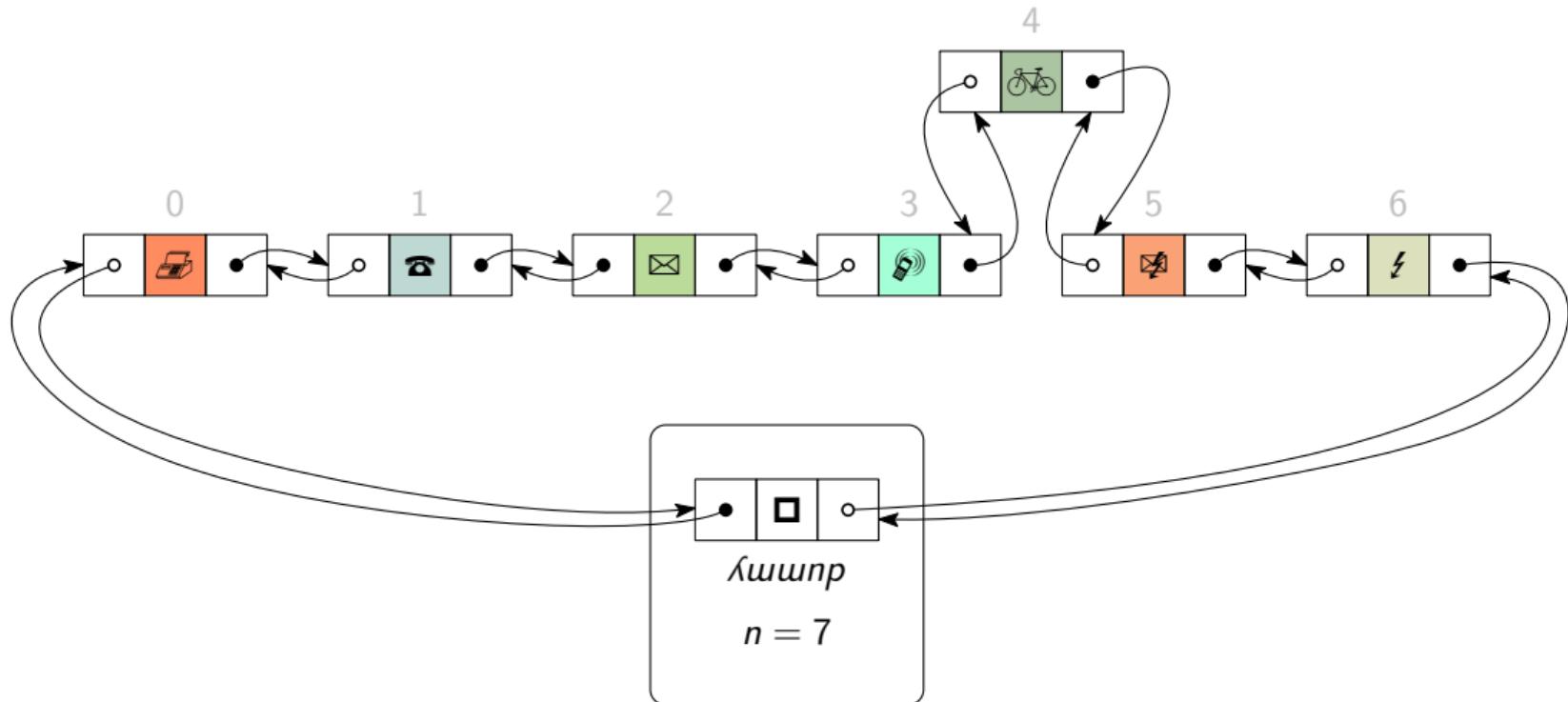
# DLLList

A doubly-linked list



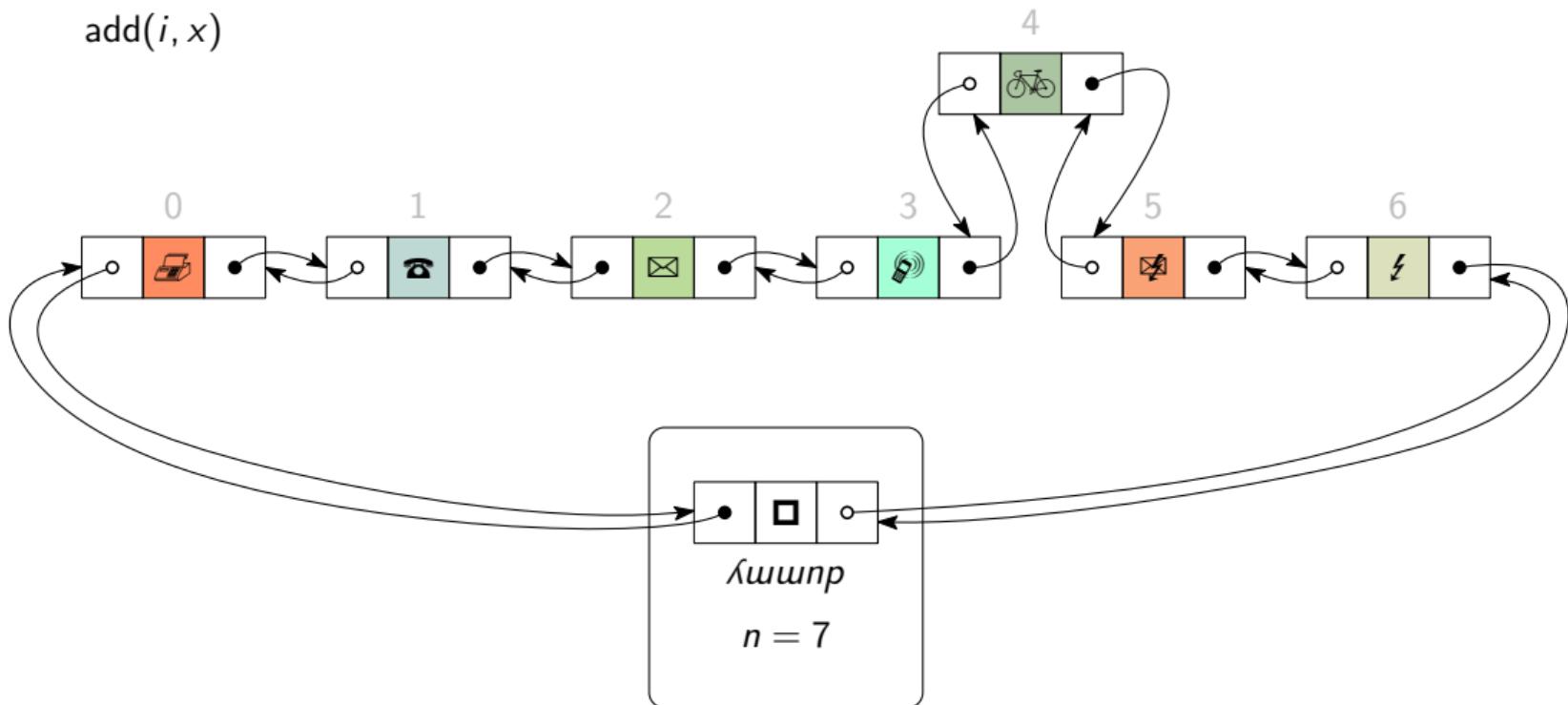
# DLLList

A doubly-linked list



# DLLList

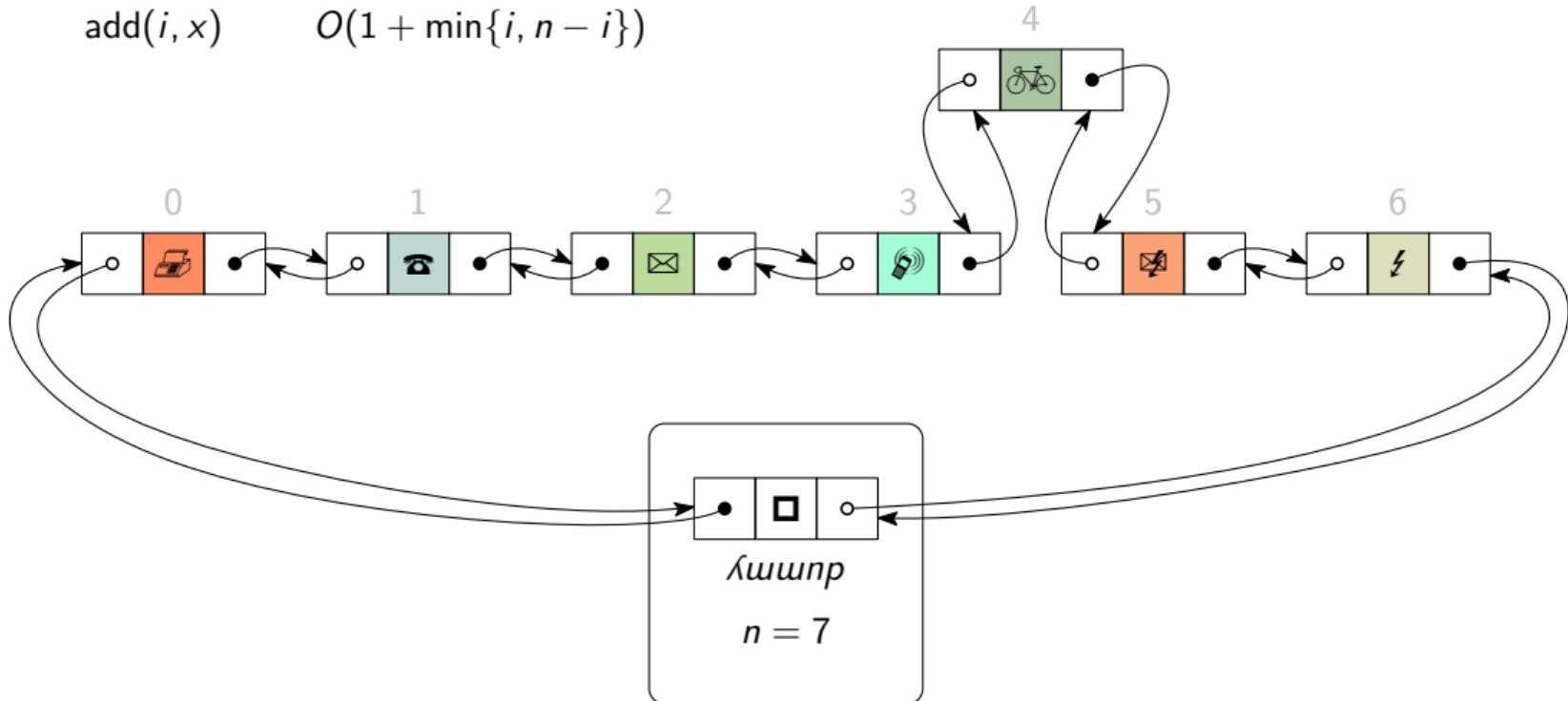
A doubly-linked list



# DLLList

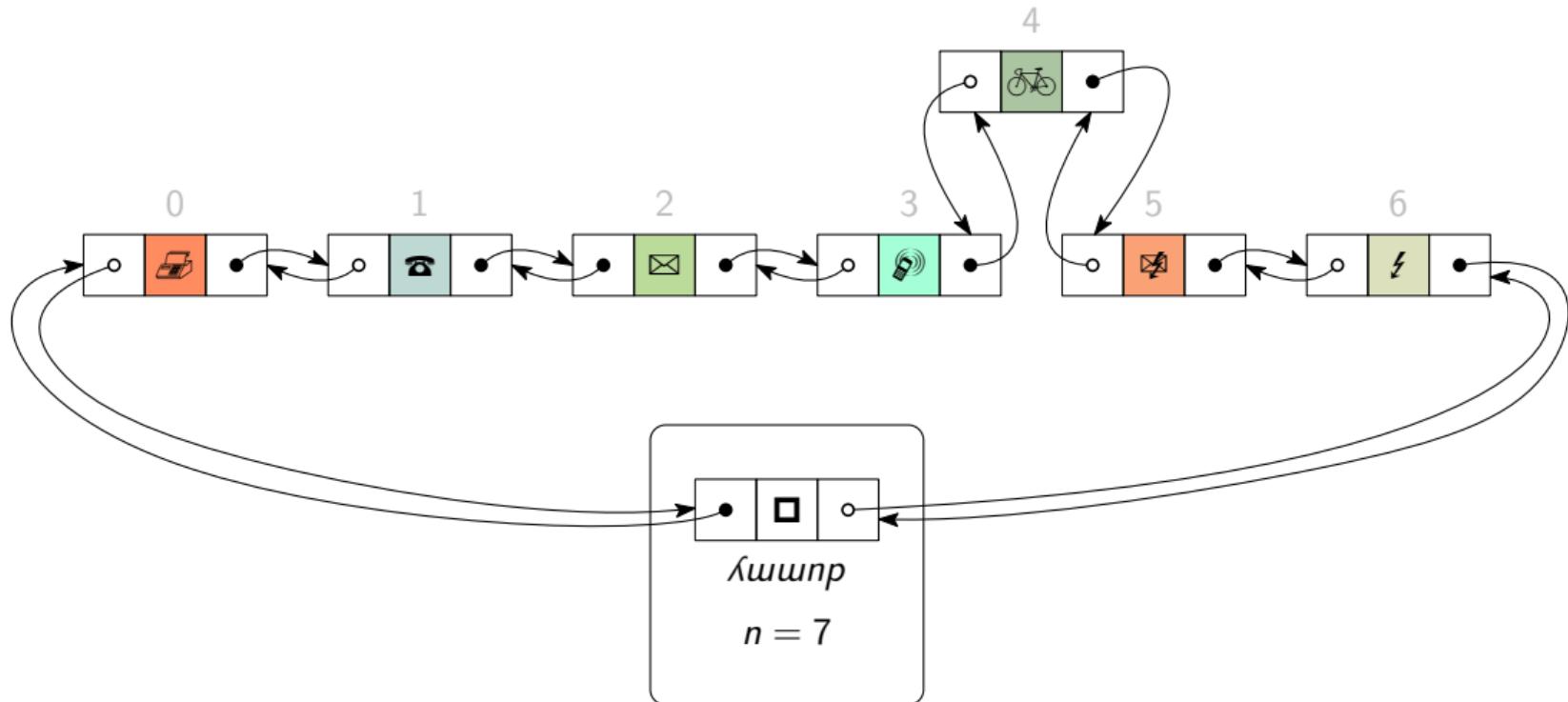
A doubly-linked list

$\text{add}(i, x)$        $O(1 + \min\{i, n - i\})$



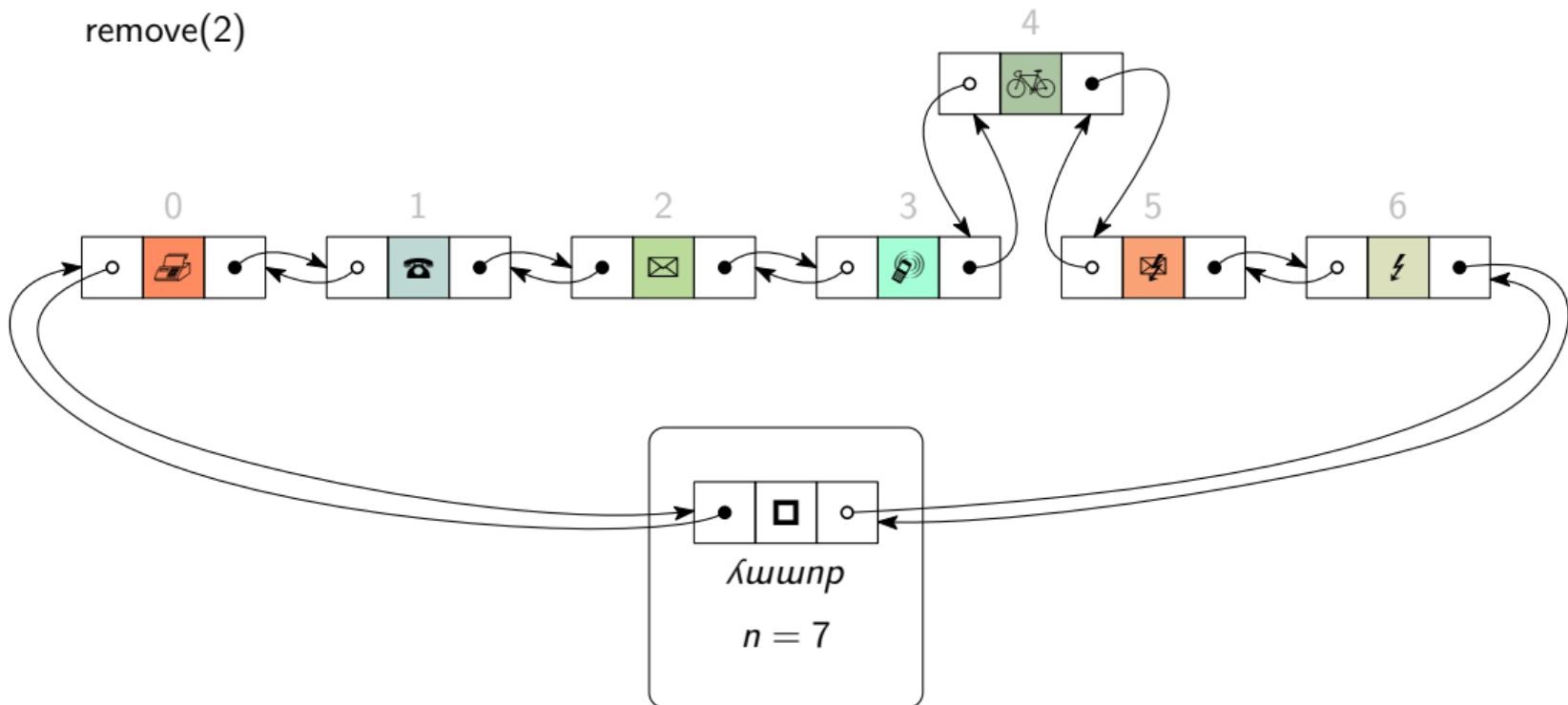
# DLLList

A doubly-linked list



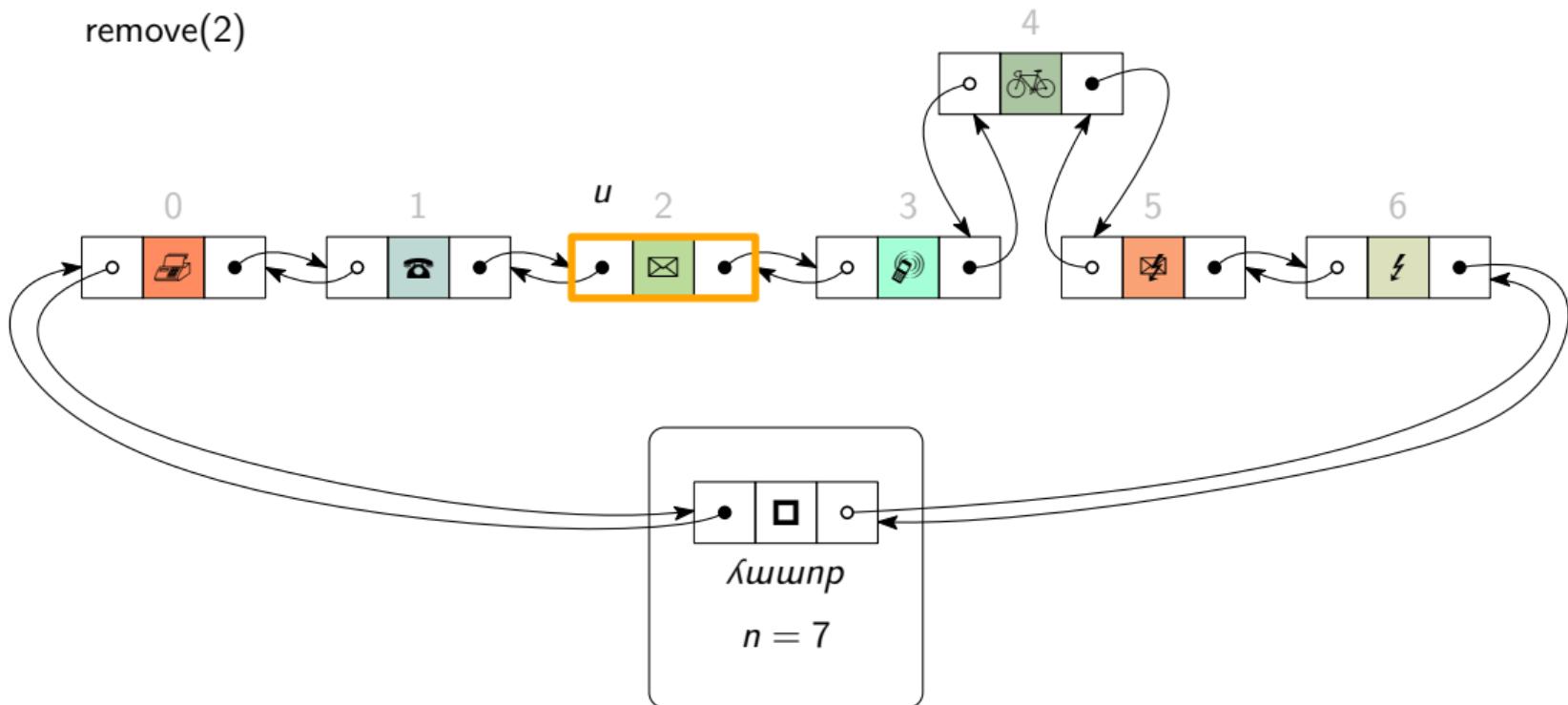
# DLLList

A doubly-linked list



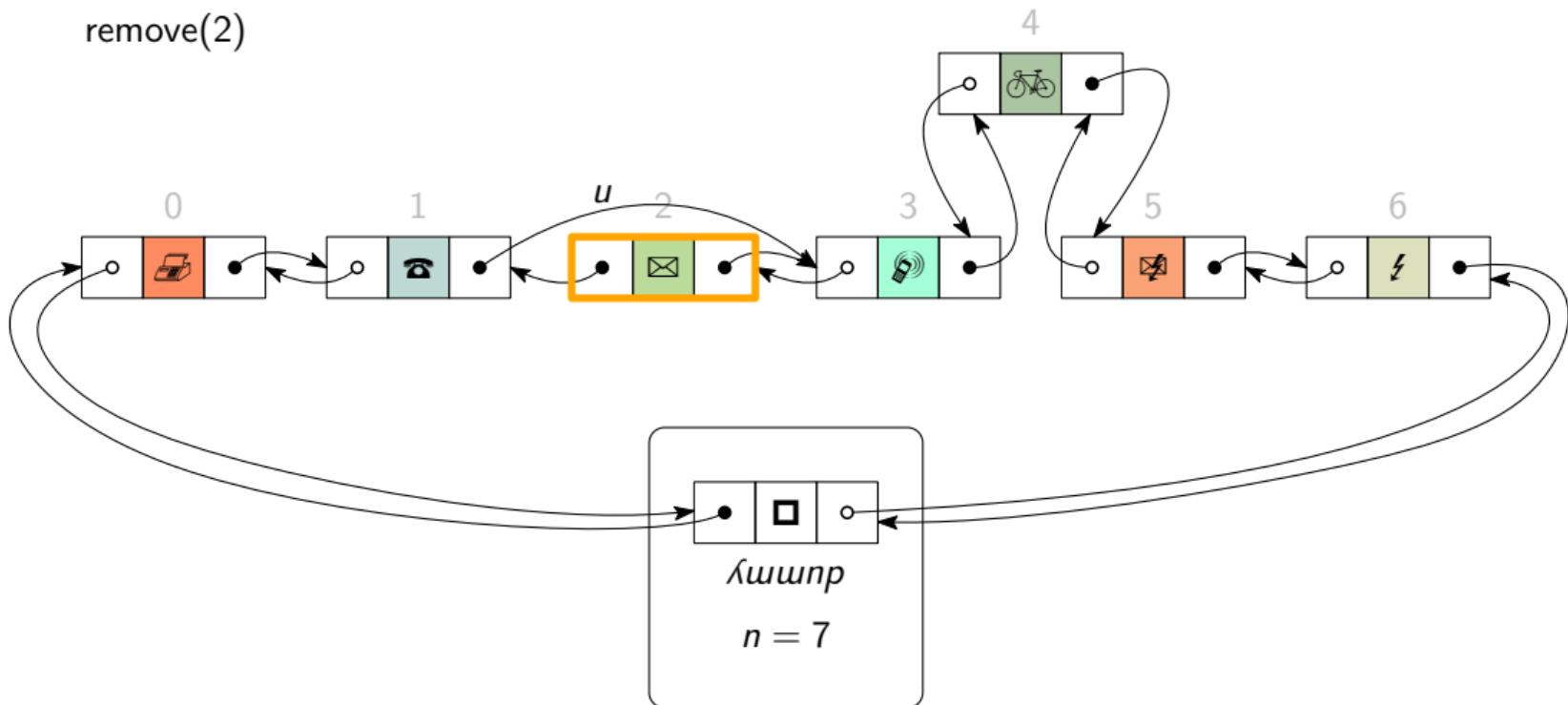
# DLLList

A doubly-linked list



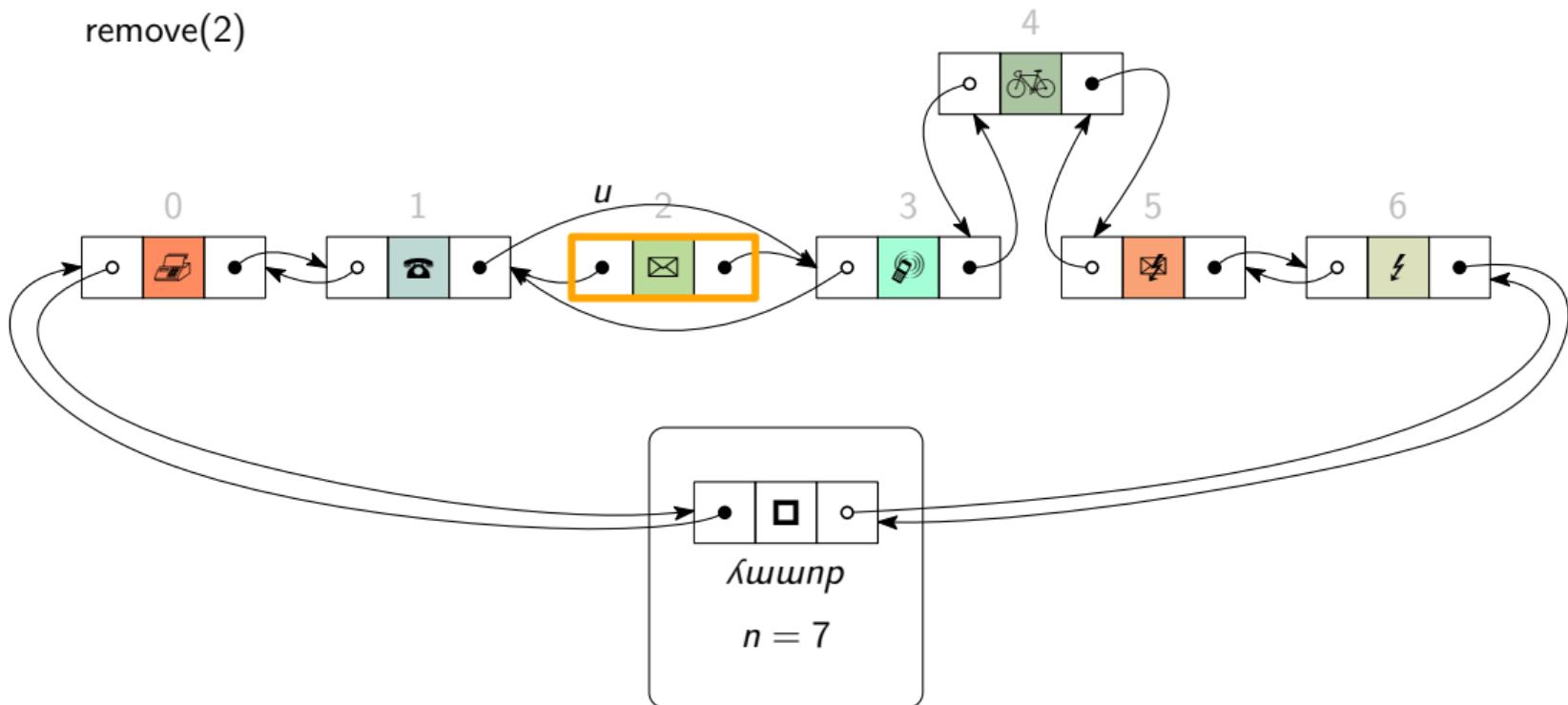
# DLLList

A doubly-linked list



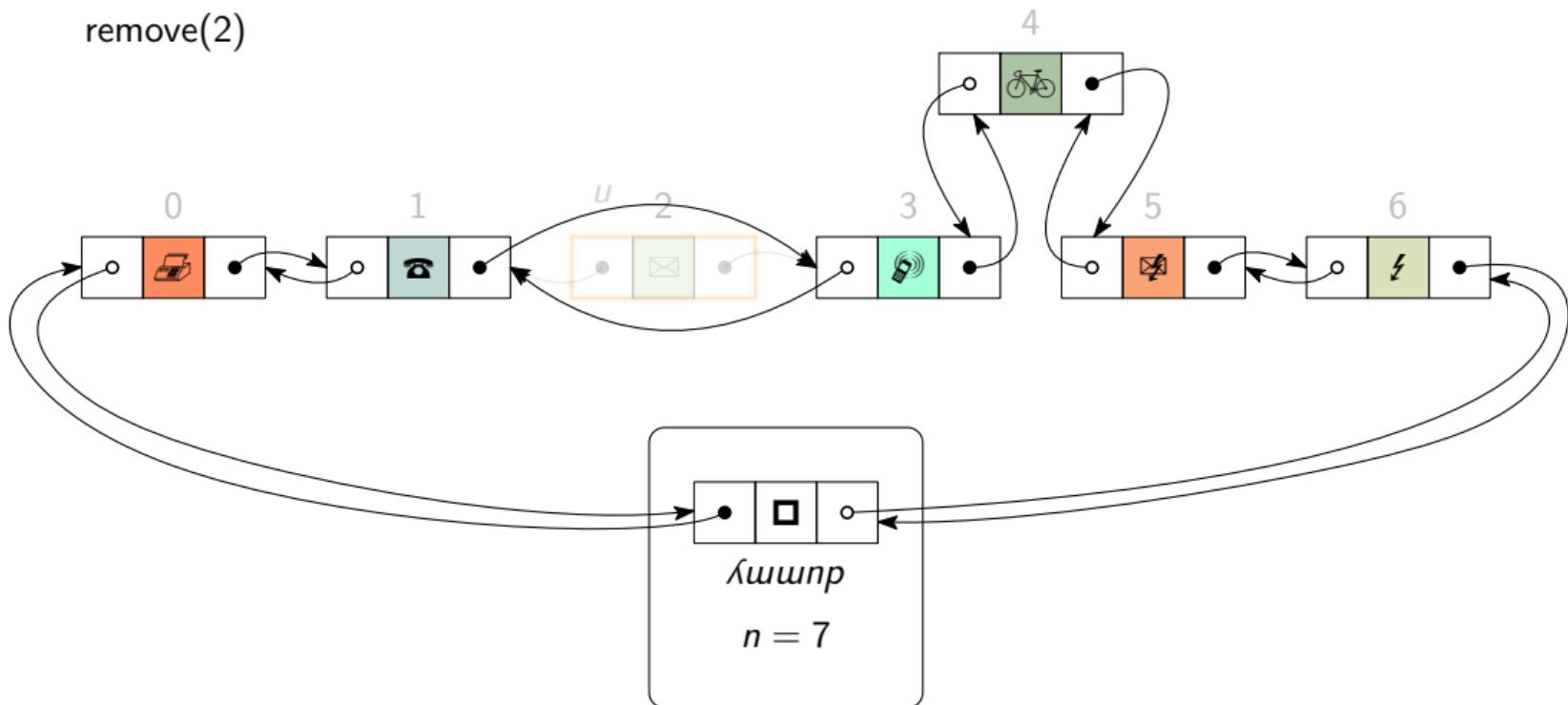
# DLLList

A doubly-linked list



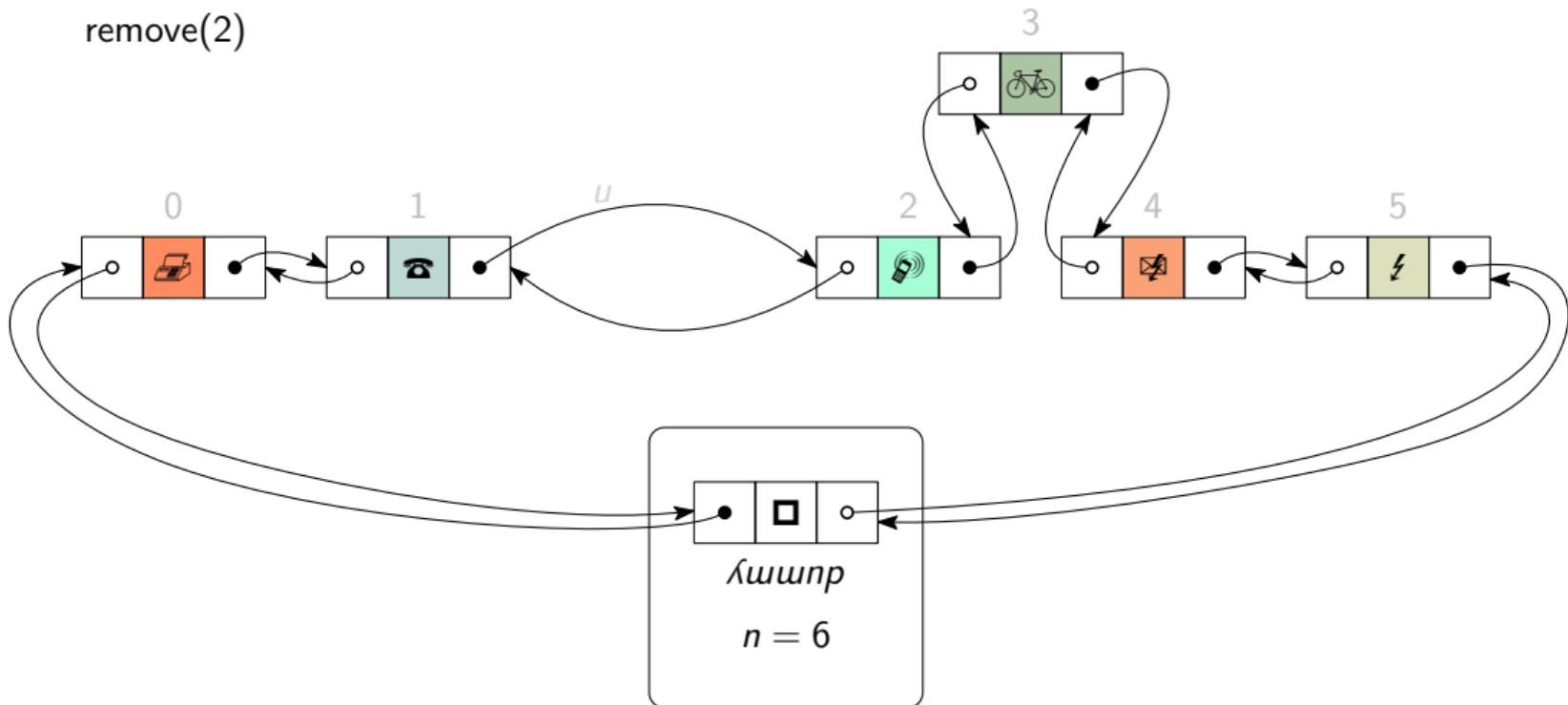
# DLLList

A doubly-linked list



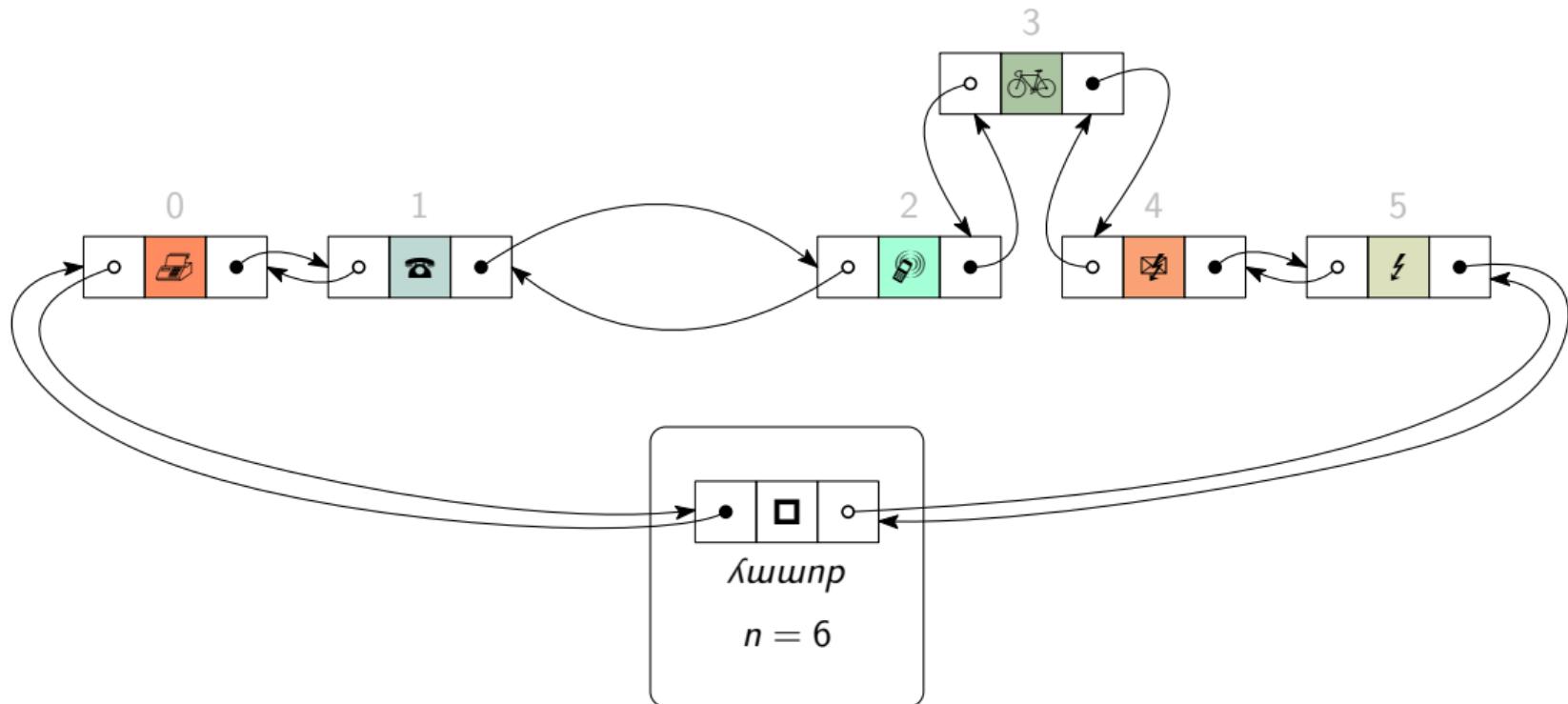
# DLLList

A doubly-linked list



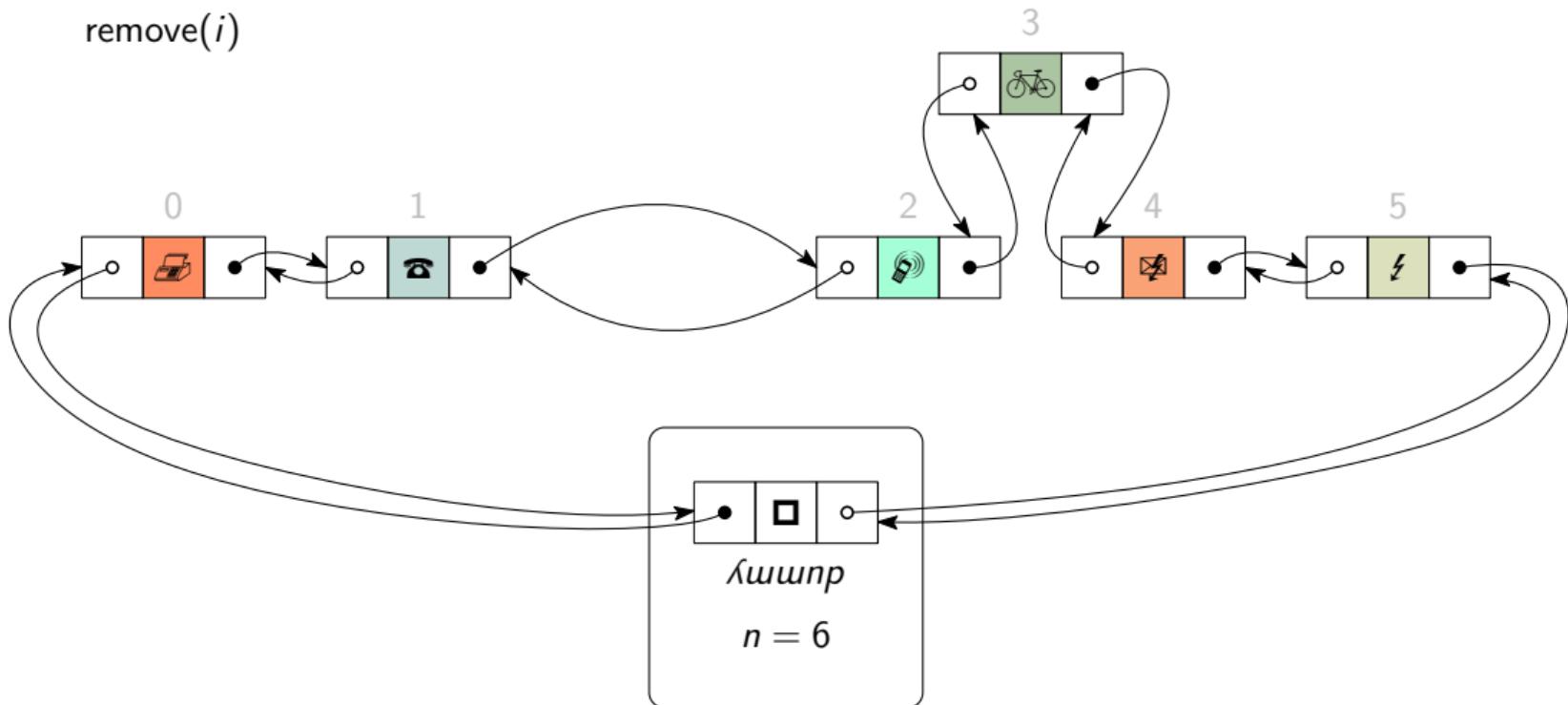
# DLLList

A doubly-linked list



# DLLList

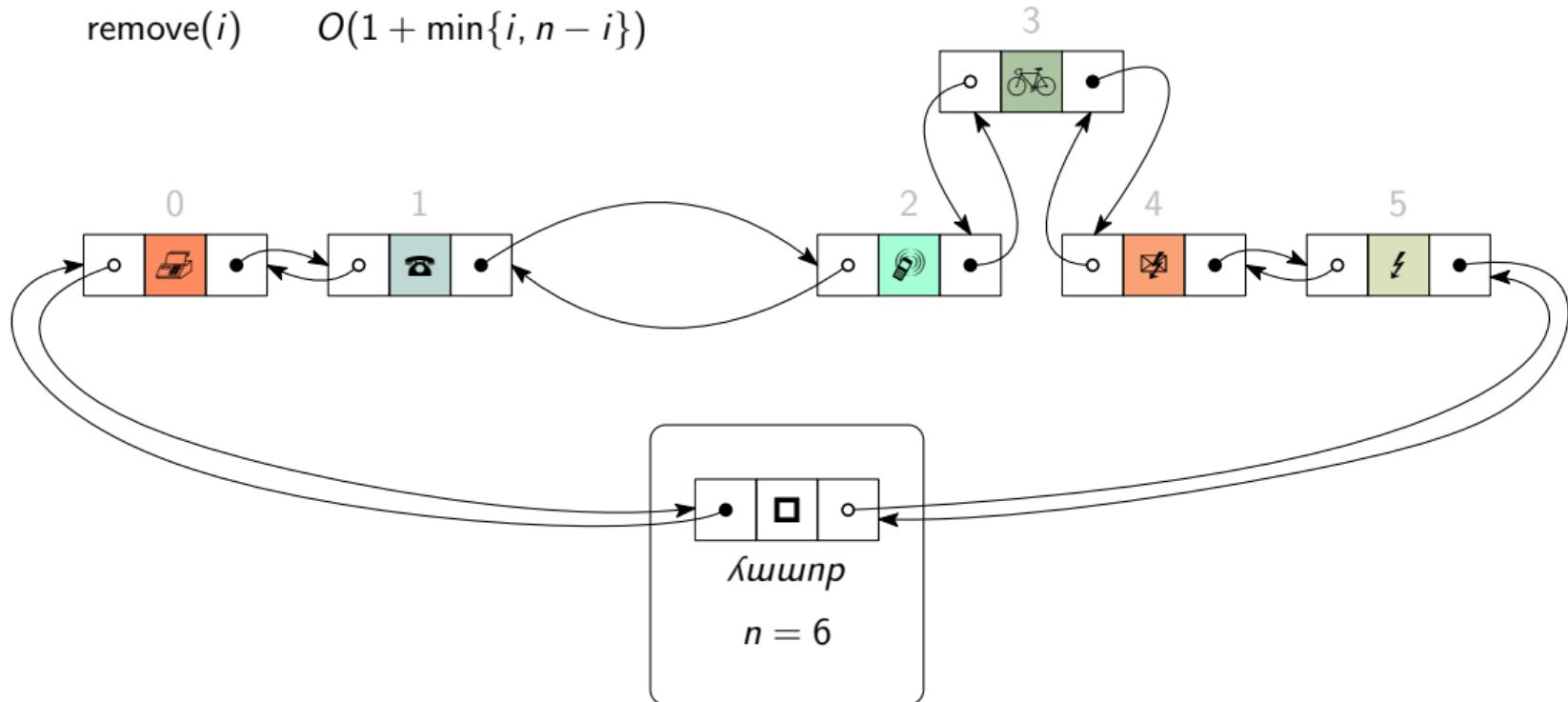
A doubly-linked list



# DLLList

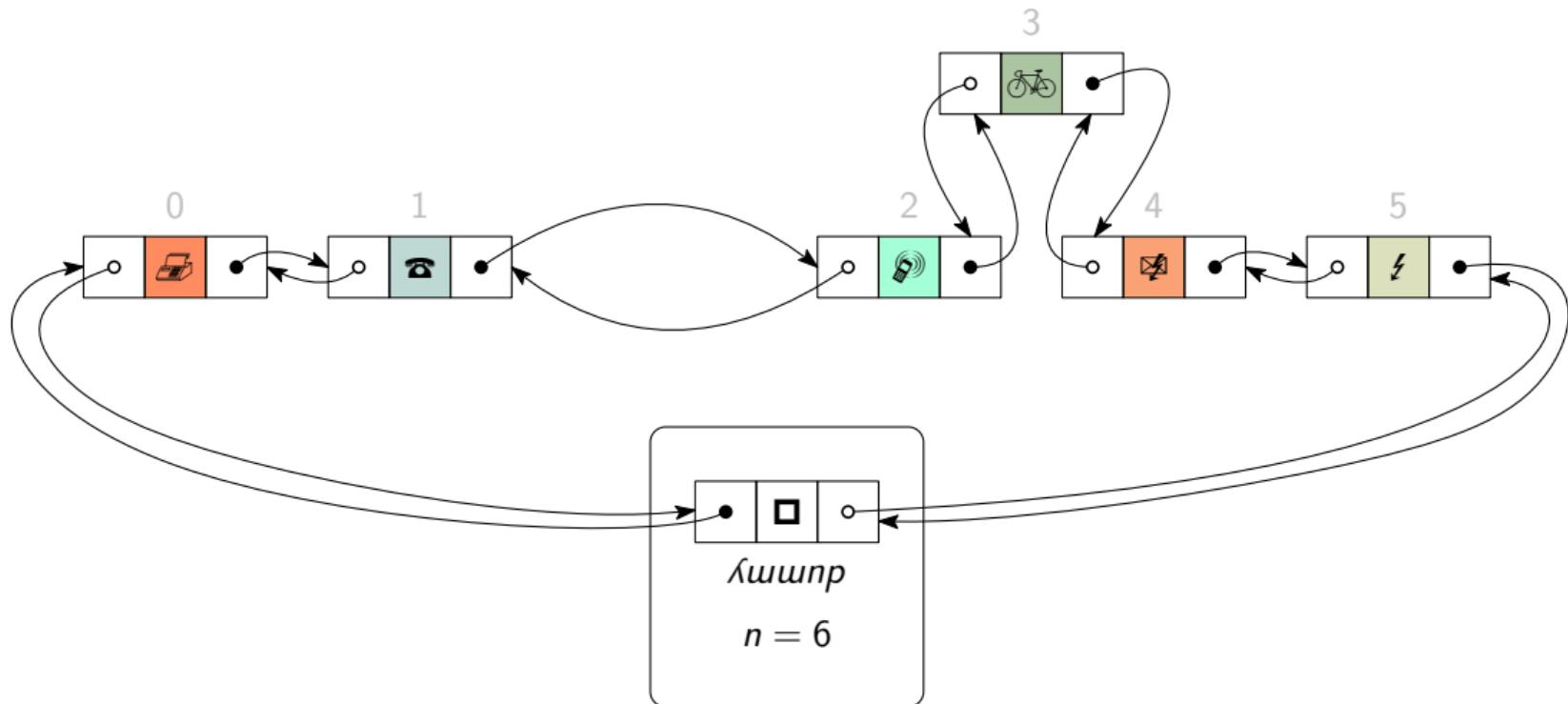
A doubly-linked list

$\text{remove}(i) \quad O(1 + \min\{i, n - i\})$



# DLLList

A doubly-linked list



# DLLList

A doubly-linked list

## Theorem

A DLLList implements the List and Deque interfaces.

# DLList

A doubly-linked list

## Theorem

A DLList implements the List and Deque interfaces.

- The List operations  $\text{get}(i)$ ,  $\text{set}(i, x)$ ,  $\text{add}(i, x)$  and  $\text{remove}(i)$  operations each run in  $O(1 + \min\{i, n - i\})$  time.

# DLLList

A doubly-linked list

## Theorem

A DLLList implements the List and Deque interfaces.

- The List operations  $\text{get}(i)$ ,  $\text{set}(i, x)$ ,  $\text{add}(i, x)$  and  $\text{remove}(i)$  operations each run in  $O(1 + \min\{i, n - i\})$  time.
- The Deque operations  $\text{addFront}(x)$ ,  $\text{removeFront}()$ ,  $\text{addBack}(x)$  and  $\text{removeBack}()$  operations each run in  $O(1)$  time.

# End of Lesson