

Linked Lists

```
class Node {  
    int data;  
    Node next;  
}  
  
Node head;
```

2021
Spring
Semester



Simple one-way lists (S1L)

$L \rightarrow [5] \rightarrow [2] \rightarrow [7] \rightarrow [3] \square$ } (Empty list: $L = \emptyset$)

$A : N[5] = [5|2|7|3]$ } ALTERNATIVE REPRESENTATIONS
OF THE ABSTRACT SEQUENCE

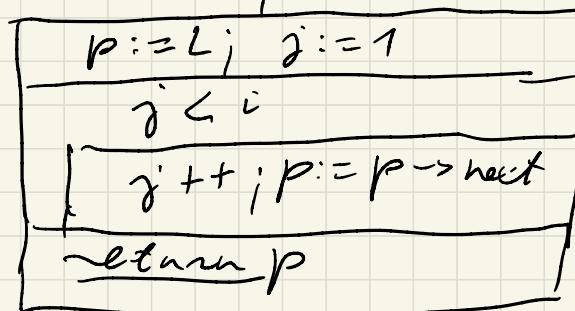
LINKED REPRESENTATION

SEQUENTIAL REPR. $\{5, 2, 7, 3\}$

$A[i]$ - refers directly to the i th element : $\Theta(1)$ time

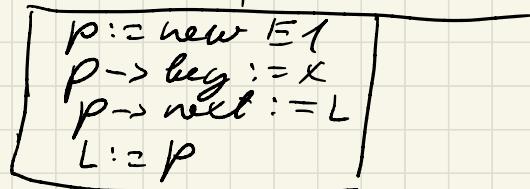
$L[i]$ is not possible.

(element($L : E1^*$; $i : N$) : $E1^*$)



$\Theta(i)$ time

(insert-at-front($L : E1^*$; $x : N$)



$q := \text{element}(L, i); \text{print}(q \rightarrow \text{begin}) \quad x$

delete_first(&L: E1*)

```

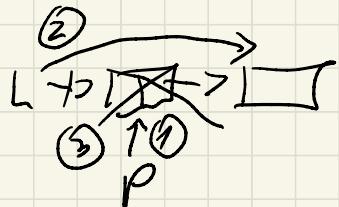
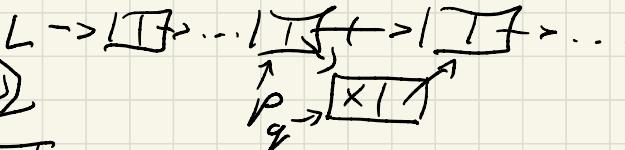
    P := L
    L := L -> next
    delete p
  
```

$L \neq \emptyset$

delete_next(p: E1*)

```

    q := p -> next
    p -> next := q -> next
    delete q
  
```



MEMORY
LEAKING,
if forgotten

P



follow(p: E1*; x: N)

```

    q := new E1
    q -> key := x
    q -> next := p -> next
    p -> next := q
  
```

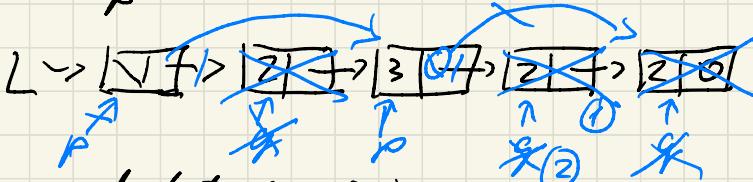
One-way lists with sentinel head (H1L)

$\langle 5, 2, 7, 3 \rangle$ is represented : $L \rightarrow [] \rightarrow [5] \rightarrow [2] \rightarrow [7] \rightarrow [3] \otimes$

(delete_all(L: E1*; x: N))

```

    P := L
    P -> next != &
      P -> next -> key = x
      delete_next(p) | P := P -> next
  
```



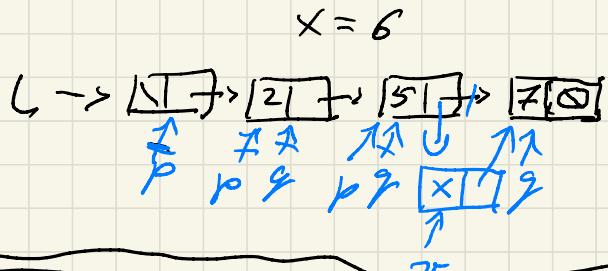
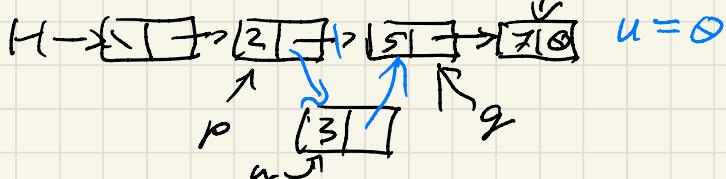
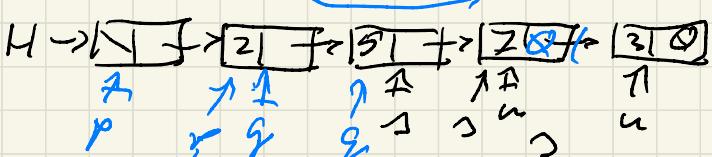
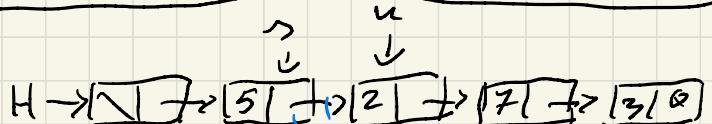
delete (L, 2)

Empty Lds f: L -> [] \otimes

Sorted_insert(L: E1x; x:N)

```

p := L; q := L->next
q != ⊖ & q->key < x
  p := q; q := q->next
n := new E1
n->key := x
n->next := q
p->next := n
    
```



Insertion Sort(H: E1x)

$s := H->next$

$s \neq \ominus$

$u := s->next$

$u \neq \ominus$

$s->key \leq u->key$

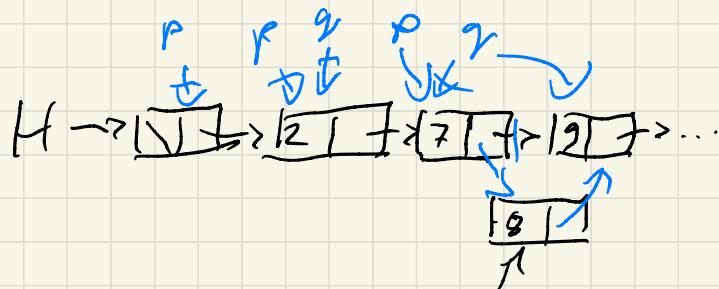
$s->next := u->next$

$u := sorted_insert(H, u)$

$u := s->next$

$$AT(n), MT(n) \in \Theta(n^2)$$

$$mT(n) \in \Theta(n)$$



$$\begin{aligned}
 MT_{IS}(n) &= 1 + \underbrace{n-1}_{n} + n-1 + \\
 &\quad + 0 + 1 + 2 + 3 + (n-2) \\
 &= \sum_{i=0}^{n-1} i = \frac{n(n+1)}{2} = \frac{1}{2}n^2 + \frac{1}{2}n \in \Theta(n^2)
 \end{aligned}$$

$$\begin{aligned}
 AT_{IS}(n) &= \overbrace{n + (n-1)}^{2n-1} + \frac{0 + 1 + 2 + \dots + (n-2)}{2} = n + (n-1) + \frac{(n-1)(n-2)}{2} = \\
 &= \frac{1}{2}n^2 + 1\frac{1}{2}n - \frac{1}{2} \in \Theta(n^2)
 \end{aligned}$$

(sorted_insert($H, u : E1*$))

```

 $p := H; q := H \rightarrow \text{next}$ 
 $q \rightarrow \text{key} \leq u \rightarrow \text{key}$ 
 $p := q; q := q \rightarrow \text{next}$ 
 $u \rightarrow \text{next} := q$ 
 $p \rightarrow \text{next} := u$ 
  
```

Queue

- $n: N$

- first, tail: $\mathbb{F} 1 \times$

+ Queue() {first := tail := new E1; n := 0}

+ add(x: N) {tail \rightarrow begin := x; tail :=
n++ tail \rightarrow next :=
new E1}

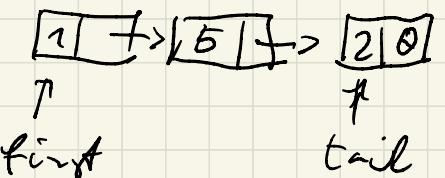
+ rem(): N

+ length(): N { return n }

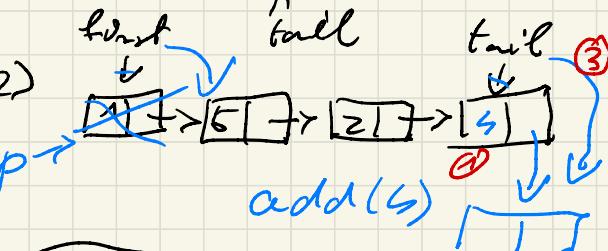
+ first(): N

Empty Queue: first = \emptyset

Queue: $\langle 1, 5, 2 \rangle$



Empty queue: first \rightarrow [] sentinel tail



Queue::rem(): N

first = tail

Queue-
under-
flow

x := first \rightarrow begin

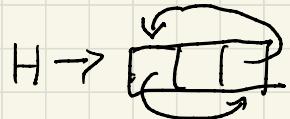
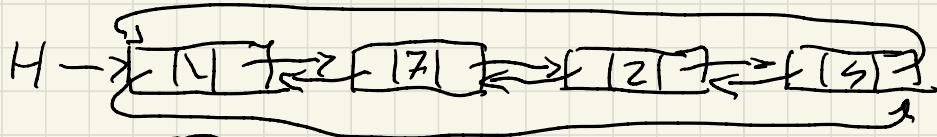
p := first ; n--

first := first \rightarrow next

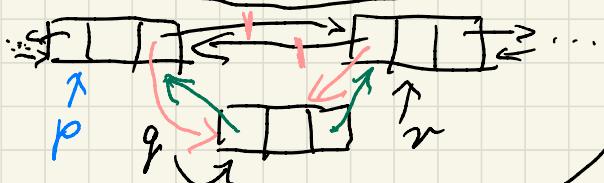
delete p ; return x

Cyclic two-way lists (C2Ls)

Default: the list has header



(precede(q , r : E2*)



$p := r \rightarrow p_{\text{prev}}$
 $\text{insert}(p, q, r)$

follow(p, q : E2*)

$r := p \rightarrow \text{next}$
 $\text{insert}(p, q, r)$

E2
+ key : \mathbb{Z}
+ next, prev : E2*
+ {E2()} {next := prev := this}

insert(p, q, r : E2*)

$q \rightarrow \text{next} := r; q \rightarrow \text{prev} := p$
 $p \rightarrow \text{next} := q; r \rightarrow \text{prev} := q$



unlink(q: E2*)

$p := q \rightarrow \text{prev}; r := q \rightarrow \text{next}$

$p \rightarrow \text{next} := r; r \rightarrow \text{prev} := p$

$q \rightarrow \text{next} := q \rightarrow \text{prev} := q$

$H \rightarrow \boxed{\quad} - \circled{7} - \boxed{\quad} \leftarrow H$

$H \rightarrow \boxed{\quad} - \circled{7} - \circled{2} - \boxed{\quad} \leftarrow H$

$H \rightarrow \boxed{\quad} - \circled{7} - \circled{2} - \circled{3} - \boxed{\quad} \leftarrow H$

More abstract view:

$H \rightarrow \boxed{\quad} - \circled{7} - \circled{2} - \circled{3} - \boxed{\quad} \leftarrow H$

$H \rightarrow \boxed{\quad} - \boxed{\quad} \leftarrow H$ File: $\langle 7, 2, 3 \rangle$

read_list(&H: E2*)

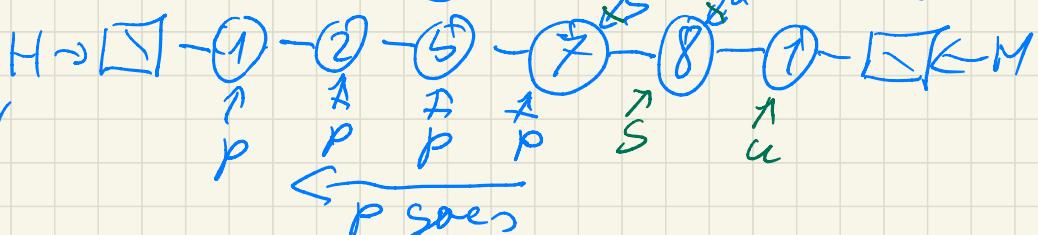
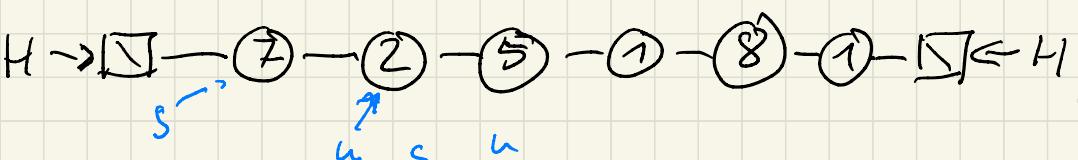
$H := \text{new } E2$

read(x)

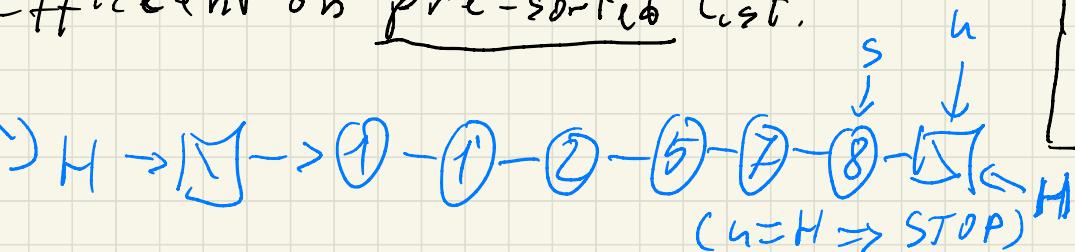
$q := \text{new } E2$

$q \rightarrow \text{help} := x$

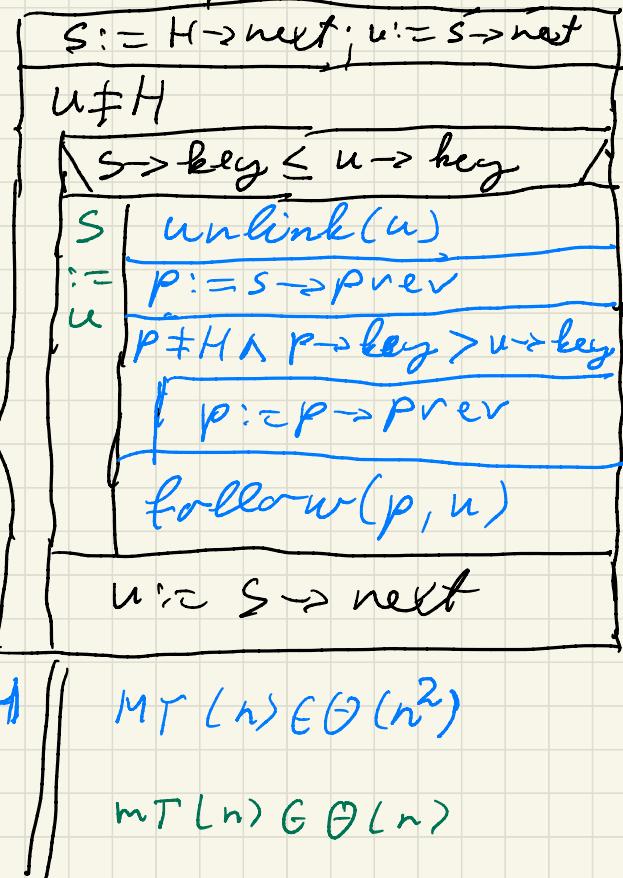
precede(q, H)



Efficient on pre-sorted list.



insertionSort($H: E[2..k]$)



destroy (H: E2*)

$p := H \rightarrow \text{next}$

$p \neq H$

unlink(p)

delete p

$p := H \rightarrow \text{next}$

delete H

$H := \emptyset$

print (H: E2*)

$p := H \rightarrow \text{next}$

$p \neq H$

write(p->key + " ")

$p := p \rightarrow \text{next}$