

2-10 Data Structures

Hengfeng Wei

hfwei@nju.edu.cn

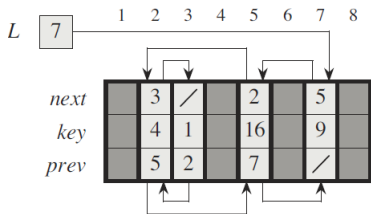
May 09, 2020



Compactation (Problem 10.3-4)

Keep all elements of a doubly linked list compact in storage, using the first n index locations in the multiple-array representation.

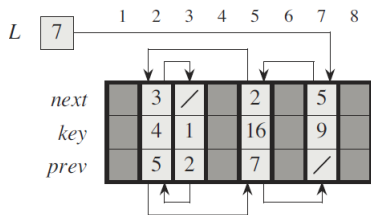
ALLOCATE-OBJECT() FREE-OBJECT(x)



Compactation (Problem 10.3-4)

Keep all elements of a doubly linked list compact in storage, using the first n index locations in the multiple-array representation.

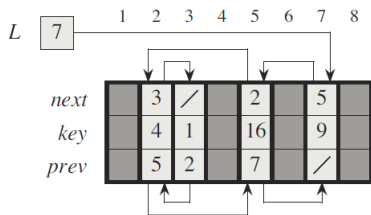
ALLOCATE-OBJECT() FREE-OBJECT(x)



Compactation (Problem 10.3-4)

Keep all elements of a doubly linked list compact in storage, using the first n index locations in the multiple-array representation.

ALLOCATE-OBJECT() FREE-OBJECT(x)



LIST-INSERT(L, x) LIST-DELETE(L, x)

ALLOCATE-OBJECT() LIST-INSERT(L, x)

ALLOCATE-OBJECT() LIST-INSERT(L, x)

Table: $L = 1$

Λ					
x					
Λ					

ALLOCATE-OBJECT() LIST-INSERT(L, x)

Table: $L = 1$

Λ					
x					
Λ					

Table: $L = 2$

Λ	1				
x	y				
1	Λ				

ALLOCATE-OBJECT() LIST-INSERT(L, x)

Table: $L = 1$

Λ					
x					
Λ					

Table: $L = 2$

Λ	1				
x	y				
1	Λ				

Table: $L = 4$

Λ	1	2	3		
x	y	s	t		
2	3	4	Λ		

ALLOCATE-OBJECT() LIST-INSERT(L, x)

Table: $L = 1$

Λ					
x					
Λ					

Table: $L = 2$

Λ	1				
x	y				
1	Λ				

Table: $L = 4$

Λ	1	2	3		
x	y	s	t		
2	3	4	Λ		

$x \leftarrow free$ $free \leftarrow free + 1$ Return x

LIST-DELETE(L, x) FREE-OBJECT(x)

LIST-DELETE(L, x) FREE-OBJECT(x)

LIST-DELETE($L, 2$)

Table: $L = 4$

Λ	1	2	3		
x	y	s	t		
2	3	4	Λ		

LIST-DELETE(L, x) FREE-OBJECT(x)

LIST-DELETE($L, 2$)

Table: $L = 4$

Λ	1	2	3		
x	y	s	t		
2	3	4	Λ		

Table: $L = 3$

Λ	1	2			
x	s	t			
2	3	Λ			

LIST-DELETE(L, x) FREE-OBJECT(x)

LIST-DELETE($L, 2$)

Table: $L = 4$

Λ	1	2	3		
x	y	s	t		
2	3	4	Λ		

Table: $L = 3$

Λ	1	2			
x	s	t			
2	3	Λ			

Moving the **elements** (not pointers) after x forward

$O(n)$

COMPACTIFY-LIST (Problem 10.3-5)

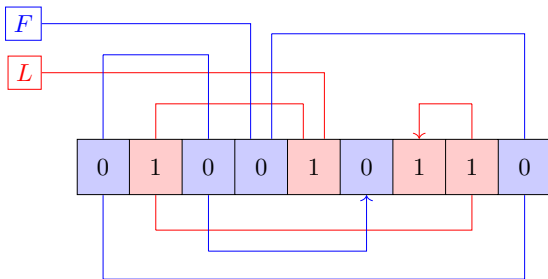
COMPACTIFY-LIST(L, F)

L : doubly linked list, $|L| = n$

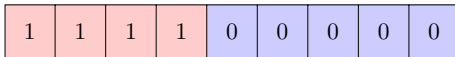
F : **doubly** linked free list, $|F| = m - n$

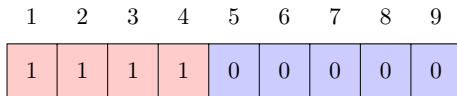
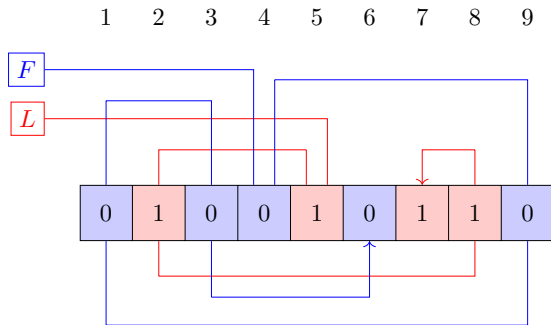
$\Theta(n)$

1 2 3 4 5 6 7 8 9

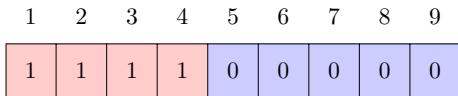
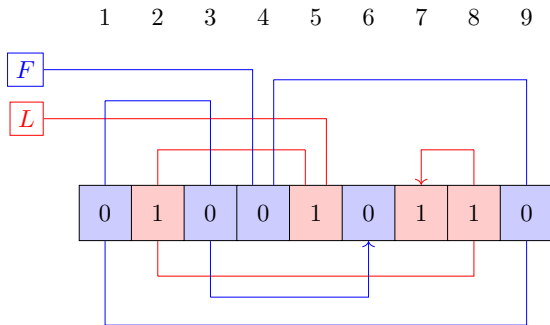


1 2 3 4 5 6 7 8 9





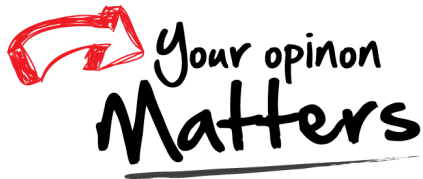
Swap (0, 1) pairs following (F, L)



Swap (0, 1) pairs following (F, L)

Swap only when $L > n \wedge F \leq n$

Thank
You!



Office 302

Mailbox: H016

hfwei@nju.edu.cn