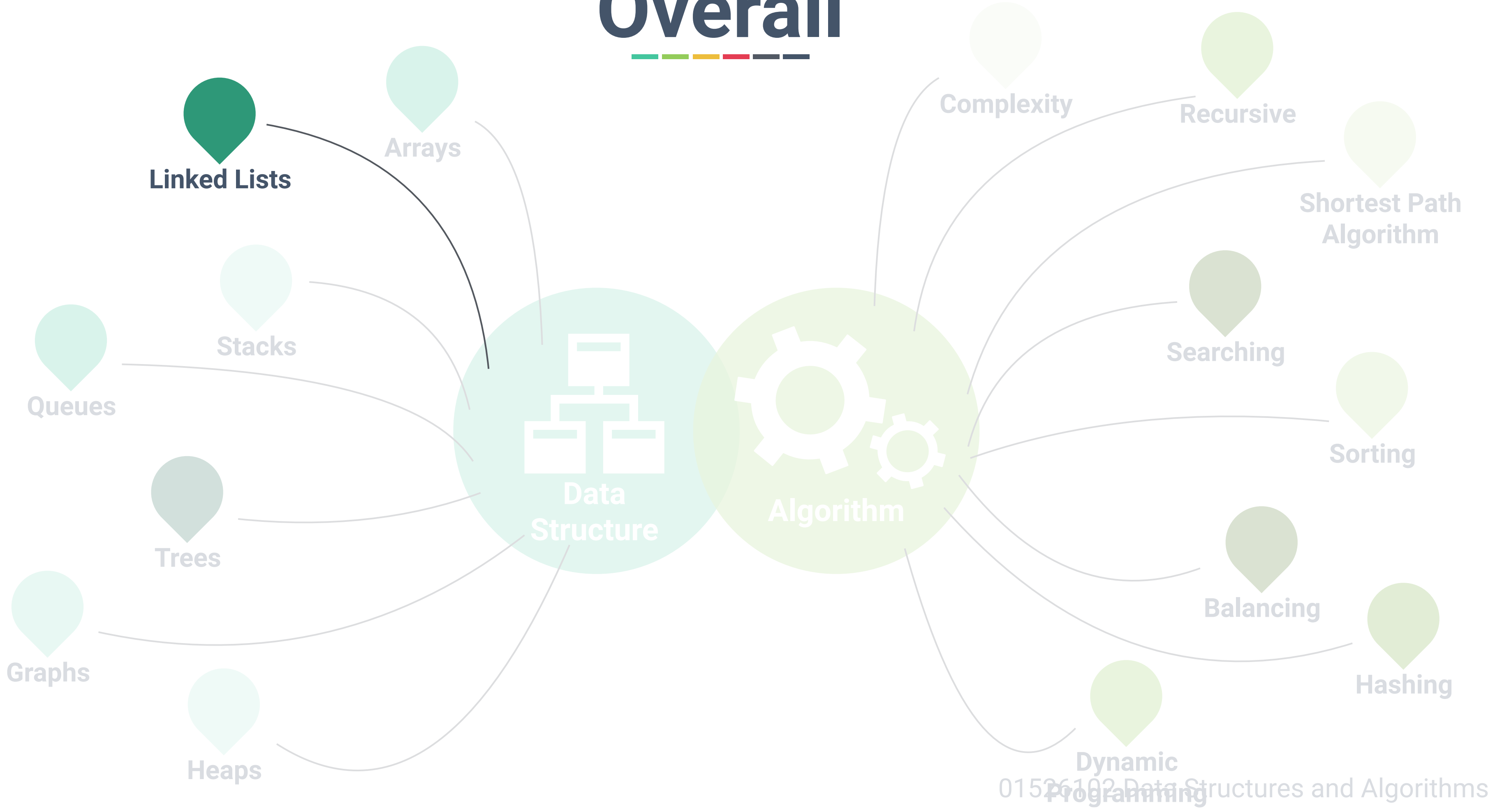
SCHOOL OF
INTERNATIONAL & INTERDISCIPLINARY
ENGINEERING PROGRAMS
King Mongkut's Institute of Technology Ladkrabang

Chapter 5: Linked Lists

Dr. Sirasit Lochanachit

Overall


Linked Lists
Arrays
Stacks
Queues
Trees
Graphs
Heaps
Data Structure
Algorithm
Complexity
Recursive
Shortest Path Algorithm
Searching
Sorting
Balancing
Hashing
Dynamic Programming

Today's Outline

1. What is a Linked List?
2. Singly Linked Lists
   - Traversing
   - Insert a node
   - Delete a node
   - Stack and Queue Implementation

Previously

Python's array-based list

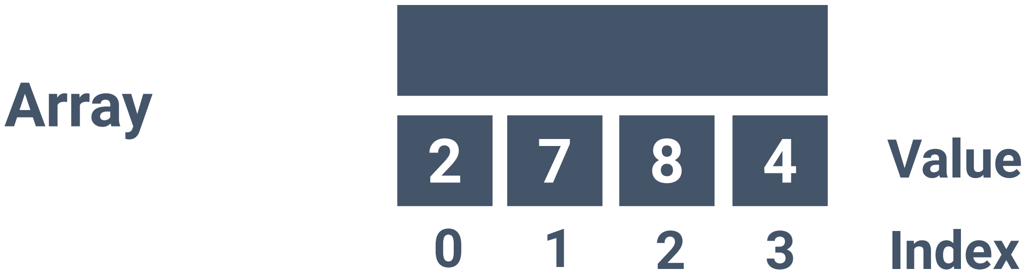
- Stack
- Queue

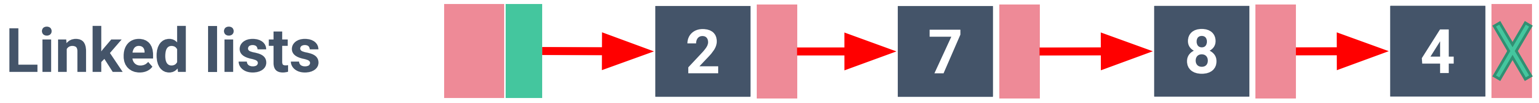
Disadvantages of array:

- Length of array has to be pre-allocated, empty space wasted.
- Adding or removing elements between values in the array is expensive - O(n)

Linked Lists

To avoid these limitations, an alternative to array is **linked list**.





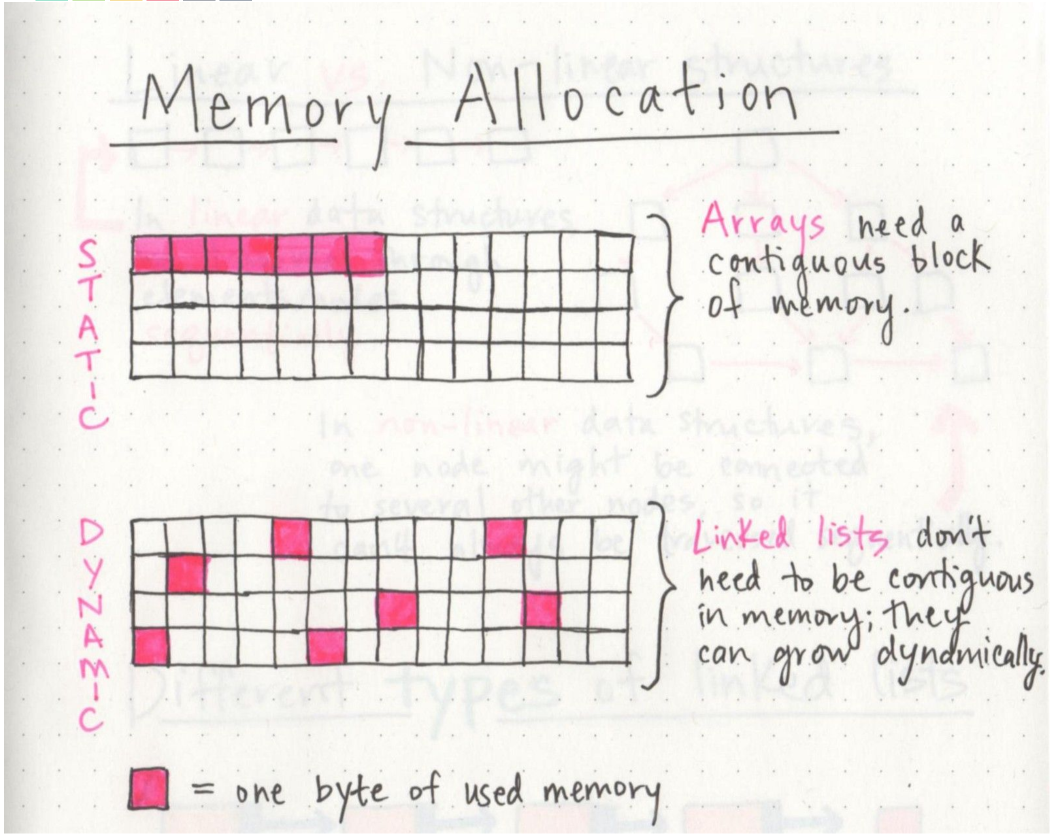
Linked Lists

Static

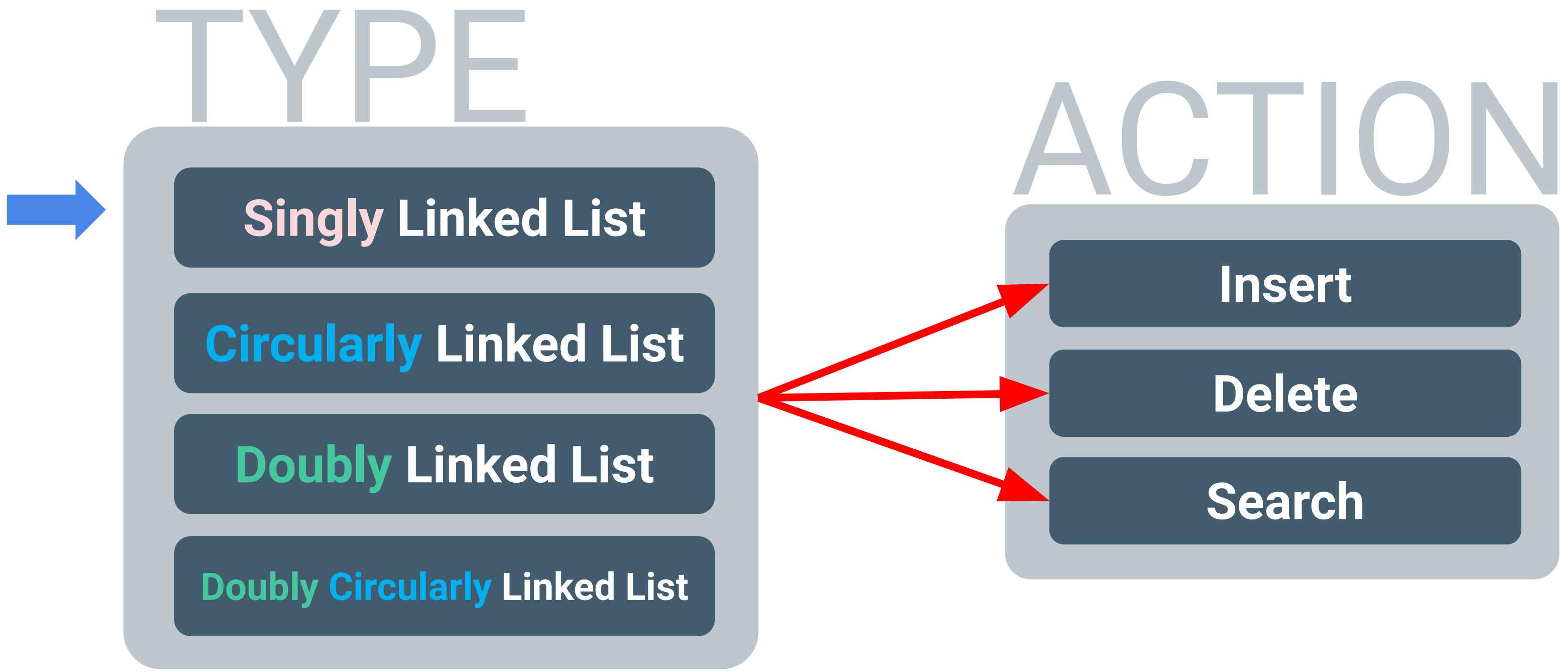
- Pre-allocated
- Fixed size
  - Unable to grow

Dynamic

- Allocated as needed
- Able to grow


Memory Allocation
STATIC
Arrays need a contiguous block of memory.
DYNAMIC
Linked lists don't need to be contiguous in memory; they can grow dyndmically.
= one byte of used memory

Linked Lists



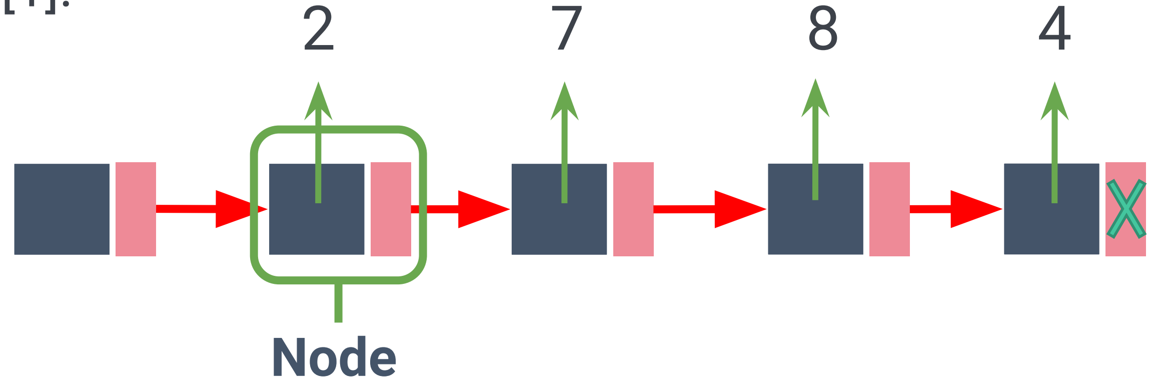
What is a Linked List?

A singly **linked list** is a collection of nodes that form a linear order of a sequence [1].

[1] Michael T. Goodrich et al., Data Structures and Algorithms in Python, 2013

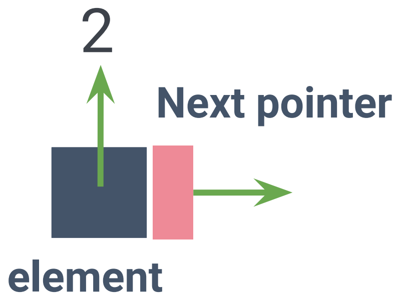
What is a Linked List?

A singly **linked list** is a collection of nodes that form a linear order of a sequence [1].

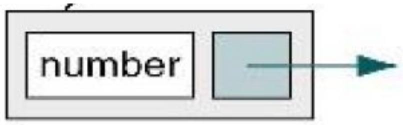


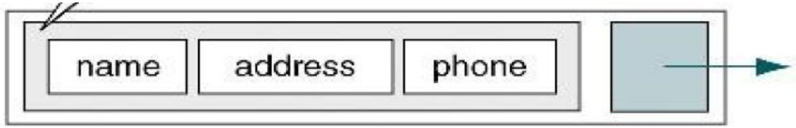
SCHOOL OF INTERNATIONAL & INTERDISCIPLINARY ENGINEERING PROGRAMS
King Mongkut's Institute of Technology Ladkrabang

Linked List Node



Linked List Node Structures

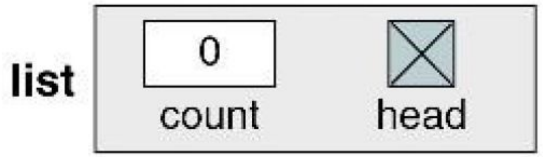




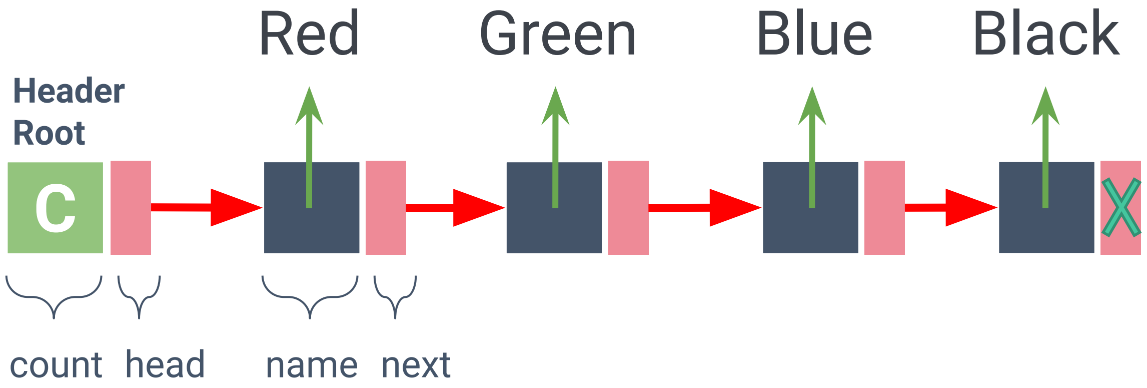
Create a Linked List

1. **Create a header/root node**

```
Algorithm createList (list)
    1. Allocate a list
    2. Set list head to null
    3. Set list count to 0
End createList
```



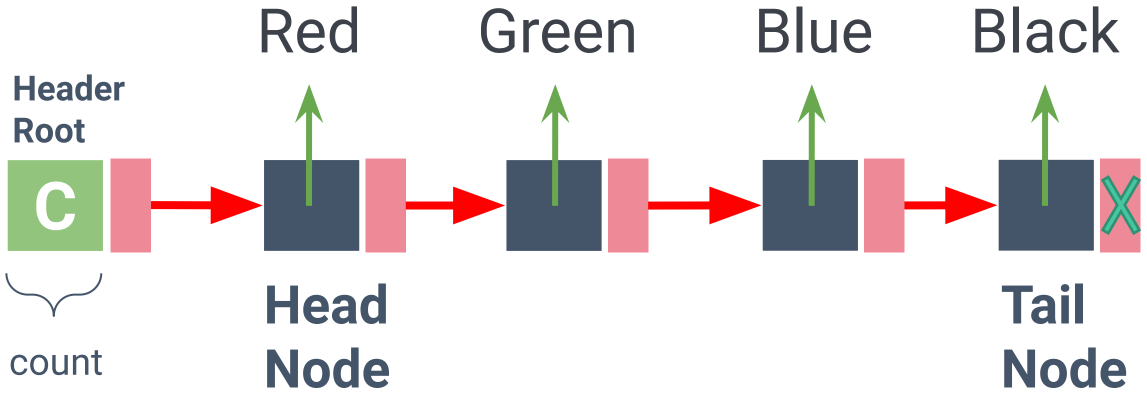
Singly Linked Lists



```
Color
    String                                              name
    Color next

End Color
```

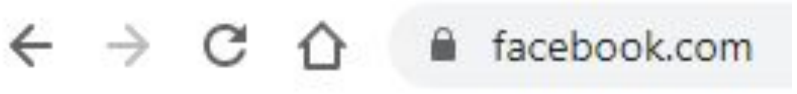
SCHOOL OF INTERNATIONAL & INTERDISCIPLINARY ENGINEERING PROGRAMS
King Mongkut's Institute of Technology Ladkrabang

Singly Linked Lists



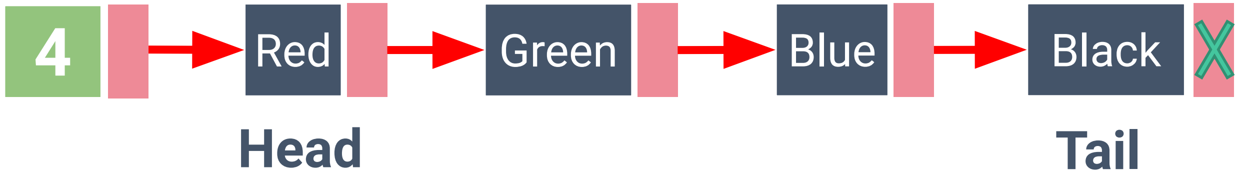
Linked Lists Examples

Real-life examples of Linked Lists:



Retrieved from https://live.staticflickr.com/5610/15429943089_edc7011843_o_d.jpg CC BY 2.0
https://live.staticflickr.com/23/26472155_8cc5066b66_o_d.jpg CC BY-SA 2.0

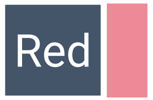
Singly Linked Lists



Create a Linked List

2. Create a data/element Node

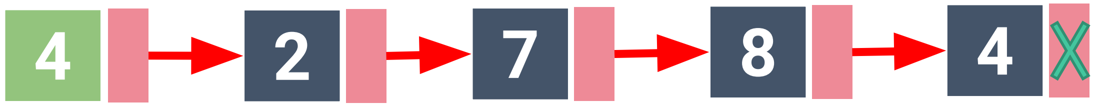
```
Algorithm createDataNode (d, p)

    colorNew = allocate(Color)

    name = d

    next = p

    return colorNew

End createDataNode
```



Traversing Singly Linked Lists

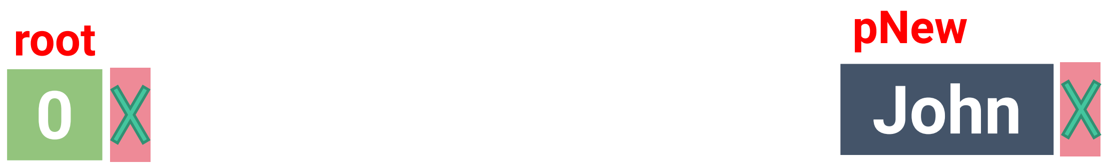
Address/ Byte#	Value
6000	4
6001	6002
6002	2
6003	6008
6004	8
6005	6012
6006	
6007	
6008	7
6009	6004
6010	
6011	
6012	4
6013	None

Suppose that it takes 1 byte to store an integer.



Insertion

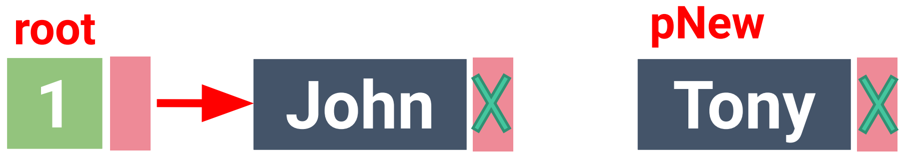
Add “John” Node into a list



Insertion

at the tail of the list

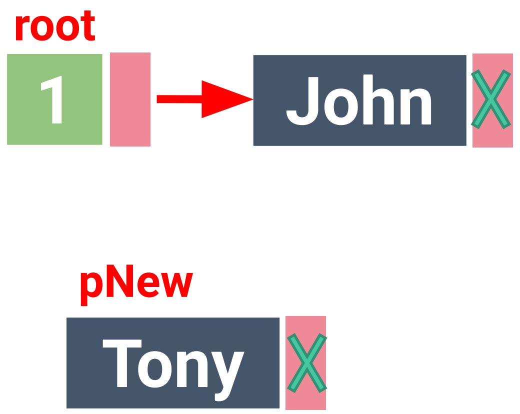
Add “Tony” Node into a list



Insertion

at the head of the list

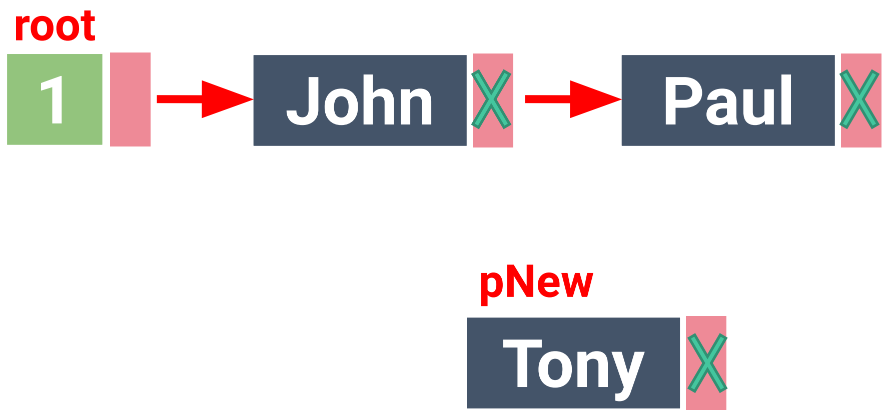
Add “Tony” Node at the front of a list



Insertion

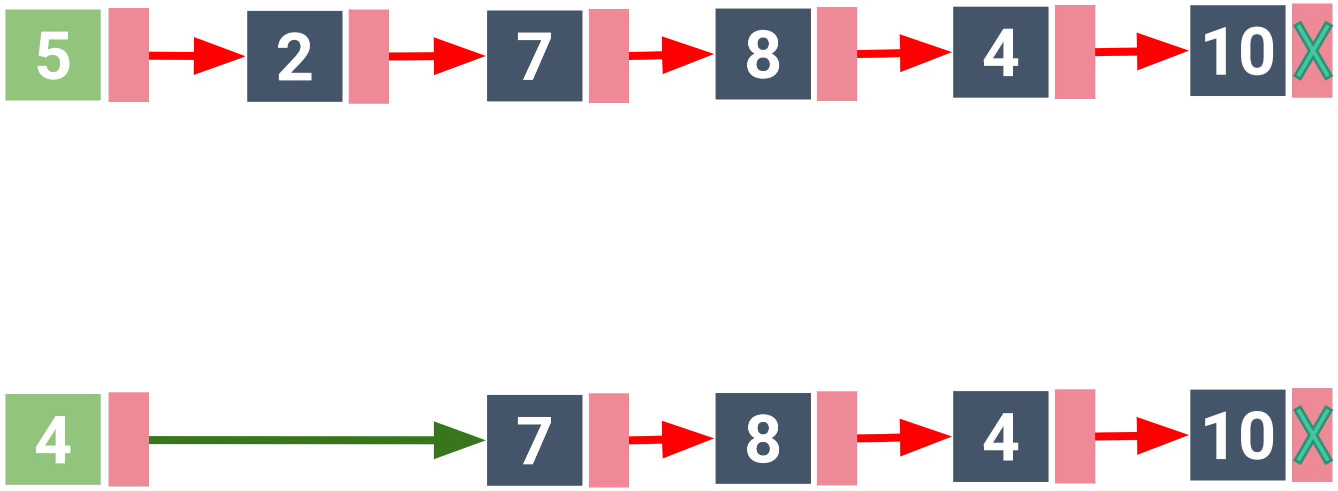
between nodes

Add “Tony” Node between John and Paul



Deletion

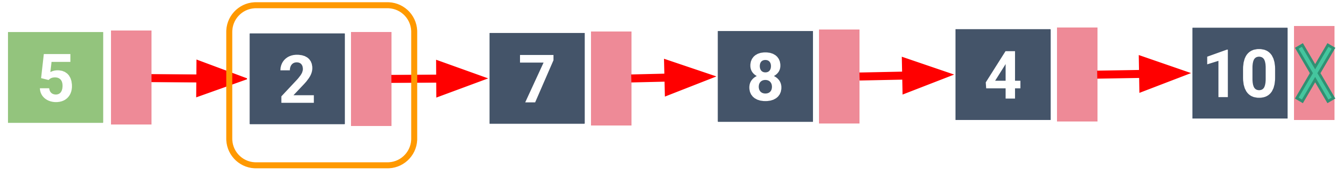
Delete the head node



SCHOOL OF INTERNATIONAL & INTERDISCIPLINARY ENGINEERING PROGRAMS
King Mongkut's Institute of Technology Ladkrabang

Deletion

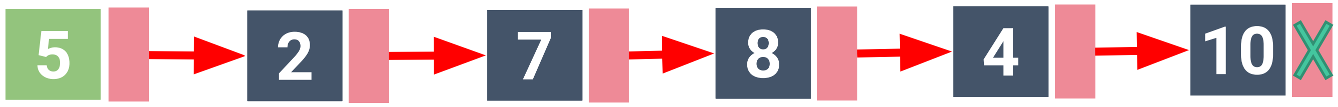
Delete the head node

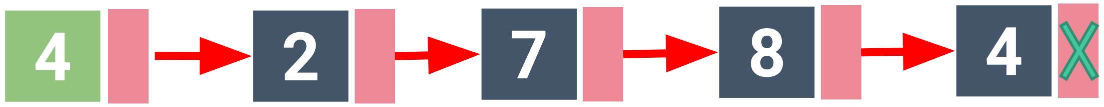


SCHOOL OF INTERNATIONAL & INTERDISCIPLINARY ENGINEERING PROGRAMS
King Mongkut's Institute of Technology Ladkrabang

Deletion

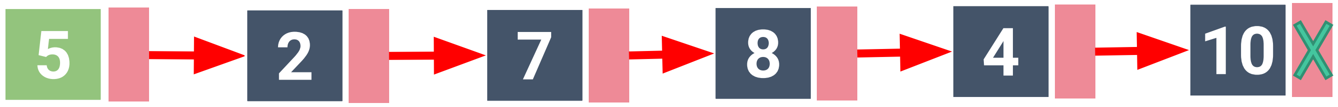
Delete the tail node





Deletion

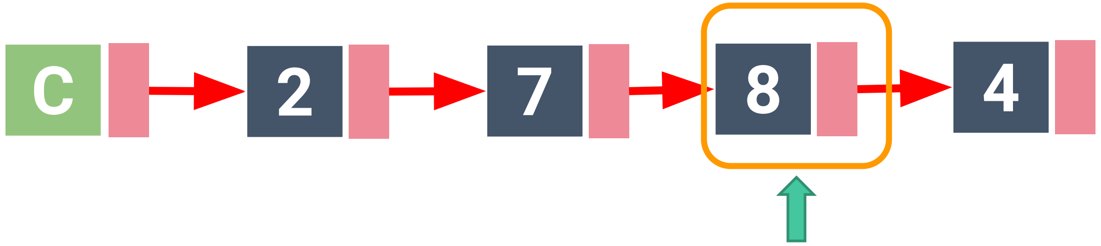
Delete the tail node



Deletion

Delete between nodes

Step 1



Step 2

Step 3

Singly Linked Lists: Stacks

How to Implement a Stack?

Array!!

and

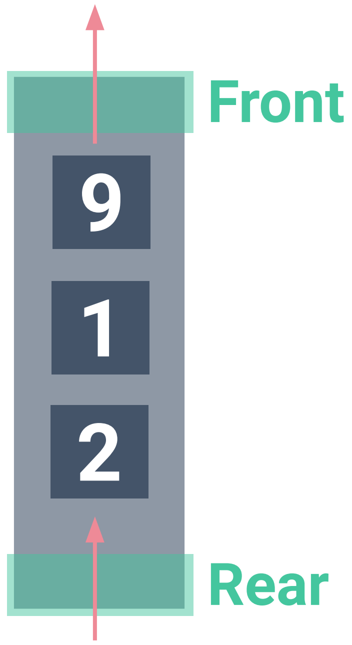
Linked Lists!!

- **Singly Linked Lists**

Asymptotic Performance

Operation	**Running Time - Array**	**Running Time - Singly Linked List**
S.push(element)	$O(1)$	
S.pop()	$O(1)$	
S.top()	$O(1)$	
S.is_empty()	$O(1)$	
len(S)	$O(1)$	

Singly Linked Lists: Queues



How to Implement a Queue?

Array!!

and

Linked Lists!!

- **Singly Linked Lists**

Asymptotic Performance

Operation	Running Time - Array	Running Time - Singly Linked List
Q.enqueue(e)	$O(1)$ or O(n)	
Q.dequeue()	$O(1)$ or O(n)	
Q.first()	$O(1)$	
Q.is_empty()	$O(1)$	
len(Q)	$O(1)$	