Marco Alvarez
Department of Computer Science and Statistics
University of Rhode Island

Spring 2019

Linked Lists

FATHERS OF THE DEEP LEARNING REVOLUTION RECEIVE ACM A.M. TURING AWARD

Bengio, Hinton, and LeCun Ushered in Major Breakthroughs in Artificial Intelligence

ACM named Yoshua Bengio, Geoffrey Hinton, and Yann LeCun recipients of the 2018 ACM A.M. Turing Award for conceptual and engineering breakthroughs that have made deep neural networks a critical component of computing. Bengio is Professor at the University of Montreal and Scientific Director at Mila, Quebec’s Artificial Intelligence Institute; Hinton is VP and Engineering Fellow of Google, Chief Scientific Adviser of The Vector Institute, and University Professor Emeritus at the University of Toronto; and LeCun is Professor at New York University and VP and Chief AI Scientist at Facebook.

The ACM A.M. Turing Award, often referred to as the “Nobel Prize of Computing,” carries a $1 million prize, with financial support provided by Google, Inc. It is named for Alan M. Turing, the British mathematician who articulated the mathematical foundation and limits of computing.

https://aidemos.microsoft.com/
Arrays

- Think about making insertions efficiently
  - what is the cost? (rear, front, middle)

- Think about making deletions efficiently
  - what is the cost?

Linked Lists

- Collections of sequential elements stored at **non-contiguous** locations

- Nodes are connected by **links**
  - every node keeps a pointer to the next node

- Can **grow** and **shrink** dynamically

Singly Linked List

<table>
<thead>
<tr>
<th>Head</th>
<th>Tail</th>
</tr>
</thead>
<tbody>
<tr>
<td><img src="data" alt="Node 1" /></td>
<td><img src="data" alt="Node 5" /></td>
</tr>
<tr>
<td><img src="data" alt="Node 7" /></td>
<td><img src="data" alt="Node 3" /></td>
</tr>
<tr>
<td><img src="data" alt="Node 2" /></td>
<td><img src="data" alt="Node 10" /></td>
</tr>
<tr>
<td><img src="data" alt="Node 4" /></td>
<td><img src="data" alt="Node 20" /></td>
</tr>
<tr>
<td>NULL pointer</td>
<td></td>
</tr>
</tbody>
</table>

Operations on Linked Lists

- Linked lists are just **collections** of sequential data
  - can **insert** 1 or more elements
    - front, end, by index, by value (sorted lists)
  - can **delete** 1 or more elements
    - front, end, by index, by value
  - can **search** for a specific element
  - can **get** an element at a given index
  - can **traverse** the list
    - visit all nodes and perform an operation (e.g. print or destroy)
  - …
Implementing a Singly Linked List

**Linked lists in C++**

- Need to review:
  - C++ Classes
  - Pointers
    - NULL pointers
  - Dynamic Memory Allocation
    - `new`
    - `delete`
  - Pointers and Classes
    - dot notation (`.`)
    - arrow notation (`->`)

```cpp
class Node {
  private:
    int data;
    Node *next;
    // add private data/methods
    // ...
  public:
    Node(int d);
    ~Node();
    // add public methods
    // ...
  friend class List;
};
```

```cpp
class List {
  private:
    Node *head;
    Node *tail;
    // add private data/methods
    // ...
  public:
    List();
    ~List();
    // add public methods
    // ...
};
```
Append (insert at end)

Prepend (insert at front)

Insert by index

Delete at front
Traversing a linked list

- Using a `for` loop
- Using recursion

Circular Singly Linked List

Head

1 -> 7 -> 3 -> 5

Tail