Linked Nodes w/ Loops

Hitesh Boinpally
Summer 2023
Agenda

• Linked Nodes review
• Traversing ListNode sequences
• Modifying ListNode sequences
• Reminders
Agenda

• Linked Nodes review
• Traversing ListNode sequences
• Modifying ListNode sequences
• Reminders
Linked Nodes Review

• New data structure to represent non-contiguous memory
• “Building blocks” for Linked Lists
  • “Legos”
• Today: arbitrary length sequences
Problems so far

• What statements turn this picture:

```
ListNode temp = list1.next;
list1.next = list1.next.next;
temp.next = list2;
list2 = temp;
```

(1 Possible) Answer:
Agenda

• Linked Nodes review
• Traversing ListNode sequences
• Modifying ListNode sequences
• Reminders
Suppose we have a sequence of nodes like the above
• With unknown number of elements

How would we print all the values out?
Printing a Sequence

• Pseudocode:
Printing a Sequence

• Pseudocode:
  Start at the front of the list
  While there are nodes to print:
    Print the current node’s data
    Go to the next node
Printing a Sequence

- **Pseudocode:**
  - Start at the front of the list
  - While there are nodes to print:
    - Print the current node’s data
    - Go to the next node

- How to go to the next node?
  - list = list.next;

```
data  next
10    
```

```
data  next
20    
```

...
Printing a Sequence

- **Pseudocode:**
  
  Start at the front of the list
  While there are nodes to print:
  - Print the current node’s data
  - Go to the next node

- How to go to the next node?
  - list = list.next;

```
list
```

<table>
<thead>
<tr>
<th>data</th>
<th>next</th>
</tr>
</thead>
<tbody>
<tr>
<td>10</td>
<td></td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th>data</th>
<th>next</th>
</tr>
</thead>
<tbody>
<tr>
<td>20</td>
<td></td>
</tr>
</tbody>
</table>

... 

<table>
<thead>
<tr>
<th>data</th>
<th>next</th>
</tr>
</thead>
<tbody>
<tr>
<td>800</td>
<td></td>
</tr>
</tbody>
</table>
Printing a Sequence

- **Pseudocode:**
  - Start at the front of the list
  - While there are nodes to print:
    - Print the *current* node’s *data*
    - Go to the *next* node

- How to go to the next node?
  - `list = list.next;`

```
<table>
<thead>
<tr>
<th>data</th>
<th>next</th>
</tr>
</thead>
<tbody>
<tr>
<td>10</td>
<td></td>
</tr>
<tr>
<td>20</td>
<td></td>
</tr>
<tr>
<td>...</td>
<td></td>
</tr>
<tr>
<td>800</td>
<td></td>
</tr>
</tbody>
</table>
```

```
Console:
10
20
```
Printing a Sequence

• **Pseudocode:**
  Start at the front of the list
  While there are nodes to print:
  Print the *current* node’s *data*
  Go to the *next* node

• How to go to the next node?
  • `list = list.next;`

```javascript
list
---
data  next
10  
---
data  next
data  next
20  
---
...
---
data  next
800  
```

Console:
10
20
...
800
Printing a Sequence

• **Pseudocode:**
  
  Start at the front of the list
  
  While there are nodes to print:
  
  Print the *current* node’s data
  
  Go to the *next* node

• How to go to the next node?
  
  • `list = list.next;`

<table>
<thead>
<tr>
<th>data</th>
<th>next</th>
</tr>
</thead>
<tbody>
<tr>
<td>10</td>
<td></td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th>data</th>
<th>next</th>
</tr>
</thead>
<tbody>
<tr>
<td>20</td>
<td></td>
</tr>
</tbody>
</table>

...
Printing a Sequence

- **Pseudocode:**
  
  Start at the front of the list
  While there are nodes to print:
  
  Print the *current* node’s *data*
  Go to the *next* node

- How to go to the next node?
  - `list = list.next;` *Destroys the list!*

<table>
<thead>
<tr>
<th>data</th>
<th>next</th>
</tr>
</thead>
<tbody>
<tr>
<td>10</td>
<td></td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th>data</th>
<th>next</th>
</tr>
</thead>
<tbody>
<tr>
<td>20</td>
<td></td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th>data</th>
<th>next</th>
</tr>
</thead>
<tbody>
<tr>
<td>800</td>
<td></td>
</tr>
</tbody>
</table>

Console:
- 10
- 20
- ...
- 800
Printing a Sequence

- **Pseudocode:**
  
  Start at the front of the list
  While there are nodes to print:
  Print the current node’s data
  Go to the next node

- **How to go to the next node?**
  - Create a new variable and change it
  - ListNode current = list
  - current = current.next

```
<p>| | |</p>
<table>
<thead>
<tr>
<th></th>
<th></th>
</tr>
</thead>
<tbody>
<tr>
<td>10</td>
<td></td>
</tr>
<tr>
<td></td>
<td>20</td>
</tr>
</tbody>
</table>
|   |...
|   | 800|
```
Printing a Sequence

• **Pseudocode:**
  Start at the front of the list
  While there are nodes to print:
    Print the current node’s data
    Go to the next node

• How to go to the next node?
  • Create a new variable and change it
  • ListNode current = list
  • current = current.next

Console:

| list
<table>
<thead>
<tr>
<th>data</th>
<th>next</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td>10</td>
</tr>
</tbody>
</table>

data

| current
<table>
<thead>
<tr>
<th>data</th>
</tr>
</thead>
<tbody>
<tr>
<td>10</td>
</tr>
</tbody>
</table>

next

<table>
<thead>
<tr>
<th>data</th>
<th>next</th>
</tr>
</thead>
<tbody>
<tr>
<td>20</td>
<td></td>
</tr>
</tbody>
</table>

...
Printing a Sequence

• **Pseudocode:**
  - Start at the front of the list
  - While there are nodes to print:
    - Print the current node’s data
    - Go to the next node

• **How to go to the next node?**
  - Create a new variable and change it
  - `ListNode current = list`
  - `current = current.next`

```
<table>
<thead>
<tr>
<th>data</th>
<th>next</th>
</tr>
</thead>
<tbody>
<tr>
<td>10</td>
<td></td>
</tr>
<tr>
<td></td>
<td>20</td>
</tr>
<tr>
<td>...</td>
<td></td>
</tr>
</tbody>
</table>
```

Console:
10
Printing a Sequence

- **Pseudocode:**
  - Start at the front of the list
  - While there are nodes to print:
    - Print the current node’s data
    - Go to the next node

- **How to go to the next node?**
  - Create a new variable and change it
  - ListNode current = list
  - current = current.next

```
<table>
<thead>
<tr>
<th>data</th>
<th>next</th>
</tr>
</thead>
<tbody>
<tr>
<td>10</td>
<td></td>
</tr>
<tr>
<td></td>
<td>20</td>
</tr>
</tbody>
</table>
```

**Console:**
```
10
20
```

```
<table>
<thead>
<tr>
<th>data</th>
<th>next</th>
</tr>
</thead>
<tbody>
<tr>
<td>10</td>
<td></td>
</tr>
<tr>
<td></td>
<td>20</td>
</tr>
<tr>
<td></td>
<td></td>
</tr>
<tr>
<td>800</td>
<td></td>
</tr>
</tbody>
</table>
```
Printing a Sequence

- **Pseudocode:**
  - Start at the front of the list
  - While there are nodes to print:
    - Print the current node’s data
    - Go to the next node

- **How to go to the next node?**
  - Create a new variable and change it
  - ListNode current = list
  - current = current.next

```
<table>
<thead>
<tr>
<th>Console:</th>
</tr>
</thead>
<tbody>
<tr>
<td>10</td>
</tr>
<tr>
<td>20</td>
</tr>
<tr>
<td>...</td>
</tr>
<tr>
<td>800</td>
</tr>
</tbody>
</table>
```

```
<table>
<thead>
<tr>
<th>list</th>
<th>data</th>
<th>next</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td>10</td>
<td></td>
</tr>
<tr>
<td></td>
<td>20</td>
<td></td>
</tr>
<tr>
<td></td>
<td>...</td>
<td></td>
</tr>
<tr>
<td></td>
<td>800</td>
<td></td>
</tr>
</tbody>
</table>
```

Diagram showing the sequence of nodes starting from list and moving to 10, 20, ..., 800.
Printing a Sequence

• **Pseudocode:**
  Start at the front of the list
  While there are nodes to print:
  Print the current node’s data
  Go to the next node

• How to go to the next node?
  • Create a new variable and change it
  • ListNode current = list
  • current = current.next

```plaintext
current
```

List:

```
data  next
10     
20     
...    
800    
```

Console:

```
10
20
...
800
```
Printing a Sequence

• **Pseudocode:**
  Start at the front of the list
  While there are nodes to print:
  Print the current node’s data
  Go to the next node

• **How to go to the next node?**
  - `list = list.next;`  **Destroys the list!**
  - `ListNode current = list`
  - `current = current.next`

  Current

  List is still intact!

```plaintext
list
  data  next
  10

  20

  ...

  800
```

**Console:**

10
20
...
800
Agenda

• Linked Nodes review
• Traversing ListNode sequences
• Modifying ListNode sequences
• Reminders
Adding to a Sequence

• Suppose we had the following sequence:

```
list1
1
```

What happens when we execute the below code?

```java
ListNode current = list1;
while (current != null) {
    current = current.next;
}

current = new ListNode(data: 4);
```
Agenda

• Linked Nodes review
• Traversing ListNode sequences
• Modifying ListNode sequences
• Reminders
Reminders

• First resubmission form released tomorrow
  • Due next Friday
  • Can only resubmit assignments we have given feedback on!

• No section on Tuesday
  • We will still post materials
  • Optional like always