



CSC110: Tutorial 7

Mr Tony Chung
a.chung@lancaster.ac.uk
<http://www.tonychung.net/>

+ Today's Objectives

- Linked Lists (20 mins)
- ASCII and Basic Ciphers (20 mins)
- Quiz (20 mins)
- Questions / General Help
- **If you need help with a specific topic, shout now!**



Linked Lists

+ Quick Recap: How Stuff is Stored

- Some programming languages store data in structures.
 - A slice of memory is allocated and all the data is stored together.
- Other languages store data in different memory locations.
 - Then the data is linked together by using pointers (addresses).
- This has already been seen in Java.
 - Primitives are stored directly.
 - Objects are stored in memory and then referenced.
 - Hence why `==` wont always work.
- Linked lists use the principle of references.
 - Other data structures exist that do the same.

+ Quick Recap: Arrays

- Recall that arrays have a finite length.
 - If the array is too small, it has to be expanded.
 - This can sometimes mean allocating a whole new block.
 - Then all the data has to be moved across.
 - Sometimes arrays are too large – so there is waste.
 - These are inefficiencies.
- There are benefits, however:
 - Arrays are easy to use.
 - Elements are in fixed blocks and so get stored together.
 - This usually results in faster speeds.

+ Linked Lists

- Rather than using arrays, elements can be used to store data and a pointer to the next element.
- The list is accessed by using a pointer, which points to the first element.
- The end of the list is when `next == null`;

```
class LinkedListElement{  
    Object o;  
    LinkedListElement next;  
}
```

+ Linked Lists: Print all the Elements

```
for( LinkedListElement e = l; e != null; e = e.next ){  
    System.out.println( e.o );  
}
```

... or ...

```
public class LinkedListElement{  
    ...  
    public void printList(){  
        System.out.println( o );  
        if( next != null ){    next.printList();  }  
    }  
    ...  
}
```



Exercise 1: Obtain Length

- Obtain the length of a linked list in two ways:
 - 1. Using a for loop.
 - 2. Using recursion.

+ Answer 1: Obtain Length

```
LinkedListElement l = .....  
int size = 0;  
for( LinkedListElement e = l; e != null; e = e.next ){  
    size++;  
}
```

or...

+ Answer 1b: Obtain Length

```
LinkedListElement l = ....
```

```
l.getSize();
```

```
public class LinkedListElement{
```

```
    ...
```

```
    public int getSize(){
```

```
        if( next == null ){
```

```
            return 1;
```

```
        }
```

```
        else{
```

```
            return 1 + next.getSize();
```

```
        }
```

```
    }
```

```
    ...
```

+ Exercise 2: Summation

- Imagine the Objects are all Integers.
- Obtain the sum of all Integers in the linked list in two ways:
 - 1. Using a for loop.
 - 2. Using recursion.

+ Answer 2: Summation

```
LinkedListElement l = .....  
int sum = 0;  
for( LinkedListElement e = l; e != null; e = e.next ){  
    if( e.o instanceof Integer ){  
        sum += (Integer)(e.o);  
    }  
}
```

or...

+ Answer 2b: Summation

```
LinkedListElement l = ....
```

```
l.getSum();
```

```
public class LinkedListElement{
```

```
    ...
```

```
    public int getSum(){
```

```
        if( next == null ){ // Warning: Check if Integer
```

```
            return (Integer)0;
```

```
        }
```

```
        else{
```

```
            return (Integer)0 + next.getSum();
```

```
        }
```

```
    }
```

```
    ...
```



ASCII and Basic Ciphers

+ ASCII / Character Encodings

- All characters are fundamentally stored as numbers.
- In ASCII, these are often bytes (8-bit).
 - Other encodings exist, such as unicode and big5.
- The ASCII table is a common way to lookup the character code of a letter, number or symbol.
 - Special characters exist for stuff like newline, ESC and bell!
 - Look it up on Google.
 - Remember that lower-case and upper-case have different codes!
- In Java...
 - You can convert a string to bytes using the `.getBytes()` method.
 - You can convert a byte array to String using a constructor.

+ Basic Ciphers

- I am not going to try to teach you cryptography.
- I will teach you some simple ciphers involving character codes.
- The assignment talks about several of these:
 - Rotation
 - Shifting
 - Keys
- We will do rotation. The others depend on you.
 - Feel free to ask questions at the end.

+ Exercise 3: ROT-13

- ROT-13 was common on usenet to protect spoilers!

```
String s = "Hello";  
byte ba[] = _____;  
  
for( int i = 0; i < _____; i++ ){  
    _____  
    ba[i] = _____; // Hint %  
    _____  
    ba[i] = _____; // Hint %  
}  
  
String s2 = _____;
```

+ Answer 3: ROT-13: Almost

```
String s = "Hello";
byte ba[] = s.getBytes();

for( int i = 0; i < s.length(); i++ ){
    if( ba[i] >= 'a' && ba[i] <= 'z' )
        ba[i] = ba[i] + 13;
    else if( ba[i] >= 'A' && ba[i] <= 'Z' )
        ba[i] = ba[i] + 13;
}

String s2 = new String( ba );
```

+ Answer 3: ROT-13 Complete

```
String s = "Hello";
byte ba[] = s.getBytes();

for( int i = 0; i < s.length(); i++ ){
    if( ba[i] >= 'a' && ba[i] <= 'z' )
        ba[i] = 'a' + ( ( ( ba[i] - 'a' ) + 13 ) % 26 );
    else if( ba[i] >= 'A' && ba[i] <= 'Z' )
        ba[i] = 'A' + ( ( ( ba[i] - 'A' ) + 13 ) % 26 );
}

String s2 = new String( ba );
// See BlueJ – this is quite compacted...
```

+ Quiz

20

- 6 Questions (if we have time)

+ Q1.

- What problem could occur if a circular link occurs in a linked list?
 - Memory will be wasted.
 - There is a risk of a system freeze.
 - None of the above.

+ Q1.

- What problem could occur if a circular link occurs in a linked list?
 - Memory will be wasted.
 - **There is a risk of a system freeze. (The method will not end.)**
 - None of the above.
- **NB: Most modern systems will not freeze, they will just slow down.**
- **Memory will eventually be used up, as the stack grows out of control.**

+ Q2.

- A LinkedList is beneficial because it allows for variable length lists.
 - What disadvantage does it have when compared to an array?

+ Q2.

- A LinkedList is beneficial because it allows for variable length lists.
 - What disadvantage does it have when compared to an array?
 - It uses more memory. Each element must store the memory address of the next element. Memory addresses are usually 4 – 8 bytes long.

+ Q3.

- There are two ways to access a LinkedList we discussed.
 - (a) A loop.
 - (b) Recursion.

- What are the advantages and disadvantages of each?

+ Q3.

- There are two ways to access a LinkedList we discussed.
 - (a) A loop.
 - (b) Recursion.
- What are the advantages and disadvantages of each?

	Advantage	Disadvantage
Loop	Simple. Fast.	Must have raw array access.
Recursion	Can be provided as a public method.	High overhead. Risk of crash.

+

Q4.

27

- LinkedLists are one example of a pointer-based data structure. Which of the following is not an example?
 - Binary tree.
 - Structure.
 - Map.

+ Q4.

- LinkedLists are one example of a pointer-based data structure. Which of the following is not an example?
 - Binary tree.
 - **Structure.**
 - Map.

A structure is a group of data items, use of pointers is optional.

+ Q5.

- Which of the following characters are rotatable in ROT-13?
 - a
 - F
 - ?
 - \$
 - 8

+ Q5.

30

■ Which of the following characters are rotatable in ROT-13?

- a
- F
- ?
- \$
- 8

+ Q6.

- A key is...?
 - Part of the algorithm?
 - Always suitable to be shared?
 - Intended to be replaced so that the algorithm can be left alone.

+ Q6.

- A key is...?
 - Part of the algorithm?
 - Always suitable to be shared?
 - **Intended to be replaced so that the algorithm can be left alone.**

+ Questions

- Please email topic suggestions.