COMP 202 Final Review

In the Final Exam

1. Binary numbers (BInary to Decimal, Decimal To Binary)
2. Primitive Types: `int`, `boolean`, `String`, `double`
3. Type casting
4. Operations: `+`, `-`, `*`, `/`, `++`, `--`, `+=`, `-=`, `%`
5. Boolean Logic: AND, OR, NOT (&&, `||`, `!`)
6. Comparisons: `>`, `>=`, `<`, `<=`, `==`, `!=`
7. Conditional Statements: `if`, `else if`, `else`
8. Loops: `for` and `while`
9. Arrays
10. Methods, Passing variables to methods: Primitive types, and reference types (Arrays, Objects)
11. Sorting (Bubble Sort, Insertion Sort, and other sorting algorithms)

In the Final Exam

12. Data structures
    - Node class in LinkedList and doubly LinkedList
    - what is a hashing function, what is a collision
    - ArrayList and Hashtable, and the reference types (Integer, String, Double, Character)
13. Classes and Objects
    - constructors, object methods and variables
    - static variables and methods
    - public and private variables
14. Exceptions
    - try, catch, throws
    - ArrayIndexOutOfBounds
    - NullPointerException

NOT int the Final Exam

1. Complexity
2. Recursion
3. Merge sort
4. Advanced topics (Except for tthe part where we created a Robot object)

How would you know you are prepared for the Final exam?

If you feel comfortable answering all of the questions in all of the Assignments, you are prepared for the final

Some exercises

Mystery method 1

```java
public static void mysteryMethod1( int[] input, int i, int j){
    int tmp = input [i];
    input[i] = input[j];
    input[j] = tmp;
}

```

Mystery method 2

```java
public static void mysteryMethod2( int[] input)
{
    boolean testCondition = true;

    while (testCondition)
    {
        testCondition = false;
        for( int k = 0; k < input.length; k++)
        {
            if(input[k] > input[k+1])
            {
                mysteryMethod1(input, k, k+1);
                testCondition = true;
            }
        }
        System.out.println(Arrays.toString(input));
    }
}

```

Mystery method 3

```java
public static void mysteryMethod3( int[] input)
{
    for( int k = 0; k < input.length; k++)
    {
        int magicNumber = input[k];
        int magicNumberIndex = k;

        for( int m = k; m < input.length; m++)
        {
            if(input[m] < magicNumber)
            {
                magicNumber = input[m];
                magicNumberIndex = m;
            }
        }
        mysteryMethod1(input,m,k);
        System.out.println(Arrays.toString(input));
    }
}

```

Mystery method 4

```
public myMysteryBox
{
    public static void main(String args[])
    {
        int x = 10;
        int y = 20;
        y = swap(x,y);
        System.out.println(x+" , "+y);
    }

    public static int mysteryMethod4(int y, int x)
    {
        int tmp;
        tmp = x;
        x = y;
        y = tmp;
        return x;
    }
}

```

Mystery methods revealed

1. `mysteryMethod1`: the good old `swap` method
2. `mysteryMethod2`: the good old `bubbleSort` algorithm
3. `mysteryMethod3`: the previously unknown `selectionSort` algorithm
4. `MyMysteryBox`, `mysteryMethod4`: Question 15 from the midterm

For the final, you just need to be able to keep track of the *state* of variable when executing a piece of code *line by line*

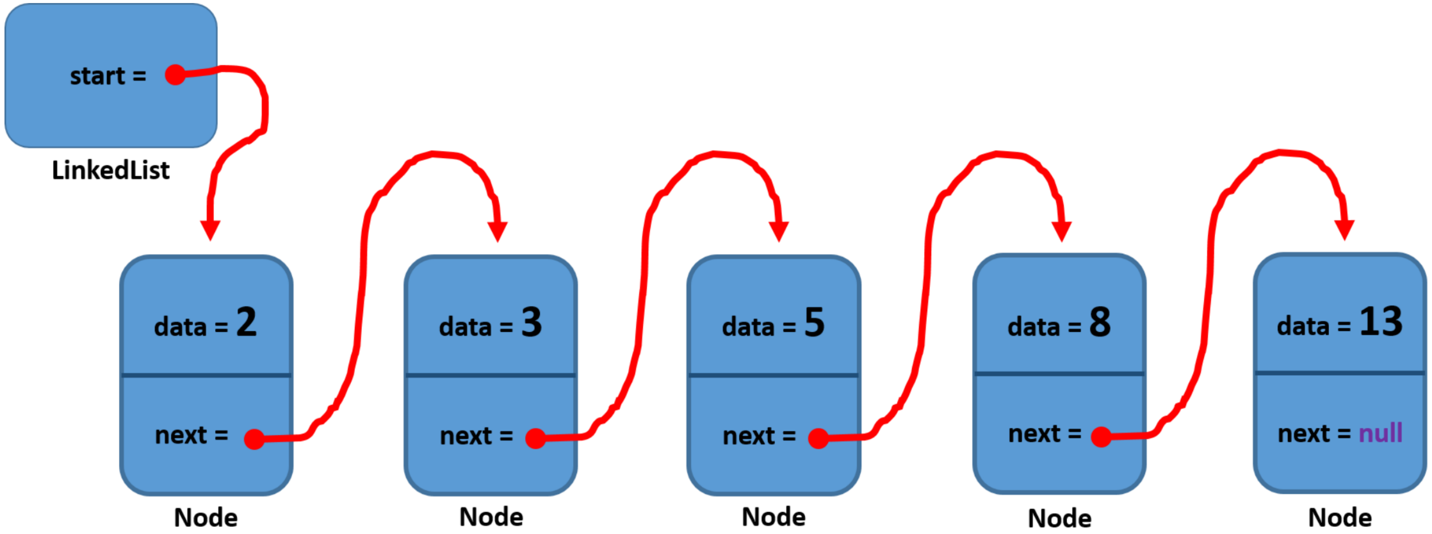
Practice Exercise on Objects - Implementing a `MiniString` class

Create a replacement for the Java `String` class, call it `MiniString`

Follow the specification form this file: http://cim.mcgill.ca/~gamboa/cs202/Material/MiniString.pdf

Linked Lists

This is a visual representation of a linked list using `int` values



1. How would you write a method that finds the *maximum* number in a LinkedList?

2. How would you write a method that finds the *average* of the numbers in a LinkedList?

Exceptions

What type of eception would these pieces of code throw?

```java
ArrayList<Integer> myList = new ArrayList<Integer>();

myList.get(27); // This will make Java scream
                // at you with red lines
```

```java
City[] cities = new City[35];

City someCity = cities[10]; // This is less than 34,
                            // so it is ok

someCity.setNeighbours(); // This will make Java scream
                          // at you with red lines
```

```java
String notANumber =  "This is not a number.";

// This will make Java scream at you with red lines
int aNumber = Integer.parseInt(notAnumber);
```

Tips for the final

- Study the programming and comprehension questions from all assignments
- Study the explore method from Assignment 4
- Try to write the java code for other sorting algorithms: ShellSort, CocktailSort, GnomeSort, Combsort
- Try to follow by hand the execution of any sorting algorithms
- Practice binary conversions
- Do the MiniString exercise

Resources

- The final exam from last year: http://cim.mcgill.ca/~gamboa/cs202/Material/notes/Final-1.pdf