

CMSC 10200 Practice Midterm Exam

revision 0, 4/25/2006

University of Chicago, Spring 2006

Look Closely

The class `Range` is intended to represent consecutive ranges of integers. The object `new Range(2,7)`, for example, represents the set of integers $\{2, 3, 4, 5, 6, 7\}$. Sadly, the following class definition is riddled with errors. Find them.

```
public class Range {
    private int lo, hi;

    public Range(int lo, hi) {
        if (lo > hi) throw new IllegalArgumentException("none in that range");
        this.lo = lo;
        this.hi = hi;
    }

    public int size() {
        s = hi - lo + 1;
        return s;
    }

    public int sum() {
        int total = 0;
        for (int i=0; i<=hi; i++)
            total = total + hi;
        return total;
    }

    public boolean contains(int n) {
        return (lo <= n <= hi);
    }

    public String toString() {
        String s = "{";
        for (int i=lo; i>lo; i++)
            s = s + i + " ";
        return (s + "}");
    }
}
```

Read Some Functions

The following class is a collection of utility methods, similar in shape to the `Math` class. State in plain language what each of its three methods does, and suggest a better name for each one.

```
public class Utilities {

    public static boolean f1(int a, int b) {
        return ((a<0) && (b>0)) || ((a>0) && (b<0));
    }

    public static int f2(int[] ms, int[] ns) {
        int mL = ms.length;
        int nL = ns.length;
        if (mL > nL) {return mL;}
        else      {return nL;}
    }

    public static boolean f3(int lo, int hi, int[] ns) {
        int L = ns.length;
        int c = 0;
        for (int n : ns)
            if ((lo <= n) && (n <= hi))
                c = c + 1;
        return (c == L);
    }
}
```

Write Some Functions

Write the class `MoreUtilities` containing three static methods as follows:

1. a function that is given two `doubles` and returns the one closer to 1
2. a function that is given an array of `Objects` and an `Object` and returns a `boolean` indicating whether or not the latter is in the former
3. a function to test whether or not a `String` contains¹ the character '@'

¹The class `String` includes the method `charAt(int)`, where `"Java".charAt(0)` is 'J' and `"Java".charAt(1)` is 'a'.

A Generic Class

Consider the generic class `Numbered<T>`. Its purpose is to pair an `int` with an element of type `T`.

You should be able to use `Numbered<T>` objects as follows:

```
Numbered<Pres> Washington = new Numbered<Pres>(1, new Pres( ... ));
Numbered<Pres> Adams = new Numbered<Pres>(2, new Pres( ... ));

Numbered<Course> CS102 = new Numbered<Course>(102, new Course(...));
```

Design and define the class `Numbered<T>`.

Object-Oriented Design

You reach into your pocket and pull out three scraps of paper. The scraps say

1. Greg Maddux (773) 555-3746, maddux@hotmail.com
2. Dave Schwimmer (773) 555-4444 home, (773) 555-1122 cell, www.schwimmer.com
3. Oprah's fax (773) 555-9283

Being an object-oriented programmer, you think *A-ha! An opportunity to build a Java class hierarchy!* which fills you with joy.

Implement a set of classes, including one that implements this interface:

```
public interface AddressBook {
    public boolean isEmpty();
    public int numItems();
    // you may assume unique full names in your address book
    public String email(String firstname, String lastname);
    // but you may not assume unique first names
    public Set<String> distinctFirstNames();
}
```

This problem is loosely specified by design; much is left up to you.

more space for previous item

JSP

Fill in the following sketch of a simple JSP web application. The application will convert kilometers to miles. Assume in your calculations that one kilometer is 0.62 miles.

```
<!-- this is the file input.jsp -->
```

```
<html><head> ... </head><body>  
<h1>Kilometer to Mile Converter: Input</h1>
```

```
</body></html>
```

```
<!-- this is output.jsp -->
```

```
<html><head> ... </head><body>  
<h1>Kilometer to Mile Converter: Output</h1>
```

```
</body></html>
```