Iterators & Two Towers

Lab 4

Wednesday, February 16, 2011

CSC 062: Spring, 2011

The programming exercise for today's laboratory is about the Two Towers problem. To make the problem tractable, we are taking a "bottom up" approach. It is not the way you would normally approach a problem, but it makes sense here because we have already thought about the situation. You will first write an integer subset class and then use it to solve the problem.

Before coming to lab, please read carefully section 8.7 of the "Java Structures" text or you won't be able to make much progress on this lab. As before, feel free to work in a pair, but with someone you have not worked with before.

IntegerSubsetIterator

For the first step, you are to write a class

```
public class IntegerSubsetIterator
             implements Iterator<ArrayList<Integer>>
```

whose constructor takes a single integer `size` and creates an object that iterates over all the subsets of $\{0, 1, \ldots, \texttt{size-1}\}$. There will be $2^{\texttt{size}}$ subsets.

The iterator will return "subsets" as ArrayLists whose elements are the elements of a subset. For example, running the code

```
IntegerSubsetIterator iter = new IntegerSubsetIterator(3);
while (iter.hasNext())
    System.out.println(iter.next());
```

will produce the output:

```
[]
[0]
[1]
[0, 1]
[2]
[0, 2]
[1, 2]
[0, 1, 2]
```

Be sure that you understand what the iterator is doing before writing any code. When you are ready, open a new Eclipse project for this lab, and use one of the strategies from the preliminary reading for this lab to implement the class.

The `Iterator` interface requires only three methods, `hasNext`, `next`, and `remove`. The last of these is not used here and can be the "do nothing" method.

Two Towers

(insert favorite Lord of the Rings reference here :-)

Now that you have created the subset iterator, you are ready to use it to solve the Two Towers problem. As Bailey describes in his book, one approach is to compute the heights of all the blocks

$$h = \sum_{i=1}^{15} \sqrt{i},$$

and then to look through all subsets of $\{1, 2, \ldots, 15\}$ to find the one whose height comes closest to, without exceeding, $h/2$.

In the class `Two Towers`, we provide you with a program that will use your iterator to solve the Two Towers program. Test your iterator thoroughly before using it in this program, as you will have a hard time determining whether your iterator is correct by looking at the answers printed by your program.

Once you have a working program, experiment by increasing the number of blocks. Try values like 24 and 28. Go up slowly until you have a value that causes a measurable delay. What happens to the time if you increase the value by 1? What if you increase it by 2. Based on these timings, what do you think the big-O complexity of the problem is?

When you are done, send me an e-mail (kim@cs.pomona.edu) with the output of the program for number of blocks equal to 24 and 28. Include a discussion of what you think the big-O complexity is of this problem and why. As e-mail return addresses are often obscure, please include your name, and, if you worked in a pair, the name of your partner,