

CSC 148H1 S Midterm

Winter 2004

Last Name: _____ # 1: _____ / 5
First Name: _____ # 2: _____ / 5
Student Number: _____ # 3: _____ / 5
Lecture Section: _____
Lab day, time, room: _____ TOTAL: _____ / 15

Duration — 50 minutes Aids allowed: none

*Do **not** turn this page until you have received the signal to start.*
(Please fill out the identification section above,
and read the instructions below.) *Good Luck!*

This midterm consists of 3 questions on 5 pages (including this one). *When you receive the signal to start, please make sure that your copy is complete.*
Comments are not required except where indicated, although they may help us mark your answers. They may also get you part marks if you can't figure out how to write the code.
Write your student number at the bottom of pages 2-5 of this test.
If you use any space for rough work, please indicate clearly what you want marked.
Please remove everything from your desk except for a writing implement.

Question 1. [5 MARKS]

The Josephus Problem involves a group of n people standing in a circle. Every k th person is executed until only one remains.

Here is an example. Consider $n = 5$ and $k = 2$, where persons P1, P2, P3, P4 and P5 are standing in a circle. Starting the count at P1, P2 is the second person, and so P2 is the first to be executed. P4 is next, then P1, and then P5. P3 is the survivor. (If you have trouble understanding this then you should draw the circle of people and start putting X's through the people who are executed; continue to eliminate every second survivor until only P3 is left.)

In this question you will complete a method that performs this operation on a circular doubly-linked list (a circular list is one where the last Node's `next` value refers to the first Node and the first Node's `previous` value refers to the last Node). Fill in the blanks in method `josephus` on the next page.

Hint: draw a picture of a circular linked list and trace your code to make sure that it is correct.

Here is class Node:

```
public class Node {
    public Object data;
    public Node next;
    public Node previous;
}
```

```

public class JosephusProblem {

    /** Remove every kth node from the circular list pointed to by first
        and return the last remaining Node. */
    public static Node josephus(Node first, int k) {
        Node temp = first; // refer to Node #1.

        // As long as there is more than one Node
        while (                ) {

            // Move temp to the next Node to be eliminated
            for (                ) {

                } // end for

            // Now unlink the Node that temp refers to and fix temp

        } // end while

        return temp;
    }
}

```

Question 2. [5 MARKS]

Complete class `MultipliesIterator` on the next page. A `MultipliesIterator` returns `Integers` that are multiples of a particular number in a particular range. For example, `new MultipliesIterator(3, 7)` would create an iterator that returns `Integer` objects containing 0, 3 and 6.

Student #: _____

Page 3 of 5

CONT'D...

```
import java.util.*;
public class MultiplesIterator implements Iterator {

    /** Return an iterator of Integers that are multiples of i in
        the range 0 .. n, not including n. */
    public MultiplesIterator(int i, int n) {

    }

    public Object next() {

    }

    public boolean hasNext() {

    }

    public void remove() { throw new UnsupportedOperationException(); }
}
```

