Read and fill in this page now.
Do NOT turn the page until you are told to do so.

Your name: ____________________________
Your login name: ____________________________
Your lab section day and time: ____________________________
Your lab t.a.: ____________________________
Name of the person sitting to your left: ____________________________
Name of the person sitting to your right: ____________________________

Problem 0  ____  Total:  ____  /30
Problem 1  ____  ____  Problem 3  ____
Problem 2  ____  ____  Problem 4  ____  ____

This is an open-book test. You have approximately two hours to complete it; time estimates indicate a pace sufficient to finish in one hour. You may consult any books, notes, or other paper-based inanimate objects available to you. To avoid confusion, read the problems carefully. If you find it hard to understand a problem, ask us to explain it. If you have a question during the test, please come to the front or the side of the room to ask it.

Restrict yourself to Scheme constructs covered in chapters 3 through 6 of Simply Scheme and the “Difference Between Dates” case study, part 1.

This exam comprises 15% of the points on which your final grade will be based. Partial credit may be given for wrong answers. Your exam should contain five problems (numbered 0 through 4) on seven pages, along with the code from the “Difference Between Dates” program.

Please write your answers in the spaces provided in the test; in particular, we will not grade anything on the back of an exam page unless we are clearly told on the front of the page to look there. We believe we have provided more than enough space for your answers, so please don’t try to fill it all up.

A few students are taking this exam late. Please refrain from discussing the exam with them and from positing any exam-related comments on discussions or newsgroups until Wednesday.

Relax—this exam is not worth having heart failure about.
Problem 0 (2 points, 1 minute)
Put your login name on each page. Also make sure you have provided the information requested on the first page.

Problem 1 (6 points, 10 minutes)

Part a
Clearly add parentheses and quotes in the line below so that when you evaluate the result, you get astabc.

        butfirst word last abc

Part b
Fill in the blanks below to show the intermediate steps of the evaluation and the final value of the expression.

        (sentence 'ab '(cd ef) (butfirst '(gh)) (butfirst 'x) (butfirst 'yz))

Steps:

The result of evaluating (butfirst '(gh)) is _______________________________ .

The result of evaluating (butfirst 'x) is _______________________________ .

The result of evaluating (butfirst 'yz) is _______________________________ .

The result of evaluating the entire expression is

________________________________________________________________________ .

Problem 2 (4 points, 10 minutes)
Give a good comment for the following procedure.

        (define (mystery x)
             (first (butfirst (last (butlast x)))) )

Your comment should describe all x for which mystery returns a value without crashing. It should also describe (in English, not Scheme) what mystery returns for a given x when it doesn’t crash.
Problem 3 (10 points, 20 minutes)
Write a procedure named 2003-date that, given an integer between 1 and 365—a day of the year in a non-leap year—returns the corresponding date in 2003 in the same format as the argument to day-span. (2003-date will be an inverse for the day-of-year procedure in Appendix B of the “Difference Between Dates” case study.) Examples:

<table>
<thead>
<tr>
<th>expression</th>
<th>value</th>
</tr>
</thead>
<tbody>
<tr>
<td>(2003-date 1)</td>
<td>(january 1)</td>
</tr>
<tr>
<td>(2003-date 32)</td>
<td>(february 1)</td>
</tr>
<tr>
<td>(2003-date 365)</td>
<td>(december 31)</td>
</tr>
</tbody>
</table>

You should assume that a procedure named 2003-month-number has already been defined that, given a day of the year in a non-leap year, returns the corresponding month number (1 for January, 2 for February, and so on). Examples:

<table>
<thead>
<tr>
<th>expression</th>
<th>value</th>
</tr>
</thead>
<tbody>
<tr>
<td>(2003-month-number 1)</td>
<td>1</td>
</tr>
<tr>
<td>(2003-month-number 32)</td>
<td>2</td>
</tr>
<tr>
<td>(2003-month-number 365)</td>
<td>12</td>
</tr>
</tbody>
</table>

Your solution may include helper procedures. You will lose points for using cond or if in any part of your solution. You should call procedures in the existing “Difference Between Dates” code (either appendix; both are listed at the end of this exam) wherever appropriate. To minimize writing, you may abbreviate january to jan, february to feb, and so on.
Problem 4 (8 points, 19 minutes)

Write code for both parts of this problem assuming that you know nothing about how a date is represented, only that month-name and date-in-month when given a date as argument will return the date’s month name and integer date-in-month correctly.

Part a
Write a procedure is-special-day? that, given a legal date, returns true when the date is either March 11, April 2, or April 27, and returns #f otherwise. For this part, you may not use and or or; you may use if and cond. Make no more comparisons—uses of equal?, =, or member?—than necessary.

Part b
Rewrite is-special-day? so that it avoids the use of if and cond. You may use and and or. As in part a, you should make no more comparisons—uses of equal?, =, or member?—than necessary.