CSC236 – Problem Set 8

There are two components of this problem set: a preliminary question designed to check your understanding of the basic topics covered this week, and a set of more challenging questions designed to make you think critically about the material and apply it in new contexts. Get in the habit of starting work early – the less time you give yourself, the most stressed you’ll find yourself each week!

Caution: you must submit two separate files in two separate locations on MarkUs, one for the Preliminary and one for the Challenge.

To avoid suspicions of plagiarism, clearly state any resources (people, print, electronic) outside of your group, the course notes, and the course staff, you consulted at the beginning of your assignment submission.

Preliminary: due March 13, 2014

This question is an opportunity for you to check your understanding of the topics and practice writing formal solutions. This is a valuable learning opportunity – if you see that you’re at a loss, get help quickly!

Your goal should not be to get the right answer, but to convince the marker that you know what you’re doing. This question is marked on the following 3-point scale:

<table>
<thead>
<tr>
<th>Score</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>3</td>
<td>You’ve mastered this topic</td>
</tr>
<tr>
<td>2</td>
<td>You’re almost there, but missing something</td>
</tr>
<tr>
<td>1</td>
<td>You don’t really know what you’re doing</td>
</tr>
<tr>
<td>0</td>
<td>You didn’t submit/had absolutely no clue</td>
</tr>
</tbody>
</table>

This question must be completed INDIVIDUALLY. Prove that the following code terminates. There is an “obvious” loop measure to use; however, remember that the loop measure must always be a natural number, i.e., \( \geq 0 \). You will need to prove a suitable loop invariant about \( m \) and \( x \) in order to conclude that \( m \) is a loop measure.

```python
def mystery(n):
    ''' Pre: n is a natural number '''
    m = n * n
    x = n
    while m != 0:
        m -= 2 * x - 1
        x -= 1
    return m
```
Challenges: due March 15, 2014

Answer each question completely, always justifying your claims and reasoning. Your solution will not just be graded on correctness, but on its clarity as well. Technically correct answers that are hard to understand will not receive full marks. Mark values for each question are contained in the [square brackets].

You may work in groups of up to THREE to complete these questions.

1. Consider the following modification of a termination example from lecture.

```python
1 def term_ex2(x,y):
2     ''' Pre: x and y are natural numbers. '''
3     a = x
4     b = y
5     while a >= 0 || b >= 0:
6         if a > 0:
7             a -= 1
8         else:
9             b -= 1
10     return x * y
```

(a) [2] Demonstrate via example that term_ex2 doesn’t always terminate; be sure to justify your answer.

(b) [2] In the Course Notes (pg. 55), we use the loop measure \( m = a + b \) to prove termination. Why doesn’t this loop measure work for term_ex2?

2. When we talk about “how sorted” a list is, one thing we can measure is the number of misordered pairs; i.e., the size of the set \( S(A) = \{(i,j) | i < j \land A[i] > A[j]\} \). For example, if \( A = [3,1,2] \), then \( S(A) = \{(0,1),(0,2)\} \), so \( |S(A)| = 2 \). It is useful to note that in a sorted list, \( |S(A)| = 0 \), and in the list \([n,n-1,\ldots,2,1]\), \( S(A) = \frac{n(n-1)}{2} \).

Now consider the following code.

```python
1 def isort2(A):
2     '''
3     Pre: A is a list (of numbers)
4     Post: A is sorted (in non-decreasing order)
5     '''
6     i = 1
7     while i < len(A):
8         if A[i] < A[i-1]:
9             swap A[i], A[i-1]
10         if i > 1:
11             i -= 1
12         else:
13             i += 1
```

(a) [6] Prove that this code terminates. Hint: use \( |S(A)| \) somehow in your loop measure. You may assume the following loop invariant (no need to prove it): \( 0 \leq i \leq \text{len}(A) \).

(b) [5] Prove that this code is correct, by finding a suitable loop invariant. Again, you may assume that \( 0 \leq i \leq \text{len}(A) \) is a loop invariant. (Hint: remember that the goal of isort2 is to sort the list. This should be reflected in your loop invariant.)

You may get full marks for this question, even if you don’t complete part (a).