CSC236 – Problem Set 7

There are two components of this problem set. The preliminary question is not marked or submitted: it is there as a suggested exercise that you should do early to make sure that you're on track. The problem set itself is what you will submit for marks.

Get in the habit of starting work early – the less time you give yourself, the more stressed you'll find yourself each week!

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

The PDF file you submit must be typed, scanned handwritten submissions will not be marked.

Preliminary: Not Marked

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!

Consider the following function.

```python
def mystery(a, b):
  ''' Pre: a >= 0 and b > 0'''
  x = a
  y = 0
  while x >= b:
    x -= b
    y += 1
  return x
```

Prove the following loop invariant for the while-loop: $x \geq 0$ and $a = x + yb$. Recall that to prove an invariant requires two things: that the invariant holds when the loop is reached, and that one arbitrary loop iteration maintains the invariant.

Problem Set: due November 11, 2016 22:00, required filename: ps7sol.pdf

Answer each question completely, always justifying your claims and reasoning. Your solution will be graded not only on correctness, but also on clarity.

Answers that are technically correct 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. [**8**] Consider the following function which returns the **least common multiple** (LCM) of two natural numbers a and b, i.e, the smallest positive integer $m > 0$ such that $m \bmod a = 0$ and $m \bmod b = 0$.

```python
def get_lcm(a, b):
  '''
  Pre: a, b are natural numbers, a >= 1, b >= 1
  Post: return LCM(a, b)
  '''
  x = a
  y = b
  while x != y:
    if x < y:
      x = x + a
    else:
      y = y + b
  return x
```

(a) Find the proper loop-invariant for the while-loop so that it is sufficient for proving the correctness of `get_lcm`. **Hint:** The invariant should probably include (but not limited to) the following conjuncts: $x \leq \mathrm{LCM}(a, b)$ and $x > 0$. Also, your invariant should be as simple as possible given that it is sufficient for proving the correctness of the function.

(b) Prove the invariant that your find in (a), using induction.

(c) Use the invariant to prove that the postcondition of `get_lcm` is satisfied if the loop terminates.

(d) Prove that the loop terminates. Remember that this involves finding a **variant** and proving two things: that the loop body is guaranteed to decrease the variant, and that the guard and invariant together imply that the variant is a natural number ≥ 0.

2. **[4]** Show that the following function terminates by finding and proving a suitable variant, i.e., you need to prove that the loop body is guaranteed to decrease the variant, and that the variant must be ≥ 0 while in the loop.

```python
def mystery(a, b):
  '''Pre: a >= 0, b >= 0'''
  while a >= 0 and b >= 0:
    if a < b:
      a, b = b, a
    else:
      a = a - 1
  return a
```

Hint: It is unlikely that $a + b$ would work (why?), but consider some other kind of combination of a and b.