Programming Languages [Fall 2017]
Test I

NAME: ____

Instructions:

1) This test is 7 pages in length.

2) You have 75 minutes to complete and turn in this test.

3) Short-answer questions include a guideline for how much to write. Respond in complete English sentences.

4) This test is closed books, notes, laptops, phones, smartwatches, friends, neighbors, etc.

5) Use the backs of pages in this test packet for scratch work. If you write more than a final answer in the area next to a question, circle your final answer.

6) Write and sign the following: "I pledge my Honor that I have not cheated, and will not cheat, on this test."

____

____

Signed: ____

1. [5 points]
Why might it be preferable to define functions in Curried form, versus taking a tuple of arguments? [1-2 sentences]

2. [15 points]
a) Does SML use dynamic or static scoping?

b) Define an SML program P and unequal values v and v', such that P evaluates to v using dynamic scoping and to v' using static scoping.

3. [40 points]

a) Implement, subject to the constraints of Assignment I, a function mkPairs that takes a list L and returns a duplicate-free list of all pairs that can be built from elements of L. E.g., mkPairs [1,1,2,3,2] returns a list containing the elements (1,1), (1,2), (1,3), (2,1), (2,2), (2,3), (3,1), (3,2), and (3,3). The order of the elements in the returned list doesn't matter. The expected type of mkPairs is $\alpha_{=}$ list $\rightarrow$ ($\alpha_{=}\times\alpha_{=}$) list, where $\alpha_{=}$ is what SML/NJ calls ''a. It should be clear that the proper running time for mkPairs is $O(n^2)$; to help with attaining this running time, your response may not use the @ operator.

b) Re-implement mkPairs, this time subject to the constraints of Assignment II (including no use of the keywords let/local, and all recursion must be in maps/folds).

4. [15 points]
For the remainder of this test, N always refers to a natural number as defined in class.

Assignment II used a different definition (*axiomatization*) of addition than the definition shown in class. Let's call the in-class rules the $+_1$ rules, and the assignment rules the $+_2$ rules. Here they are:

$$\boxed{N_1 +_1 N_2 = N_3} \qquad \frac{}{Z +_1 N = N}\ \text{ZP} \qquad \frac{N_1' +_1 N_2 = N_3'}{S(N_1') +_1 N_2 = S(N_3')}\ \text{SP}$$

$$\boxed{N_1 +_2 N_2 = N_3} \qquad \frac{}{N +_2 Z = N}\ \text{PZ} \qquad \frac{N_1 +_2 N_2' = N_3'}{N_1 +_2 S(N_2') = S(N_3')}\ \text{PS}$$

a) Intuitively, we believe that these two definitions of addition are equivalent, i.e., produce the same sums. Formalize this intuition as a theorem statement.

b) How would you prove the theorem described in Part (a)? Don't show a complete proof; just state the proof technique(s) and show which cases would need to be proved. Don't worry about whether extra lemmas would be needed; you don't need to show them.

5. [25 points]
Let's return to the axiomatization of addition that only uses the ZP and SP rules.

$\boxed{N_1+N_2=N_3}$

$$\frac{}{Z+N=N}\ ZP \qquad \frac{N_1'+N_2=N_3'}{S(N_1')+N_2=S(N_3')}\ SP$$

a) Pretend that you've proved the following lemma, which says that sums are unique.
$\forall$ $N_1..N_4$: If $N_1+N_2=N_3$ and $N_1+N_2=N_4$, then $N_3=N_4$.

Now prove the following theorem, which says that addition is associative.
$\forall$ $N_1..N_7$: If $N_1+N_2=N_4$, $N_4+N_3=N_5$, $N_2+N_3=N_6$, and $N_1+N_6=N_7$, then $N_5=N_7$.

[Undergraduates stop here. The remaining problem is for graduate students.]

b) [10 points]
Pretend-time is over.☺ Prove the lemma stated in Part (a).