Instructions:

1) This test is 7 pages in length.

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

3) Short-answer and essay questions include a guideline for how much to write. Respond in complete English sentences and avoid using bulleted and itemized lists.

4) For full credit on ML-response questions, implementations must avoid the @ operator and run in linear (O(n)) time.

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

6) 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.

7) 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]
   What is a programming language? [1-2 sentences]

2. [13 points] [Essay, 5-8 sentences]
   Compare and contrast left associativity with right associativity. Provide examples of each.
3. [15 points]
Consider the function \texttt{zip(L1,L2)}. The SML documentation says that \texttt{zip} will “combine the two lists \texttt{L1} and \texttt{L2} into a list of pairs, with the first element of each list comprising the first element of the result, the second elements comprising the second element of the result, and so on. If the lists are of unequal lengths, \texttt{zip} ignores the excess elements from the tail of the longer one”.

(a) What is the type of \texttt{zip}?

(b) Implement \texttt{zip} subject to the constraints of Assignment I.

(c) Implement \texttt{zip} subject to the constraints of Assignment II (including with currying).

4. [12 points]
Consider the function \texttt{porder L}. My documentation says that \texttt{porder} will “reorder \texttt{L}, placing it in ‘pong’ order, that is, bouncing between left and right elements. For example, \texttt{porder [1,2,3,4,5]} returns \texttt{[1,5,2,4,3]} and \texttt{porder [1.1,2.2]} returns \texttt{[1.1,2.2]}.” Implement \texttt{porder} subject to the constraints of Assignment II.
5. [55 points]
For this problem, N always refers to a natural number, so you never need to write N nat.
(a) Define inference rules for deriving less-than and inequality relationships between
natural numbers. The judgment forms are N₁<N₂ and N₁≠N₂.

(b) Prove that < is asymmetric. Theorem. ∀N₁, N₂, N₃: ((N₁<N₂ ∧ N₂<N₃) ⇒ N₁≠N₃)
(c) Define inference rules for deriving sums and remainders of natural numbers. The judgment forms are $N_1 + N_2 = N_3$ and $N_1 \% N_2 = N_3$. For example, 3 divided by 2 has a remainder of 1, so $S(S(S(Z))) \% S(S(Z)) = S(Z)$ is a valid judgment. Your definition of remainders may use judgments from Part (a) but not addition judgments.

(d) Prove that the difference between a number and its remainder has zero remainder. In case you need it, the following page provides additional space for your proof.

**Theorem.** $\forall N_1, N_2, N_3, N_4: ((N_1 + N_2 = N_3 \land N_3 \% N_4 = N_1) \Rightarrow N_2 \% N_4 = Z)$
[This page provides additional space for Problem 5(d).]
6. [10 points]
Consider the function unzip L. The SML documentation says that unzip “returns a pair of lists formed by splitting the elements of L. This is the inverse of zip for equal length lists.”

(a) Implement unzip subject to the constraints of Assignment I.

(b) Implement unzip subject to the constraints of Assignment II.