Assignment #2

Name: ______________________________ ID: ______________________________

This assignment has **4** questions, for a total of **25** marks.

Recall the following acronyms: SOS (structural operational semantics), COS (contextual operational semantics), SM (small step), BG (big step), CBV (call by value), CBN (call by name).

Question 1: **Safe untypable term** .... 6 marks

Write out a term that is safe (i.e., it does not reduce to fail) but that cannot be typed. Show the typing derivation until it fails (i.e., no rule is applicable) [3]. Also, show how the same term would reduce according to SOS-SM-CBV semantics [3].

Question 2: **Big step semantics for pairs and sums** . . . . 6 marks

Write the operational semantics rules for a big-step, call-by-value reduction for pairs [3] and sums [3]. Write the semantically correct ones only, but write them all.

Question 3: **Typing derivation** . . . . 6 marks

Show the typing derivation of these terms, with the following environment $\Gamma = f : \mathbb{N} \to \mathbb{N}$

- $t_1 = f\ (3 + 5) : \mathbb{N}$ [3]

- $t_2 = f\ ((\lambda x : \mathbb{N}. x + 2) 5) : \mathbb{N}$ [3]

Question 4: **Encoding** .... 7 marks

For each of the following constructs, create an encoding in STLC. Show that your encodings behave as the related construct by showing the reductions of your encoding using COS-SM-CBV. The intended semantics for the constructs is given after each construct in text.

- sequencing: $t ::= \cdots \mid t; t'$. Semantics: t is evaluated first, then t' is evaluated. [1]

- let-in: $t ::= \cdots \mid let\ x = t\ in\ t'$. Semantics: t is evaluated into a value v and then t' is evaluated for v in place of x. [1]

- arrays of length 4: $t ::= \cdots \mid [t, t, t, t]$. Values include arrays of values: $v ::= \cdots \mid [v, v, v, v]$. (no semantics for this case) [1]

- array field access: $t ::= \cdots \mid t.i \quad (i \in 0..3)$. Semantics: for $i \in 0..3$ we have that $[v_0, v_1, v_2, v_3].i$ returns v_i (show the encodings for at least two cases of i). [2]

- array update: $t ::= \cdots \mid t.i = t \quad (i \in 0..3)$. Semantics: for $i \in 0..3$ we have that $[v_0, v_1, v_2, v_3].2 = v$ returns $[v_0, v_1, v, v_3]$ (show the encodings for at least two cases of i). [2]