Assignment 2:
Defining and Implementing a Simple Language

COS441: Programming Languages
Fall 2004

Assigned: Wednesday, 22 September 2004
Due: in class on Wednesday, 29 September 2004

Submit Problem 3 online. All other problems should be turned in on paper at the beginning of class on the due date. There are three problems to this assignment totaling 100 points.

Problem 1 [10 points]
(a) [3 points] Exercise 6.2.1 from Ullman (p206)
(b) [5 points] Exercise 6.2.3 from Ullman (p206)
(c) [2 points] Exercise 6.2.5 from Ullman (p208)

Problem 2 [42 points]
(a) [11 points] Exercise 1 from 7.4 of Harper
(b) [20 points] Exercise 2 from 7.4 of Harper
(c) [11 points] Exercise 4 from 7.4 of Harper
(d) [10 points] (Extra Credit) Exercise 5 from 7.4 of Harper

Problem 3 [48 points] Starting from absyn.sml and names.sml you will complete the implementation of semantics.sml. You will use the SML/NJ compilation manager to compile the code. You should not need to modify any file other than semantics.sml.

(a) [8 points] Implement the ok relation described by 6.1 of Harper by finishing the definition of ok in semantics.sml. Implement the relation exactly as defined.

(b) [8 points] Implement evaluation relation ⇓ in 7.2 of Harper by finishing the definition of eval in semantics.sml.
Notice that the signature of eval states that eval must return a Absyn.exp. The final definition of eval should compile without any non-exhaustive match warnings from the compiler. You will have to use case in place of let in certain places and explicitly throw the Impossible exception. See the lecture slides for examples. The eval function should explicitly raise the Wrong exception when the function is not properly defined.

(c) [8 points] Reimplement the evaluation relation ⇓ in 7.2 of Harper by finishing the definition of eval’ in semantics.sml.
Notice that the signature of eval states that eval must return an int. The eval’ function should explicitly raise the Wrong exception when the function is not properly defined.
(d) [8 points] Implement the relation $\rightarrow$ in 7.1 of Harper by finishing the definition of $\text{step}$ in $\text{semantics.sml}$. The $\text{step}$ function should raise the exception $\text{Stuck}$ when the relation is not defined.

(e) [8 points] Implement the relation $\rightarrow^1$ using your answer from (d) in 2.1 of Harper by finishing the definition of $\text{stepsFinal}$ in $\text{semantics.sml}$.

(f) [8 points] Implement the relation $\rightarrow^n$ using your answer from (d) in 2.1 of Harper by finishing the definition of $\text{steps}_n$ in $\text{semantics.sml}$.

The file $\text{test.sml}$ includes a few simple tests that you can run against your solutions. Issue the command $\text{Test.allTests()}$ in the SML/NJ toplevel to run the tests after compiling your code. Simply passing all the tests does not guarantee you will get full credit! You are allowed and encouraged to share any test cases or testing code you write with other students in the course. However, you should not reveal any code in $\text{semantics.sml}$ to other students.