

Homework 4  
SNU 4541.664A Program Analysis  
Spring 2006

Prof. Kwangkeun Yi

**due: 5/6 24:00** via email to TA

**Exercise 1** “Collecting Interpreter (Reachability Interpreter)”  
Consider the imperative language C--:

$$\begin{array}{l} C \rightarrow \text{skip} \\ \quad | \quad x := E \quad | \quad *x := E \\ \quad | \quad C ; C \\ \quad | \quad \text{if } E \ C \ C \\ \quad | \quad \text{while } E \ C \\ E \rightarrow \text{readint} \quad | \quad n \quad (n \in \mathbb{Z}) \\ \quad | \quad E + E \quad | \quad - E \\ \quad | \quad E < E \quad | \quad E = E \quad | \quad E \ \&\& \ E \\ \quad | \quad x \quad | \quad *x \quad | \quad \&x \end{array}$$

Implement a collecting interpreter `eval` for the above language C--:

$$\text{eval} : \text{PGM}_C \rightarrow (\text{PGM}_C \rightarrow 2^{\text{Mem}})$$

that executes input program  $P$  with an empty memory and returns the table that has collected all the memories occurring right before executing each command inside  $P$  (a table from each command inside  $P$  to the set of memories that has occurred right before executing the command).

The interpreter shall be based on the semantics definition that you have done for HW2.

I recommend you to use nML ([ropas.snu.ac.kr/n](http://ropas.snu.ac.kr/n)). As an encouragement, C-- parser in nML will be provided.  $\square$