;;; FOR USE IN HOMEWORK 1

;;;==================================================================================

;;; a macro to do FOR loops ;; equivalent to Java for-loop control: for (int var=start, var<=stop; var+=update) body

(defmacro for ((var start stop update) &body body)

(let ((gstop (gensym)) ;; generate new symbols, GUARANTEED to be new; prevents capture

(gupdate (gensym)))

`(do ( (,gupdate ,update) ;; needed so that the update expression is evaluated just once

(,var ,start (+ ,gupdate ,var)) ;; needed so that the stop expression is evaluated just once

(,gstop ,stop))

((> ,var ,gstop))

,@body)))

;; EXAMPLE

;; CL-USER> (for (i 1 6 1) (print i)) ;; equivalent to Java for-loop control: (for int i=1, i<=6; i++)

;;

;; 1

;; 2

;; 3

;; 4

;; 5

;; 6

;; NIL

;;;==================================================================================

(defun comparefibonaccis()

(for (i 10 35 5) ;; for (int i=10; i<=35, i+=5) ...

(format t "TAIL REC FIBONACCI ~a~%" i)

(time(fibonacci-TR i))

(format t "FIBONACCI ~a~%" i)

(time(fibonacci i))

(format t "=======================================================~%")

)

)

;;;==================================================================================

;;; END