```
;; Function To Be Tested: BROWSER (Part I) (Program Analysis)
;;
;; Source: Lisp Library Modules Manual (Lyric Beta Release 2)
;;              Browser, Page 11
;; Section: Program Analysis (Library)
;;
;; Created By:   John Park
;;
;; Creation Date: March 10, 1987
;;
;; Last Update:   March 12, 1987
;;
;; Somewhat Repaired:  June 16, 1988
;; Munged more:                June 24, 1988 by Rene P. S. Bane
;;
;; Filed As:       {ERIS}<test>env>program-analysis>hand>browser-part1.u
;;
;;
;; Syntax:       (BROWSER T/NIL)
;;
;; Function Description: BROWSER modifies the SHOW PATHS command of Masterscope so that
;; the command's output is displayed as an undirected graph.  It creates a new window
;; for each SHOW PATHS command, but will reuse a window if that window has an earlier
;; instance of the same SHOW PATHS command displayed in it.  Part 1 of this test is
;; to determine if masterscope is unaffected when the BROWSER is not enabled or
;; (IL:BROWSER NIL). Part I also tests to see if grapher is automatically loaded
;; by browser.

;;
;; Argument(s):  T or NIL (SEE Browser documentation)
;;
;; Returns:  IL:MSPATHS
;;
;; Constraints/Limitations: ; Part 1 of this test is to determine if masterscope is
;; unaffected when  the BROWSER is not enabled or (IL:BROWSER NIL)
;; Part 2 tests masterscope modification by BROWSER.  This is test part 2, which
;; is stored in {eris}<lispcore>test>program-analysis>browser-part2.u
;; Part 1 test must be run first before Part 2 test since the former loads functions
;; utilized by the latter.  Since Browser calls LAYOUTFOREST module of GRAPHER to
;; display the graph at a user-designated location, user interface is required and
;; testing will not be totally automatic.  Instructions will be given for user input
;; during testing. The test will utilize do-test and the interlisp function bksysfuf.
;; Comments or messages are incorporated within each command file, which will be run
;; by using the function bksysbuf. Each test setup is titled "BROWSER-TEST-SETUP",
;; which executes the command  string. The do-test form within the command file will
;; return T or "testfailed"  This test file requires MASTERSCOPE, TEDIT, BROWSER, and
;; GRAPHER packages
;;
;; The tree structure of the functions being analyzed are as follows:
;;
;;                               Top-GFuntion
;;                                       |
;;            GFun-A-------------------------GFun-B
;;                |                                            |
;;    -----------------------              ---------------------
;;     |          |           |                |          |        |
;;  GFun-A1  GFun-A2   GFun-A3          GFun-B1         GFun-B2         GFun-B3
;;                                                        |
;;                                               ---------------
;;                                               |                  |
;;                                        GFun-C1             GFun-A1
;;
;;
;; Messages will be printed before each command in the command files is executed
;; for user monitoring.   Test result is logged on
;; {eris}<lispcore>test>program-analysis>browser.report

(DO-TEST "BROWSER-TEST-SETUP"

(SETQ TEST-SUCCEEDED T)
(DEFUN PASS-FAIL (COMMAND-LANGUAGE TEST-ITEM)
        (IL:IF (NOT TEST-ITEM)
                IL:THEN (FORMAT *ERROR-OUTPUT* "Test ~s failed~%" COMMAND-LANGUAGE)
                           (SETQ TEST-SUCCEEDED NIL)
                IL:ELSE "Pair-fect-o"
                )
)

(IL:FILESLOAD (IL:SYSLOAD) MASTERSCOPE BROWSER GRAPHER)
```

```
(SETQ BROWSERW (IL:CREATEW '(100 100 325 90) NIL NIL T))

(IL:PAGEHEIGHT 0)
; This part of test determines if the variables BROWSERFORMAT and BROWSERBOXING
; are bound.
(PASS-FAIL "Browser variables bound" (AND (BOUNDP 'IL:BROWSERFORMAT) (BOUNDP 'IL:BROWSERBOXING)))

; Reinitialize and Define functions to be analyzed....
(IL:MASTERSCOPE '(ERASE))
(DEFUN TOP-GFUNTION NIL (AND (GFUN-A) (GFUN-B)))
(DEFUN GFUN-A NIL (OR (GFUN-A1) (GFUN-A2) (GFUN-A3)))
(DEFUN GFUN-B NIL (OR (GFUN-B1) (GFUN-B2) (GFUN-B3)))
(DEFUN GFUN-A1 NIL T)
(DEFUN GFUN-A2 NIL NIL)
(DEFUN GFUN-A3 NIL T)
(DEFUN GFUN-B1 NIL (AND (GFUN-C1)(GFUN-A1)))
(DEFUN GFUN-B2 NIL NIL)
(DEFUN GFUN-B3 NIL T)
(DEFUN GFUN-C1 NIL NIL)

; Start analyzing functions in gtop-function and others...
(IL:MASTERSCOPE '(ANALYZE TOP-GFUNTION))
(IL:MASTERSCOPE '(ANALYZE GFUN-A))
(IL:MASTERSCOPE '(ANALYZE GFUN-B))
(IL:MASTERSCOPE '(ANALYZE GFUN-A1))
(IL:MASTERSCOPE '(ANALYZE GFUN-A2))
(IL:MASTERSCOPE '(ANALYZE GFUN-A3))
(IL:MASTERSCOPE '(ANALYZE GFUN-B1))
(IL:MASTERSCOPE '(ANALYZE GFUN-B2))
(IL:MASTERSCOPE '(ANALYZE GFUN-B3))
(IL:MASTERSCOPE '(ANALYZE GFUN-C1))

; Browser is now loaded and activated
; Part 1 of this test is to determine if masterscope is unaffected when
; the BROWSER is not enabled or (IL:BROWSER NIL)
(IL:BROWSER NIL)

; Browser is now deactivated ...
; This will cause masterscope to display graphs in a teletype mode
; or in the exec.
; show paths should display the following path, which should look like;
; 1.gfun-a1 gfun-a top-gfuntion
; 2.        gfun-b1 gfun-b top-gfuntion
(DRIBBLE '{CORE}PATHS)
(IL:MASTERSCOPE '(SHOW PATHS TO GFUN-A1 FROM TOP-GFUNTION))
(DRIBBLE)
; analyzing the file that contains the masterscope interaction (show paths)
(SETQ PATHS (OPEN "{CORE}PATHS"))

(LET (NEXT-LINE)
        (IL:WHILE (AND (NOT (EQ 'EOF
                                            (SETQ NEXT-LINE (READ-LINE PATHS NIL 'EOF))))
                                (NOT (SEARCH "top-gfuntion" NEXT-LINE :TEST #'STRING-EQUAL))))
        (PASS-FAIL "Show paths (would-be graph)"
                (AND

                        (STRING-EQUAL "1.GFUN-A1GFUN-ATOP-GFUNTION" (DELETE #\Space NEXT-LINE))
                        (STRING-EQUAL "2.GFUN-B1GFUN-BTOP-GFUNTION" (DELETE #\Space (READ-LINE
PATHS NIL NIL)))
                ))
)  ; close let

(CLOSE PATHS)
(DELETE-FILE '{CORE}PATHS)

TEST-SUCCEEDED

)

STOP
```