```
;;MAKE-HOMOGENEOUS-4-BY-4
;; By Peter Reidy
;; Filed as {ERIS}<LISPCORE>TEST>DISPLAY>MATMULT>MAKE-HOMOGENEOUS-4-BY-4.TEST
;; Syntax: (MAKE-HOMOGENEOUS-4-BY-4 &key A00 A01 A02 A03 A10 A11 A12 A13 A20
A21 A22 A23 A31 A32)
;; Function description: returns a 4-by-4 matrix of element-type single-float;
some elements can be specified in the keywords; the 2, 2 element is always
1.0.  Other elements default to 0.0.
;; Arguments: keywords: where x and y are the two digits in the keyword, the
corresponding matrix element will be set to the keyword value.
;;
(do-test-group make-homogeneous-4-by-4-group
      :before
      (il:load? '{eris}<lispcore>test>display>matmult>matmult-test.source)
;;
      (do-test 4-by-4-simple-case
            (let ((matrix44a (il:make-homogeneous-4-by-4)))
                  (2dtest matrix44a 4 4 #'(lambda nil (deftest44 matrix44a
'(a33))))
            )
      )
;;
      (do-test 4-by-4-with-keys
            (let  ((randnum (- (random most-positive-single-float)))
                   (matrix44 (il:make-homogeneous-4-by-4
                              :a20 most-positive-single-float
                              :a21 Most-negative-single-float
                              :a10 randnum
                              :a01 0.0
                              :a00 4761
                              :a02 1.0
                              :a02 1000.0
                              :a11 100.001
                              :a12 1947.0
                              :a13 7491.1947
                              :a22 randnum
                              :a23 (- randnum)
                              :a30 10.10
                              :A31 20.2002
                              :a32 pi
                              :a03 most-positive-single-float)
                   )
                   (positions44 '(a00 a01 a02 a03 a10 a11 a12 a13 a20 a21 a22
a23 a30 a31 a32))
                  )
                  (2dtest matrix44 4 4
                        #'(lambda nil
                              (deftest44 matrix44
                                    (append positions44 '(a33))
                              )  ; deftest44
                          )  ; lambda - end deftest argument
                        positions44  ; positions
                        (list 4761 0.0 1.0 most-positive-single-float randnum
100.001 1947.0 7491.1947 most-positive-single-float Most-negative-single-float
randnum (- randnum) 10.10 20.2002 pi)  ; values
                  )  ; 2dtest
            )  ; let
      )
;;
      (do-test 4-by-4-error (expect-errors (error) (il:make-homogeneous-4-by-4
:a00 #c(0 3))))
)
END
```