

TRAJECTORY-FOLLOWER documentation

D. Austin Henderson, Jr.

Initially released: April 23, 1983

Last revised: April 23, 1983

<LISPUSERS>**TRAJECTORY-FOLLOWER** (and **.DCOM**) provides a function which causes a "snake" to crawl along a trajectory. Comments on both interface and functionality are welcomed.

TRAJECTORY.FOLLOW (KNOTS CLOSED N DELAY BITMAP WINDOW) The trajectory is specified by **KNOTS** (a set of knots) and **CLOSED** (a flag indicating whether it is an open or closed curve). **N** is the length of the snake in points along the curve. **DELAY** is the time (in milliseconds) between each move along the curve; **DELAY = 0** or **NIL** means go as fast as you can. **BITMAP** is the brush to be used at each point in creating the snake. **WINDOW** is the window in whose coordinate system the knots are given and in which the snake is to be drawn; if **NIL**, then the **SCREEN** bitmap is used. The snake is moved by **INVERTing** the bitmap at the points along the curve, and then **INVERTing** the bitmap back out again.

A demonstration function is also provided with the package:

TRAJECTORY.FOLLOWER.TEST () Interacts with the user through prompting in the promptwindow to gather up arguments for **TRAJECTORY.FOLLOW** and then carries it out. Closed curves are snaked around repeatedly until the left shift key is depress when it reaches the curve's starting point.

The internal functions used by this package are also available for use. They are:

TRAJECTORY.FOLLOWER.SETUP (WINDOW N DELAY BITMAP) Initializes drawing variables.

TRAJECTORY.FOLLOWER.POINT (X Y WINDOW) Defines the next point on the curve. Note that the argument structure of this function is appropriate for use as a **BRUSH** with the curve drawing functions **DRAWCURVE**, **DRAWCIRCLE**, and **DRAWELLIPSE**. (For an example, see the demonstration function **TRAJECTORY.FOLLOWER.TEST**.)

TRAJECTORY.FOLLOWER.WRAPUP () Finishes the job after all the points have been defined.