GPROLOG SOURCE CODE FOR INTERACTIVE KLUMPENHOUWER NETWORKS

By Israel Neuman and Alberto M. Segre

This source code is part of a generative compositional tool which is based on the Klumpenhouwer Networks analytical method. This tool, which is under development, will ultimately be embedded as a PD-extended external object for real-time performance. For a detailed description of this tool please see the paper:

> Neuman, Israel. “Generative Tools for Interactive Composition: Real-Time Musical Structures Based on Schaeffer’s TARTYP and on Klumpenhouwer Networks,” accepted for publication *Computer Music Journal* 38 no. 2 (2014)

The compilation of this source code requires the installation of the GNU gprolog (version 1.4.0 or later) available as open source at: http://gprolog.univ-paris1.fr

Using the terminal/command line, load this source code with an interactive gprolog session as shown in the sample session below.

```
$ gprolog
GNU Prolog 1.4.0
By Daniel Diaz
Copyright (C) 1999-2011 Daniel Diaz
| ?- consult('klump6.pl').
compiling /Users/israel-neuman/Desktop/klump/klump6.pl for byte
code...
/Users/israel-neuman/Desktop/klump/klump6.pl compiled, 159 lines
read - 26584 bytes written, 17 ms

(3 ms) yes
| ?- testa(L).

L = [[4,9,3],[10,2,9],[5,2,4],[10,3,9],[6,1,7]] ?

(1 ms) yes
| ?- listing(aKNet).

aKNet([[3, 10, 4], [4, 9, 3], [10, 2, 9], [5, 2, 4], [10, 3, 9],
[6, 1, 7]], aknet0).

yes
| ?- listing(hyperI).

hyperI([3, 10, 4], [4, 9, 3], 2, aknet0).
hyperI([3, 10, 4], [10, 2, 9], 1, aknet0).
hyperI([3, 10, 4], [5, 2, 4], 8, aknet0).
hyperI([3, 10, 4], [10, 3, 9], 2, aknet0).
hyperI([10, 3, 9], [6, 1, 7], 8, aknet0).
hyperI([5, 2, 4], [6, 1, 7], 2, aknet0).
hyperI([10, 2, 9], [6, 1, 7], 7, aknet0).
hyperI([4, 9, 3], [6, 1, 7], 8, aknet0).

(1 ms) yes
| ?- listing(hyperT).

hyperT([3, 10, 4], [6, 1, 7], 6, aknet0).
hyperT([5, 2, 4], [10, 3, 9], 6, aknet0).
hyperT([10, 2, 9], [5, 2, 4], 5, aknet0).
hyperT([10, 2, 9], [10, 3, 9], 11, aknet0).
hyperT([4, 9, 3], [10, 2, 9], 1, aknet0).
hyperT([4, 9, 3], [5, 2, 4], 6, aknet0).
hyperT([4, 9, 3], [10, 3, 9], 0, aknet0).

yes
| ?- findPath([3,10,4],[6,1,7],Path,aknet0,4,2).

Path = [[3,10,4],[6,1,7]] ? ;

Path = [[3,10,4],[6,1,7],[10,3,9],[6,1,7]] ? ;

Path = [[3,10,4],[6,1,7],[10,3,9],[3,10,4],[6,1,7]] ?

yes
| ?-
```