Goal: To creatively represent computer programs by prototyping physical, personal realizations of them. Programming as a hybrid-digital/analog kinetic sculpture. Blends art, design, and formal process representation.

Method: We will employ the visual language of Max/Msp which is in wide use by computer musicians, video jockeys (VJs), artists, and designers. Programs will be prototyped in Max/Msp and then made physical by mapping the visual programs to Arduino-based personalized artifacts.

Philosophy: Computer programs can be written. Languages such as C, C++, Java, and Python are primarily encoded in writing—using textual representation. Other kinds of languages employed by engineers as well as game designers are visual. There is a gradual shift toward hybrid text/visual languages. What would computer programs be like if they were interactive and tangible rather than written or drawn? Programming would become more of an art with personalized expression emphasized. Reference Aesthetic Computing links on the web for past history of this movement.

Requirements: A prior knowledge of programming and a willingness to explore the boundaries of art, design, and computer science.

Costs: Costs for the class are estimated between $200 and $250 to cover the cost of the student version of Max/Msp (optional since it is in the ATEC labs), your Arduinos, and your craft materials in which to embed the Arduinos. Students are free to choose different Arduino DIY kits or designs. The primary software interface for the hardware will be through the Arduino IDE.

Instructor and TA: Dr. Paul Fishwick, ATC 3.206 (paul.Fishwick@utdallas.edu) and Karen Doore (TA), (kdoore@utdallas.edu)

Grading: Project-based, teams will be made optional based on the number of students.