MAS.714 Systems and Self
Prof. Turkle
Prof. Resnick

1 November 2002
Final Project Proposal

Megan Galbraith

Many of the educational tools developed for the computer are designed for children. Bright colors, animals and other cartoon-like characters, associations with toys and children's objects, silly sounds, and graphical elements provide the means for users to explore the complex and interesting worlds of computation. These educational tools are very powerful and do not necessarily need to be limited to children in order to be effective. However adults are less likely to be drawn to or even exposed to these programs because of their age.

Many adults, young and old alike, do not have strong mathematics or science backgrounds and therefore think very differently about computers and computation than adults who come from strong technical backgrounds. However this is not to say those who do not have the same strengths will not (or do not) find computers useful in their daily life, work, or hobbies. In general, people are becoming more adept to using computer software programs or familiarizing themselves with computer operating systems and "simulation" programs in order to accomplish tasks such as word processing, design, email, or commerce. They operate on an abstraction level far from the internal structure, design, and operation of a computer as machine. Concepts used in programming or hardware design, for example iteration or the execution of algorithms, are foreign to these adults. Their understanding of the computer is one of exploration and interaction rather than the execution of procedures, the use of logical gates, or the development of structured methods and chunks of code.

For my final project I would like analyze a program I am writing for my thesis, Fuzzzzy, as a few adults with weak math and science backgrounds use the program and attempt to program a small hardware device. The software is a web-based application that lets users program small devices (such as rabbit2000 microprocessors, PIC chips, etc.) using fuzzy-logic based programming rather than procedural programming.

I would like to look at several components of the software in light of the discussions we have had in class. The first component is related to multiple styles of thinking (October 22). Does the Fuzzzzy System help the users think about and conceptualize their interactions with hardware devices or programming languages? How does it help them think about the design and control of devices, and does this follow conventional "accepted" methods of control and design? How do these adults approach the task of writing a computer program and does Fuzzzzy fall short or exceed expectations in any way?

Another component is related to constructional design (September 24). One context for using Fuzzzzy that I am focusing on in my thesis is to aid fashion designers who would like to build computational garments. How is it helpful for the adults who use Fuzzzzy for my project think of their work in a broader context rather than merely as a piece of electronics hardware?

A third component is to view Fuzzzzy through the framework of simulation and it's discontents (October 29). Fuzzzzy is a program that in essence simulates the control and behavior of small hardware devices, and it operates on several levels of abstraction away from the "innards" of the devices, of the computer, and of the attached sensors and peripherals. It also "simulates" computer programming in the sense that the procedures and methods associated with traditional programming are hidden from the user. Control of device behavior is accessed indirectly because users program the devices with a non-procedural language that is inherent to the fuzzy logic computation that controls the execution of the programs. This intentional use of simulation in the design of Fuzzzzy has it's pitfalls, therefore I would like to look seriously at how this simulation of hardware assists or affects the way the users learn about, use, and understand both programming and hardware control.