

Project Ideas

Joseph D. Kulisics

2005 May 1

Following the line of reasoning in my submissions for this course and drawing on my past observations about learning, I have developed several preliminary project ideas. The ideas that I shall outline in this paper cover three areas, human-computer interaction, environments, and lessons illustrating powerful ideas.

Before I begin a discussion of human-computer interaction, I would like to motivate my ideas with a few examples. There are four examples that are of particular interest to me, two that illustrate a concept that I call structural fixedness and two that illustrate the conceptual connectedness of a domain of knowledge and the impact of connectedness on discovery.

In the teaching assistant training seminar last quarter, a student presented an interesting puzzle, a favorite kind of presentation in the class. In the puzzle, Martians take ten humans back to Mars. Before the humans are placed in individual cells in which they will be completely isolated from one another, they are allowed to discuss a strategy. Every five minutes, a prisoner will be selected at random and taken to a switch room where there is a single light bulb connected to a switch, which is initially in the off position. The prisoner will be given the opportunity to change the position of the switch. A prisoner can declare at any time that all prisoners have visited the switch room, and if the prisoner is correct, everyone will be freed. If the prisoner is incorrect, everyone will be executed. A prisoner is never obligated to flip the switch and never obligated to make a declaration. What strategy can the prisoners follow to ensure their safe release? In solving this problem, I believe that many people initially identify the light bulb as both a communication device and a state keeping device, neither of which is a natural function of the light bulb leading me to distinguish the phenomenon from the functional fixedness described by Bruner in *Toward a Theory of Instruction* where subjects fail to see an unconventional use of an object in their environment. The puzzle confounds many people because they fail to interrogate these basic assumptions about the elements of the puzzle, and they futilely try to make the light bulb, which can only hold two states, represent ten states. Another assumption that becomes an obstacle to solving the problem is that there should be symmetry in the rules followed by each of the prisoners. The combination of these assumptions deadlocks efforts to develop a solution. By systematically listing and violating the assumptions

inherent in failed attempts to solve the puzzle, new solutions can be generated. For example, by recognizing that the light bulb cannot hold enough states to solve the problem, a subject might search for another state-keeping element in the puzzle and realize that the prisoners, themselves, can count without bound. In addition, if prisoners do not all follow the same set of rules, the idea of using the light bulb only to signal a single prisoner who acts as a counter becomes practically self-evident.

In a similar way, a problem in Abelson and Sussman's *Structure and Interpretation of Computer Programs* challenges readers. The problem says that given a linked list, detection of a cycle in the list can be done trivially in space and time linear in the size of the list—simply build a list of addresses of nodes in the list as you traverse it, and when a duplicate address occurs, report the existence of a cycle—and the problem goes on to ask the reader to develop a scheme for detecting a cycle in a linked list that requires only constant space and at worst time linear in the size of the list. I believe that most readers become stuck on this problem, which Abelson and Sussman describe as requiring ingenuity to solve, only because they confine themselves to the constraints of the solution that they already know. The solution in the problem description requires using a single pointer to walk the list one node at a time. By violating the assumptions that only one pointer is used in traversing the list and that pointers only walk the list one node at a time, a solution in which pointers traversing the list at different speeds and in which a loop can be detected when one pointer laps another emerges.

In both of the previous examples, the solutions to the problems are quite simple, and that these problems pose great difficulties for people suggests room for improvement in the training of people for critical thinking. After all, the solutions to many puzzle problems turn on carefully representing the information or finding an especially appropriate tabulation of information in the puzzle, but these two examples can be solved without picking up a pencil or paper. The puzzles require no expert knowledge of any domain. They require no mathematical sophistication. There is no insight into the solution to be had by reducing the size of the problem instance. Solving these problems requires only a simple act of invention, and my contention is that invention can be made systematic.

My third example involves a piece of classical reasoning, the proof of the Pythagorean Theorem. A common proof of the Pythagorean Theorem involves inscribing four identical copies of a right triangle inside a square so that the sides of the square are of length $a + b$. By calculating the area of the square directly as $(a + b)^2$ and indirectly as the sum of the areas of the discrete regions inscribed in the square, equating the results, and simplifying, one obtains the formula $a^2 + b^2 = c^2$. How does one discover this proof? My claim is that the proof is discoverable by exploring the deep, logical connections between concepts in the problem. The right triangle is distinctive for having a right angle, and right angles are characteristic of rectangles in general and squares in particular. In addition, the quantities a and b , representing the lengths of

the adjacent sides of the right triangle are related by the formula for the area of the triangle, $A = \frac{1}{2}ab$. The square represents the fundamental, geometric model for area, and the analytical quantity a^2 is connected to geometry and the square through the concept of area. These connections, together, suggest the possibility of exploring the relationship between the quantities a , b , and c geometrically with the square.

Similarly, in computer science one often has occasion to describe processes by recurrence relations. A recurrence relation describes the terms in a sequence by a set of initial values and a rule for generating new values from previous terms in the sequence. As an example of a generating function, consider $a_0 = 1$, $a_n = 2 \cdot a_{n-1}$ for $n > 0$ and n an integer. This rule can be used somewhat clumsily to calculate any term in the sequence by working forward from an initial value. The problem with using this recurrence relation is that the time required to compute a term in the sequence grows linearly in n . We would like to be able to calculate a value for a sequence term directly from n . By inspection we know that the above recurrence describes the sequence whose general term is given by the formula $a_n = 2^n$ for $n \geq 0$, n an integer. This rule depending only on the index, n , is called a closed form analytic expression. How do you derive this expression?

While we don't know how to calculate closed forms for sequences represented by recurrence relations, we know how to calculate closed forms for some series. Since there is a deep connection between sequences and series—a series is the sum of the terms of a sequence—the possibility of exploring closed forms for recurrences through series arises, and in the method of generating functions, the connection of sequences to series is plumbed by constructing series equations from instances of a recurrence, manipulating the series equations, using the closed forms of the series to repackage and move coefficients in the equations, and relating the coefficients of the series equations to discover a closed form for the terms of the sequence represented by the recurrence.

In both of these pieces of classical reasoning, the driving force behind the discoveries is the deep connection of the concept under investigation to other ideas. By exploring the connections systematically, one can rediscover the original constructions, and I believe that the original discovery of these constructions proceeded along exactly these lines.

In his introductory lecture, Professor Kay suggested that the printing press caused a revolution in thought by changing the density of arguments and that the computer is causing a similar revolution in thought by introducing new styles of argument such as the simulation, and extrapolating from the position of Professor Kay, one can view the graphical user interface as joining the simulating power of the computer to the human-computer interface to create a virtual world in which the user works. One wonders if there are other properties of the new medium that can be combined with the human-computer interface in a virtual world to enhance learning, discovery, and work.

In the introductory lecture, Professor Kay observed in detail that the printing press changed arguments because for the first time in history, the au-

thor would be able to view galleys of his manuscript and redact the manuscript to include page numbers. Page numbers as indirect references to previously developed pieces of an authors argument increased the density of arguments. I would like to suggest that in a manner analogous to the way that the indirect reference arrived as a style of argument because of the printing press, the computer brings a direct referencing power witnessed by the hypertext document that carries the potential to further transform and condense arguments. In addition, the computer has the ability to index and relate data dynamically as in a database enhancing arguments and presentation beyond the direct expression of an author. Can these powers of the computer find a home in the human-computer interface, and can the integration of these powers of the computer into the human-computer interface enhance learning, working, and discovery? I believe that these new powers of the computer as a medium can enhance literacy and argumentation.

Before attempting to augment the human-computer interface to enhance learning and working and to combat structural fixedness and support the process of discovery, and we must analyze the state-of-the-art in human-computer interaction to determine where we might coherently grow the interface. In looking at existing graphical user interfaces and working environments, three observations come to mind. The first observation is that the virtual world of the computer is inhabited by a single intelligent agent, a solitary user. The second observation is that there is typically only one relationship between objects in the virtual world, containment, which if you've ever seen a child play with lettered blocks, you'll realize exists side-by-side with conjoining as a way to relate objects in the real world. The third observation is that the view of the virtual world is flat compared to the real world, which presents hierarchical views in which a microscopic view exists below the visible world and a macroscopic view exists above the visible world. Of course given my limited knowledge of existing interfaces and research, existing systems and research into human-computer interaction may have made moot some of these observations, nevertheless I shall risk rehashing the work of others to continue my line of development hoping that something worthwhile will come of the effort.

Of these observations, the one that seems to hold the greatest potential for development and enhancement of the interface is that the current interfaces simulate a flat world, and I propose in my project to integrate into the human-computer interface a simulated hierarchy of views on objects and to create the analog of a microscopic view of the virtual world that will serve as a fabric hosting the deep connections between objects in the learning environment. I would like to make a model of the way that I think that an interface would look that offered idioms for use to the user suggesting a hierarchical view of the virtual world and putting at the disposal of the user the relationships hosted in the virtual, microscopic view of the world. I believe that a virtual hierarchy of views of the world represented in the computer can serve the aim of assisting discovery and address one of my concerns in the examples that I outlined above. As for the goal of fighting structural fixedness, an interface contributes

to the structural fixedness of users to the degree that the interface elides the assumptions that the users make as they work, so preventing structural fixedness requires making users explicitly list the assumptions that they use as they construct solutions to problems, something best accomplished at the level of a programming language. Consequently, I have no suggestion for enhancements to the human-computer interface to reduce structural fixedness beyond organizing programming languages in such a way that assumptions in the form of data declarations, function prototypes, and interfaces be grouped and isolated from the implementations of behaviors.

I would also like to propose as future projects two working environments. As an undergraduate student taking physics, I had the great pleasure to learn from the classic textbook, *An Introduction to Mechanics*, by Kleppner and Kolenkow, and the textbook uses the instructional device of systems of pulleys and wheels to teach the student to write and solve systems of constraint equations and to develop intuition in the reader about forces. Though I was never very good at physics, I recognized the pedagogic value of the examples and problems with pulleys and wheels, and I often wished that I had a way to gain some first hand experience with such systems. I think that a complete pulley and wheel environment in which a user can specify parameters of objects such as friction and in which the user can measure by the application of sensors to elements of the constructions the forces at work in the environment would be an invaluable aid to teaching secondary and post-secondary students of physics. In the same way, I believe that a simulated wave tank environment could be useful in teaching students about wave phenomena.

Finally, I think that there are three powerful ideas in the sense that we have been treating powerful ideas in class that are ripe for developing into lessons targeting middle school students. The first powerful idea is that constant chance of decay of a particle in a mass is representable as a half-life and *vice versa*. Unfortunately, I believe that the idea of a half-life for a mass is usually introduced without reference to phenomena at the level of particles, and consequently, the average student acquainted with the idea of half-life has no idea that it is equivalent to the notion that there is a fixed chance of decay for each particle in a mass. The second powerful idea is that many processes can be completely characterized by a recurrence relation, the classic example of which is the Fibonacci recurrence, which characterizes many natural, growth processes, for example, giving the human body its characteristic proportions. The Fibonacci recurrence developed classically from the study of toy problems of rabbit reproduction, and I believe that one could develop a set of lessons around rabbit reproduction to demonstrate the idea of recurrences and build from such basic lessons to demonstrations of the role of the Fibonacci recurrence in shaping the proportions of living things. Finally, many people seem to labor under the mistaken though understandable assumption that in dynamic, nonlinear systems, small changes in the initial conditions of the system result in only small changes in the way that a system develops over time. At the beginning of the twentieth century, Henri Poincare anticipated in his work with

differential equations that nonlinear systems could exhibit a sensitive dependence on initial conditions, and the result was later observed and popularized by Edward Lorenz in his work with models of weather. Lorenz actually observed thanks to computer simulations that outcomes of weather forecasting diverged widely over only small differences in the initial conditions of the system and popularized his results in a lecture entitled *Predictability: Does the Flap of a Butterfly's Wings in Brazil set off a Tornado in Texas?* The lesson that one should take from the work of Poincare and Lorenz is that deterministic behavior is not the same as predictability—because of the error intrinsic in measure, one can never make good predictions on a large timescale—and since computer simulation popularized this idea, a fine example of the new style of argument in the computer age, it is an excellent candidate for development as a lesson on a powerful idea. I have vague ideas in mind for how to arrange these lessons, and I intend to look into them and try my best to make fun demonstrations of the ideas, demonstrations that like the demonstrations that we have already seen in class vertically integrate enactive, iconic, and symbolic learning.

Thank you,
Joseph D. Kulisics