Q11. From Radical Constructionism to Problem-Based Learning: A radical constructionist approach to learning gives students maximum freedom to control their activities and learning. As we move from radical to moderate constructionism to project-based learning to problem-based learning, we gradually add more structure and gain more control over learning outcomes.

a. What are the advantages of less structured learning environments? Of more structured ones? Discuss from both the points of view of learning outcomes and of assessment strategies.

b. Where along this continuum do you situate your own work? What are the advantages and disadvantages of the choices you have made?

c. What would it mean to move your work to a very different spot on this continuum? Discuss advantages and disadvantages.

The choice of student freedom is motivated by both philosophical and practical concerns. Constraining what students can learn can inhibit the development of higher-order knowledge management strategies [11] and even enforce a curriculum that is personally offensive to them [5]. On the other hand, certain knowledge and skills are important and there are limits to what a teacher can use to teach specific topics.

Advantages of More Structured Learning Environments

More structured learning environments primarily provide a great deal of control over what is being learned. This has significant advantages. It allows the teacher to leverage much greater amount of pre-built educational resources. Because the outcomes are specific, it is also usually possible to assess learner’s achievement in standardized reliable fair way. Because the goals are specific it enables others (parents, governments, expert teachers) to control education.

An example of the sort of resource that can be provided when learning is structured is something like Anderson’s cognitive tutors [4]. This tutor is a resource that no teacher could develop by themselves. It employs a sophisticated model of the learner’s understanding. Because its goals are specific, it can provide contextualized hints on every problem. It can give each student as much practice as they need, at a speed customized for them. Even without going that far, moderately structured environments can have a variety of useful tools. Koldner’s ARCHIE system for example provides help to architecture students, as long as they are attempting to design a library [7]. Wilensky [16] discusses the benefits of having pre-built simulations. Though Papert learned from gears [9] he advocates the benefits of a general purpose Logo toolkit (that while eminently more flexible than a cognitive tutor nonetheless focuses attention on mathematical concepts he wants students to learn). In general, the more structure the more helpful tools exist.

Testing achievement is easier in more structured environments. Cognitive tutors can provide instant feedback that has a demonstrably significant effect on learning speed. In a slightly less structured environment, a set of external experts are used to determine the quality of assignment submissions [1]. When one student can build a simulation of termites and another a simulation of traffic patterns [10] it can be difficult to assess which shows the greater understanding.
Advantages of Less Structured Learning Environments

There are two main advantages of less structured learning environments. The first advantage is that they can teach broader concepts than can be done in more constrained cases. The second advantage is that less structured environments have greater potential to appeal to students’ interests.

To see that less structured environments can teach broad concepts, consider learning in everyday life. This is obviously a very unstructured environment but most everyone would agree that a person can’t learn to be a baseball player, programmer, or research scientist without actually performing in practice. Lave and Wenger [8] provide some explanation for the supports that exist in everyday social practices for learning. This is not to say that broad concepts can only be learned in unstructured environments but its clear they can be.

In highly constrained context some things are difficult to teach. I can read how Maxwell derived his famous equations [15] but if the goal is to learn how to design experiments it would be wise to let me design some of my own. Even better would be as in DBR [6] where (as Maxwell did) I first ‘mess about’ with semi-ad-hoc experimentation and then built more specific experiments to confirm exact numerical relationships. If the goal is to teach a student to take control of their own learning process, guiding with a teacher-built curriculum will be counterproductive. Even asking students to read certain background literature may significantly constrain the questions students think to ask [11].

With a structured learning environment, students may find it difficult to connect what they are learning with their personal goals. In Papert’s famous illustration of the gears [9], it was a personal interest in automobiles that led Papert to put forth the effort to understand how the gears in a transmission worked. If students feel that their education is in some way counter to their goals, the can even adopt strategies to avoid learning things they have a natural aptitude for [5]. Because every student is not motivated the same things even curricula with polished motivational content can be boring. In an unstructured environment like radical constructionists advocate, students can explore their own interests (with the help of some carefully constructed toolkit that teaches some key idea).

My Work

I am interested encourage students to see themselves as computer scientists. I was very cognisant in beginning my work of the negative affects of establishing a single identity as a “true” example of a computer scientist [14]. To this end, I developed several different identities, matched the students with these identities, and then (with the help of mentors, class presentations, and interest reports) attempted to encourage students to affiliate with the identity I selected.

I would classify this intervention as fairly highly structured – approximately on the level of a DBR approach. The main flexibility comes from the mentors: it is their job to interpret the students letters and suggest how the students actions can be seen as evidence of their identity group. In this way student can bring up their own ideas and the mentor creatively attempts to make personal connections. Like DBR, there are guidelines as to how these connections should be made. Obviously, my work does not provide the great variety of different scaffolds that comprise the whole DBR curriculum – but insofar as it is a single small intervention it has a similar amount of support.
The reason I choose a high level of structure was based on what I knew of the literature of identity. I knew that cultural artifacts like stereotypes [3] strongly influenced identity and also that students natural identities frequently made them feel that computers were not for them [13]. Like any conceptual change that is at odds with an individual’s experience change is difficult [12]. Given that my intervention was small, providing a few pre-built identities was a smaller change. A key question of there research, is whether more identity information can really make a difference or if deeper conceptual change is required.

Identity Construction from a Radical Constructionist Perspective

A different direction would be to encourage students to construct their own identities. Work like this would likely be similar to Bers’s [2] identity construction environments which encouraged students to construct identities that were meaningful to them with fewer explicit ideas for what a desirable identity was. Knowing that identity construction can be a social process [3], I think a constructionist version of my would need to somehow allow students to look at other computer scientists and use them to come to their own ideas about identity.

In terms of disadvantages, two main issues stand out. The first is motivation – if the goal is construction of a computing identity, getting students to invest sufficient time would be challenging. This would especially be true of students who have little natural identification. Another issue would be that, when observing computer scientists, students would naturally feel distant from them. The psychology of group identification suggests that the most salient differences form the basis of group categorization – and there are many obvious differences between students and computer scientists. This is the sort of challenge that might be alleviated by a constructionist toolkit – scaffolds of some sort could help students see similarities between themselves and computer scientists. It clear that a framework can help even in the presence of natural inclinations – Resnick documents how his StarLogo toolkit encourage students to think of distributed explanations even thought they had a initial bias for centralized explanations [10].

There are considerable potential advantages. Student could construct identities that better match their own interests. Because the process more nearly matches the normal process of social identity formation, the resulting identities would likely be stronger. Finally, the students would likely come to a deeper more realistic understanding of computer science identity than they would with a few hastily constructed identities presented by myself and mentors.

References


