Kinesthetic Learning in the Classroom

Andrew Begel
CS Division, EECS
Univ. of California, Berkeley
abegel@cs.berkeley.edu

Daniel D. Garcia
CS Division, EECS
Univ. of California, Berkeley
ddgarcia@cs.berkeley.edu

Steven A. Wolfman
CS&E
Univ. of Washington
wolf@cs.washington.edu

Categories and Subject Descriptors

K.3.2 [**Computers and Education**]: Computer and Information Science Education—*computer science education*

Keywords

Kinesthetic learning style, sensorimotor learning, active learning

1. SUMMARY

We propose a special session focusing on kinesthetic learning activities, *i.e.*, physically engaging classroom exercises. These might, for example, involve throwing a frisbee around the classroom to represent transfer of control in a procedure call, or simulating polygon scan conversion with rope for edges and students for pixels. The session will begin with a brief kinesthetic learning activity to motivate the value of these activities. We will follow with a variety of examples, and discuss how to deploy these in a classroom. In the middle of the session, the audience will divide into facilitated groups to design their own activities. Finally, we will all mingle to share and discuss the results. We will set up a public web forum for continued discussion and generation of new ideas.

2. OBJECTIVE

Our objective is to focus the attention of the SIGCSE community on an underused and ill-documented instructional technique: the kinesthetic learning activity (KLA). We define a KLA as "any activity which physically engages students in the learning process." Generally these are short (20 minutes or less) classroom-based activities. They may involve just a small number of students (*e.g.*, sorting a few students in a section) or everyone (*e.g.*, asking the entire class to link up into a human binary tree).

We will focus on CS-related activities. However, KLAs' interactive nature makes them valuable not just for "content-related" exercises, but also to address social challenges facing *any* classroom (*e.g.*, starting a term with a group shout to establish a pattern of participation). We will not focus on any particular CS subject as KLAs are applicable across the breadth of computer science.

Several individual activities which we classify as KLAs have been discussed in the literature (e.g., [5, 6]). Some particular classes of KLAs, such as manipulatives [2], have also received attention. However, we believe that the great potential of these activities merits a thorough and ongoing discussion in the SIGCSE community.

These exercises fill an important niche in CS education — energizing students and employing learning styles rarely tapped by our instructional techniques. KLAs engage students by putting them in motion and sometimes even requiring real exertion, raising heart rates that tend to lag during lecture [1]. KLAs also tap into what Piaget termed "sensorimotor learning," in which physical participation in a learning experience transfers into mental symbols representing that experience [4]. KLAs can engage other important learning styles, such as Felder and Silverman's active, sensing, intuitive, visual, or global learners [3]. Finally, KLAs can be incredibly fun (and often low prep!) for instructors.

While KLAs have great value, they can be challenging to use in the classroom. It is easy to create a KLA that misfires because it is socially inappropriate, physically challenging, difficult to manage, or simply incomprehensible. Even apparently innocuous activities can conceal dangerous pitfalls. For example, sorting students by height or hair length can be socially intimidating; sorting by student ID or social security number may be an invasion of privacy; and sorting by age or matriculation year, even if it were socially acceptable, would cause undesirable clumping of the sort keys. Many KLAs have the potential to exclude students who are shy or have motor impairments. We will present a balanced picture of each activity's strengths and weaknesses.

3. HUMAN CONS CELL JEOPARDY

Human Cons Cell Jeopardy is a KLA that can be used in any introductory Scheme (or Lisp) programming course once students have begun forming `cons` cell data structures and drawing box-and-pointer diagrams.[1] In the exercise, students learn to translate between box-and-pointer diagrams and Scheme expressions and to recognize `cons` cell data structures in non-traditional representations. Both of these skills are important for designing, using, and modifying `cons` cell data structures.

The exercise requires a few minutes of reusable prep and no special materials besides pen and paper. This variant is ideally suited for a class of around 30 students but is easily adapted to larger or smaller classes. The amount of class time used is tunable between about fifteen minutes and a full class period.

For the exercise, students divide into groups of five. One group at a time comes to the front of the room and is given a box-and-pointer diagram on paper. The students must physically act out the

[1] A `cons` cell is the primary Scheme data structure, represented as an object with two slots labeled `car` and `cdr`.

To appear as special session, SIGCSE 2004.

diagram. Each student is a `cons` cell: her left hand is the `car` pointer and her right hand is the `cdr` pointer. A folded arm represents a null pointer. Pointing at oneself represents a self-reference. If a `car` or `cdr` points to a value, the student holds a large piece of paper inscribed with the value.

Once the students are ready, groups of students in the audience must work out the Scheme expression that would create the structure presented. When they have it, they "buzz" in and announce their answer, and a new group gets to act out a structure. The box-and-pointer diagrams start out simple to warm students up to the task, but proceed to complex diagrams (even self-referential structures which require naming intermediate `cons` cells).

Human Cons Cell Jeopardy shares some pitfalls with all other KLAs. Students may be uncomfortable being singled out or acting "silly." We believe that establishing a culture of participation early in the course can ameliorate these issues. The instructor must also remain aware of the need to intervene if it stalls or gets out of hand.

Several pitfalls are specific to this exercise. It requires students to be mobile and able to stand for brief periods. Some `cons` cell structures may require the students to contort their arms and bodies (*à la* Twister). Students are also assumed to have two usable arms. Instructors should consider the physical limitations of their students and practice proposed diagrams with friends before using them in class. The instructor should instruct students to limit any contact to light touches to the shoulder. (Instructors should verify that no discomfort arises from even this touch as modesty concerns abound!) Apart from these, the biggest pitfall is poorly-prepared `cons` cell diagrams: instructors must construct meaningful structures and remember to arrange them in order from easy to challenging.

4. LITTLE PEOPLE RECURSION

The Little People KLA is a short (˜10 minute) activity that is often used to teach recursion.[2] It has been used in a 250-person introductory C course, but could work with any language or class size. It is usually conducted on the first day recursion is introduced.

The instructor poses the problem of calculating factorials. First, she selects a student to stand and represent the "base case." That student will "know" that $1! = 1$. She picks another student and asks him to calculate $2!$. The student should be instructed to say something like, "I don't know what $2!$ is, but I know it's 2 times $1!$," and to ask the base case student for the value of $1!$. Then, the instructor picks students to calculate $3!$, $4!$, *etc.* At each step, the instructor has the new student stand and join the others. She should emphasize that this new student need only know one other student to calculate his value. Once the pattern is established, the instructor selects one student to be the *last* person selected and asks *everyone else* to stand and imagine their places in the calculation as if they had continued the pattern. She asks the seated student to calculate $250!$. He will say, perhaps with some help, "I don't know what $250!$ is, but...," and ask his neighbor for the value of $249!$. The point of this exercise is that, as with many recursive procedures, the n^{th} student/case needs only the result from the $(n-1)^{st}$ student/case and can assume that case works without explicitly working through the full recursion.

5. ORGANIZATION

The session will, of course, commence with a brief KLA. We will then discuss the value of KLAs based on pedagogical literature and personal experience. Next, we will present and reflect on two more KLAs, modeling our pattern for discussing these activities (as demonstrated in Section 3):

[2]Due to space restrictions, we cannot fully describe this KLA here.

1. Describe the context and pedagogical goals,
2. Enumerate necessary materials and cognitive prerequisites,
3. Present the mechanics of the activity, and
4. Highlight particular pitfalls.

We allocate 25 minutes to this process.

The middle of the session will be spent on facilitated group work designing new KLAs. We will supply some physical materials (props) and suggested directions to get the groups started. Each group will also have a large posterboard worksheet laid out according to the pattern described above to organize and record their discussion of KLAs. (These worksheets will become posters for the next part of the session.) We allocate 25 minutes to this process.

The session will conclude with groups sharing their KLAs in a poster-session format to maximize personal discussion of the activities. Each group will divide into two parts, **A** and **B**. **A** members will remain with their poster while **B** members move about the room, discussing activities that catch their interest with those groups' **A** members. After ten minutes, the roles will reverse for another ten. Clearly, not all attendees will discuss all activities, but we hope many fruitful conversations will begin during this time.

We allocate the remaining five minutes to wrap-up activities (*e.g.*, distributing a feedback survey) and time overruns.

6. EXPECTATIONS

The session as a whole will not focus on a particular level of CS education. KLAs can be used in any class from seminars to mega-lectures, from K-12 through graduate courses, and from CS1 through the upper-division. However, the examples we present will focus on introductory courses for the sake of generality.

We have already discussed the value of KLAs for the learning process. We hope that this special session will encourage participants to enrich their teaching by using them. In the process, we will also have the opportunity to model some high-quality exercises, discuss critical features of and best practices for KLAs, and give the community hands-on practice engaging in and designing such activities. As a longer-term objective, we would like to establish a community of interest in KLAs which would create a repository of these techniques for the use of the broader SIGCSE community.

Although we have ambitious goals, our expectations of what participants should learn are modest. We will have succeeded if each participant comes away with (1) a sense of the value of KLAs, (2) an initial feel for how and why they fit into a successful CS lecture, (3) the URL for the KLAs community web site (http://www.cs.washington.edu/research/edtech/KLA/), and (4) at least one debugged, high-value KLA lodged ineluctably in their minds. We intend to solicit feedback on the session with a quick survey after we conclude. We envision that rich and ongoing discussion will take place on the web site.

7. REFERENCES

[1] Bligh, D. *What's the use of lectures?* Jossey-Bass Publishers, San Francisco, CA, 2000.

[2] Bucci, P., Long, T., Weide, B., and Hollingsworth, J. Toys Are Us: Presenting Mathematical Concepts in CS1/CS2. In *Proc. of FIE 2000* (2000), IEEE Computer Society Press.

[3] Felder, R., and Silverman, L. Learning and Teaching Styles In Engineering Education. *Engr. Ed. 78*, 7 (1988), 674–681.

[4] Hergenhahn, B., and Olson, M. *An introduction to theories of learning*. Prentice Hall, Upper Saddle River, NJ, 1997.

[5] Pollard, and Forbes. Hands-on labs without computers. *SIGCSEB: SIGCSE Bulletin (ACM Special Interest Group on Computer Science Education) 35* (2003).

[6] Wolfman, S. Making lemonade: Exploring the bright side of large lecture courses. In *Proc. of SIGCSE 2002* (2002).