

Preliminary Experiences with a Tablet PC Based System to Support Active Learning in Computer Science Courses

Beth Simon

Math & Computer Science Dept.
University of San Diego
San Diego, CA 92110
bsimon@sandiego.edu

Ruth Anderson

Computer Science Dept.
University of Virginia
Charlottesville, VA 22904-4740
ruth@cs.virginia.edu

Crystal Hoyer, Jonathan Su

Dept. of Computer Science & Engr.
University of Washington
Seattle, WA 98195
{clhoyer, jonsu}@cs.washington.edu

ABSTRACT

There has been much research on the benefits of active and collaborative learning and in its use in computer science courses. As classroom technology becomes more prevalent it is natural to develop systems that support the use of these techniques. We have developed such a system as an extension to Classroom Presenter [2], a Tablet PC-based presentation system. In our system students (or groups of students) are equipped with tablet computers and at various points during the lecture, are asked to solve a problem or respond to a question. Students respond by writing their solution on the tablet and submitting it wirelessly to the instructor. The instructor can view all student responses, select one or more to display to the class, and annotate responses with ink as they are being displayed. Student responses can be saved for review after lecture by the instructor or made available electronically to the students. In this paper we describe the system and our initial experiences using the system in two classes (CS1 and Computer Ethics) at the University of San Diego in Fall 2003.

Categories and Subject Descriptors

K.3 [Computers and Education]: General

General Terms: Human Factors

Keywords: Educational technology, active learning, collaborative learning, classroom assessment, Presentation tools, Tablet PC.

1. INTRODUCTION

Learning requires that students do more than just listen: “They must read, write, discuss, or be engaged in solving problems. Most important, to be actively involved, students must engage in such higher-order tasks as analysis, synthesis, and evaluation.” [7] Computer science educators have introduced active learning techniques into their classrooms by having students work alone or in groups to solve problems (algorithm tracing, writing code, creating UML diagrams, drawing finite state machines) and answer questions (What parse tree is generated by this sequence of tokens? What does the hash table look like after inserting these values?) [14]. When students participate in these activities during lecture, individual solutions can be shared with the class by asking (or drafting) a student or students to write their answer on the board or overhead projector. Sharing student solutions with the

Permission to make digital or hard copies of all or part of this work for personal or classroom use is granted without fee provided that copies are not made or distributed for profit or commercial advantage and that copies bear this notice and the full citation on the first page. To copy otherwise, or republish, to post on servers or to redistribute to lists, requires prior specific permission and/or a fee.

ITICSE'04, June 28–30, 2004, Leeds, United Kingdom.
Copyright 2004 ACM 1-58113-836-9/04/0006...\$5.00.

class has a number of benefits. Students are re-assured to see other students making mistakes similar to their own and are enlightened by seeing more advanced students present a particularly elegant solution. Asking several students to share their answer exposes students to a diversity of viewpoints and provides a meaningful opportunity to evaluate and compare different approaches to the same problem. The instructor also benefits from seeing several student responses presented as it provides a more complete assessment of her students’ understanding of a topic.

In this paper we describe a Tablet PC-based system that facilitates active and collaborative learning activities in the computer science classroom. In our system students and instructors are equipped with tablet computers and at various points in the lecture, students are asked to solve problems or answer questions by writing on their tablet. Students then submit their responses to the instructor, who can selectively display them to the class.

2. RELATED WORK

A large body of research exists on the benefits of active learning and we refer the interested reader to [7] for additional references. Many computer science instructors have attempted to introduce these techniques into their own classrooms [6,14].

The use of pen-based computers for note taking has been explored by several groups [1,10,12,13]. Although note taking is possible in our system, the focus of our work is on facilitating classroom activities and interaction with the instructor. The ActiveClass [15] project uses PDAs to promote interaction by allowing students to submit text-based questions to the instructor or to respond to polls posed by the instructor. Classroom Presenter also provides a mechanism for students to provide lightweight unobtrusive feedback to the instructor [3]. Several commercial systems exist that allow students to respond to multiple choice polls using remote controllers [11].

Lecturer’s Assistant [8] is an early system based on graphics tablets where students were able to remotely add ink to the instructor’s display to aid in asking questions. The DEBBIE system [6] has goals similar to our own. In DEBBIE, students sit at PCs equipped with video tablets and the instructor lectures from an electronic whiteboard or video tablet. Our system differs in that both student and instructor devices are wirelessly connected Tablet PCs that can be set up in any regular classroom.

3. SYSTEM DESCRIPTION

This work is part of the Classroom Presenter system [2] developed by the Educational Technology group at the University of Washington. Classroom Presenter¹, hereafter Presenter, is a Tablet

¹ The software is available free for educational and research use from <http://www.cs.washington.edu/education/dl/presenter/>.

PC-based electronic lecturing system allowing instructors to combine spontaneous inking with static prepared lecture notes. Display to the class is through a connection to a data projector.

We have implemented an extension to Presenter called *Student Submissions* which enables students equipped with Tablet PCs to contribute to and participate in the lecture at points determined by the instructor. This interaction takes the form of students inking over a copy of the current lecture slide displayed on their individual Tablet PC. When finished, the student wirelessly submits their ink to the instructor's Tablet PC. These submissions are captured as separate slides on the instructor's machine and can be previewed and selectively displayed to the class. Student responses are tagged with student names (visible only to the instructor) and collected (in the order they are received) in a slide deck separate from the original presentation. Preview of student responses is provided by use of a filmstrip as shown in Figure 1. By hovering the tablet pen over a slide in the filmstrip instructors can preview an enlarged version of a submission before choosing to share it with the entire class. The slide can be displayed to the rest of the class by tapping on it. Student responses are displayed to the class without student names and can be inked on by the instructor to draw attention to key points. Several ink colors are available for use by both students and instructors. At the end of lecture, both student responses and the instructor's original presentation with ink can be saved for use in the next lecture or for posting to the course web site for further discussion.

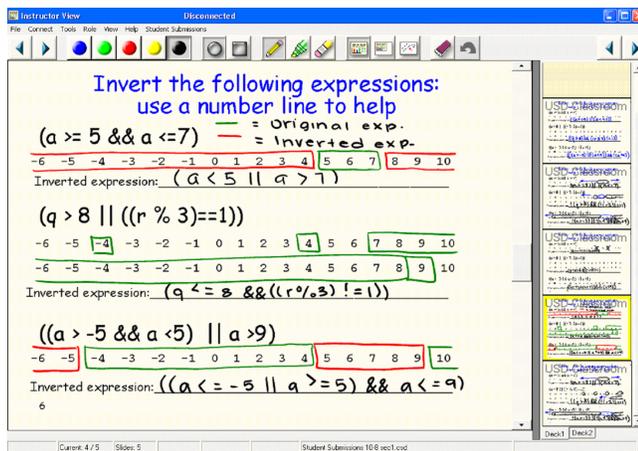


Figure 1. Instructor view of student work inking on top of a prepared slide. Here the student used color and created a legend to explain the color use.

4. SYSTEM FEATURES

Our goal in developing this system was to facilitate and improve upon active learning activities that instructors already attempt in the classroom using student notebooks, blackboards, and handouts. Our system provides several improvements over traditional methods for incorporating problem solving and active learning experiences into lecture and for sharing student solutions with the class. Here we outline several of our design goals and system features.

Support for complex problems and rich responses:

Presenter allows slides to be prepared from PowerPoint or directly from image files. The ability to use detailed color images as the starting point for an in-class activity greatly expands choices for activities. Multiple colors of ink and the high quality ink available

on the Tablet PC provide students with rich tools to use in crafting a response.

Support for spontaneous activities: Although prepared slides can be used as starting point for an activity, our system does not require that problems be devised ahead of time. Any slide can be used as a basis for a question or a question can be composed on the fly, perhaps in response to input from the class, on Presenter's blank "whiteboard".

Instructors can select answers or students: In a traditional classroom, when instructors ask a student to share their answer with the class they must make their selection based on the student rather than the *answer that the student will provide*. Asking for volunteers often results in the same eager students raising their hands and less confident students remaining silent, while drafting a shyer student to share their answer when they are ill-prepared leads to embarrassment. With our system the instructor has the ability to view all responses and select a response to share with the class based on the answer *or* the student. The instructor may choose to show a common error, a correct answer, or a particularly creative solution. Eager students can be assured that the instructor knows they can solve the problem, because perhaps they submitted a response before anyone else. Less confident students can have their confidence boosted by having the instructor select their solution to share with the class.²

Wireless submission of student responses: Our system can be used in any classroom where a wireless system is deployed. In our initial deployments the system was used in regular classrooms with freestanding desks. The instructor distributed the Tablet PCs to students and collected them after lecture. In the future one could imagine students being required or choosing to bring their own devices to class.

Use of saved responses after class: The ability to easily save all student responses supports several novel uses of the system. Student responses can be saved for discussion in the next class if class time runs short or if a particular response serves as a springboard for the next lecture topic. Saved responses could be used to substantiate classroom participation grades. Several responses could be posted to the course web site after class and students could be asked to comment or compare them. Additionally, the system could be used to administer assessments such as those described in [4].

Students work simultaneously and privately: Since each student or student group has control over their own workspace, all students can be working on the same (or different) problems simultaneously without being influenced by the work of others.

Students control their workspace: In our system students are in control of when (and whether) they submit their solutions to the instructor. Students can even submit multiple times, if they want to "update" their solution. This level of control allows students to feel at ease to make mistakes or consider several solutions while solving problems just as they would in their own notebook.

² [15] points out that instructors face a similar dilemma when students raise their hand to ask a question. ActiveClass addresses this problem by providing instructors with the set of questions students have asked, allowing the instructor to make a selection based on the fitness of question rather than the student.

Instructor controls when responses can be submitted:

The instructor can turn on or off the students' ability to submit responses to his Tablet PC. Although leaving the ability on at all times could allow an unobtrusive feedback channel for students to use to submit comments or questions, the instructor may wish to disallow this at times for example to prevent his machine being flooded with responses.

5. PRELIMINARY DEPLOYMENT

Presenter augmented with Student Submissions was deployed in Fall 2003 in two classes at the University of San Diego: one computer ethics class (18 students), and two sections of CS1 (14 students and 7 students). Student submissions have been used in a total of 8 scenarios (15 different lectures). Five Tablet PCs were used in each class to enable student submissions. Students used the tablets individually and in teams of 2-4. Here we highlight some of the activities and learning opportunities afforded by the system.

5.1 Exposure to different coding solutions

In the CS1 course, our system was often used to give students an opportunity to practice writing code during class. Reviewing several student solutions during class provided an opportunity to point out common mistakes and allowed comparison of different approaches to the same problem. The instructor also received immediate feedback on whether or not students understood the programming structure just discussed.

Common Errors: In Figure 2 we see one student's submission of a loop that was supposed to output the first 10 multiples of 8. Discussion of the loop by the class brought out several common errors including:

- Assignment of the iterator in the loop header uses == rather than =.
- The loop control expression states when the loop should stop running rather than when it should keep running.
- The value for value should not be updated each time through the loop – a new variable should be used.

Examining actual student responses is a good way of uncovering common difficulties that the instructor may not anticipate. Discussing these errors in class can help prevent students from making the same errors in the future and can boost student confidence by allowing them to see other students making errors similar to their own.

Different Approaches: In Figure 3 we see two student submissions that were supposed to print out all the numbers between 14 and 77 that are multiples of both 7 and 2. The first example allowed the class to explore the difference between a condition controlling a loop and the conditional execution of part of a loop body. (The code also includes an error in this respect.) Comparing the two submissions showed students how for and while loops can be used to solve the same problem and brought up the question of the efficiency of updating the loop control variable by 1 or by 7.

5.2 Augmenting existing diagrams

One feature of our system is that students can easily be provided a prepared diagram or background to ink over, fill in, or modify. Figure 1 shows a prepared slide from the CS1 course that a student has inked over with red and green ink. After a discussion of DeMorgan's Law, students were asked to visualize those values which will cause an expression to evaluate to true, then to find the

"inverse" of the expression. Students, without guidance, utilized two different colors to mark up the number line to show the "original" and "inverted" expressions.

```

int value = 8;
int iteration;
for (iteration == 1; iteration == 10; iteration++)
{
    value * = iteration;
    s.outln (value);
}

```

Iteration	value
1	8
2	16
3	24
4	32
5	40
6	48
7	56
8	64
9	72
10	80

Figure 2. Sample student submission that prompted a discussion of common errors with for loops.

```

int count;
for (count == 14; (count % 2 == 0) &&
    (count <= 77); count += 7)
{
    System.out.println (count);
}

```

iteration	count	prints out?
1	14	yes
2	21	no
3	28	yes
...
9	70	yes
10	77	no

```

int num = 14;
while (num <= 77)
{
    if (num % 7 == 0 && num % 2 == 0)
        s.o.p ("num")
    num = num + 7;
}

```

Figure 3. Two student submissions for the same problem. These examples showed how a for loop and a while loop could both be used to solve the same problem.

5.3 Spontaneous activities

Presenter also supports completely unplanned classroom activities. Every Presenter “lecture” automatically includes a scrollable “whiteboard” slide – an empty slide where the instructor can make additional remarks or follow a digression. In a CS1 lecture on nested loops, the instructor jumped to a whiteboard to show the output from a given set of nested loops. The instructor wrote the numbers (but not the circle or square) shown in Figure 4. Presenter automatically transfers this ink to all “viewer” tablets. The instructor then asked students to draw a circle around the output of the 3rd iteration of the inner loop within the output of the 5th iteration of the outer loop. Additionally students were asked to draw a box around all the output from the 3rd iteration of the outer loop. Figure 4 shows a sample student submission.

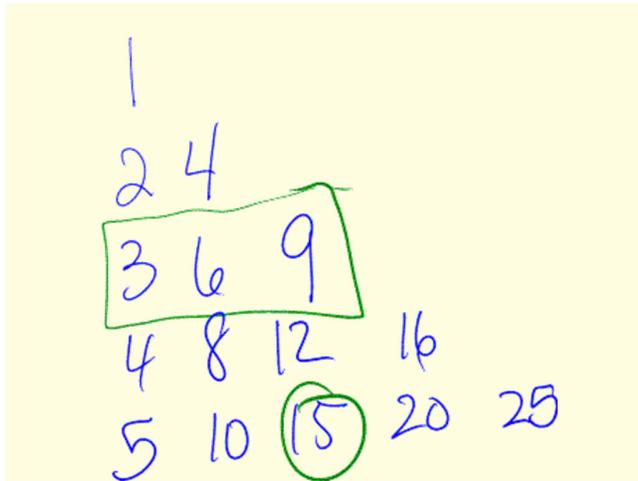


Figure 4. Example of a spontaneous in-class activity. The instructor wrote the numbers shown above as the output of a nested loop. Students then overwrote the instructor’s ink with a circle and a square identifying certain loop iterations.

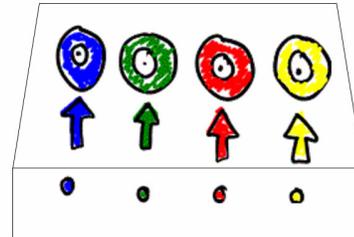
5.4 Surveying the class’s ideas

Our system provides a very efficient way to gather a large number of ideas from the class in parallel. This ability was exploited in the CS1 and ethics courses in two different ways.

One use is to assign the entire class the same problem – where the goal is to critically analyze a wide variety of responses, culling the positive features from each submission. An example of this is seen in Figure 5 from a computer ethics activity on human computer interface and design issues. Here, students were asked to design a stove to be used safely by members of the non-literate Oompa-Loompa race [5,9]. A background diagram was provided (as a slide in the instructors’ original presentation) and each student group spent time drawing features of their design. Student use of color and the ability of each team to develop a design without being able to see (as they would be at a traditional blackboard) and potentially be influenced by another team’s design were critical to getting the most from this exercise.

Another way to capitalize on the ability of students to work in parallel is to assign each group a *different* problem on which to work. This allows students to spend a relatively shorter amount of time in class on an activity, but to be exposed to a variety of problems and solutions in a time efficient manner.

How would you design a stove for the Oompa Loompas?



How would you design a stove for the Oompa Loompas?

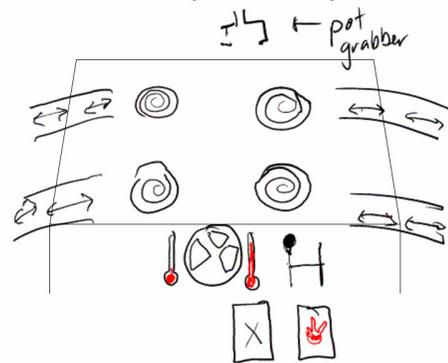


Figure 5. Two examples of student work exploring design.

In the CS1 class, after a preliminary discussion on class design, student teams were each charged with brainstorming potential instance variables and class methods that might be useful in a certain class. Each group worked on a different class (a library patron class, a gas station class, a coffee shop class, a card hand class) but was able to be exposed to other classmates’ design ideas quickly as the different teams’ submissions were reviewed.

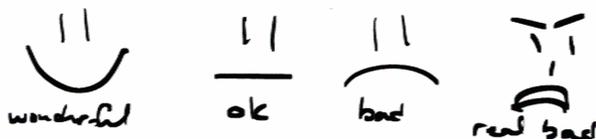
In the computer ethics class, students were each assigned an “area” of the Oompa-Loompa factory for which they were to design “icons” (packaging, shipping, chocolate quality control, etc.). Figure 6 shows some sample submissions whose merits were discussed.

5.5 Student reactions

In general, student reaction to the electronic submission of work has been positive. Students clamor to submit their work and make wistful comments if time precludes all responses being displayed to the class. We have observed that occasionally, slower students or those having difficulty with the topic will refrain from submitting work. Since each tablet submission is stamped with a login name, the instructor can choose to approach those students after class to offer help.



a) Student-designed icons for packaging.



b) Student-designed icons for chocolate quality control.

Figure 6. An example where each team simultaneously worked on a different set of icon designs.

While the sample size of student users is too small to allow for quantitative analysis, anecdotal evidence shows that students feel they benefit from use of the system. Specifically, students have commented on the ability of the system to combat student shyness by allowing “anonymous” submission of ideas. One senior computer ethics student also felt that the system afforded him benefits usually only available to instructors: “Normally the teacher is the only one that has a broad view of the diversity of the learning and thinking approaches exhibited by students. With the use of the tablet pc, this exposes and provides all students with a broad view of the diversity of learning and problem solving approaches.” A CS1 student similarly stated, “[U]sing the tablets everyone can do something and see everyone else’s way of thinking and approaching the topic being covered in class.” Another student noted the efficiency of the system saying “I think they [the tablets] saved some time, in comparison to going up to the board, trying to share markers, etc.”

6. FUTURE WORK

We have only seen a small number of deployments and there are many questions left to explore. One interesting issue to explore is whether or not student responses differ qualitatively or quantitatively when the system is used compared to solving the same problems in their notebooks or at the board. Another potential use for the system is for students to use it to submit questions to the instructor in a manner that is anonymous to the class, but non-anonymous to the instructor. The system could also be used to facilitate participation by students at a remote site when used for distance learning. Finally, it would be interesting to see how the system would be used in disciplines outside computer science as well as in a greater range of computer science courses.

7. CONCLUSIONS

The benefits of active learning in general, and in computer science classes in particular, has been widely documented. As developers of technology ourselves, computer science instructors and students are uniquely positioned to be early adopters of technology to support active learning in the classroom. We have developed such a system and have presented early experiences with its use. Although our initial deployments were small, we found that our system supports active and collaborative learning activities in the computer science classroom. With our system, students were easily exposed to common mistakes and a wide variety of solutions to the same or different problems, took advantage of a rich environment for expressing their ideas, and benefited from spontaneous activities.

8. ACKNOWLEDGEMENTS

The authors would like to thank the members of the Classroom Presenter development team, particularly Richard Anderson and Steven Wolfman.

9. REFERENCES

- [1] Abowd, G., Classroom 2000: An experiment with the instrumentation of a living educational environment. *IBM Systems Journal*, Volume 38, Number 4, 1999, pp. 508-530.
- [2] Anderson, R., Anderson R., Simon, B., Wolfman, S., VanDeGrift, T., & Yasuhara, K. Experiences with a Tablet PC Based Lecture Presentation System in Computer Science Courses. In *Proc. of SIGCSE '04*, pp. 56-60.
- [3] Anderson, R., Anderson, R., VanDeGrift, T., Wolfman, S., & Yasuhara, K., Promoting Interaction in Large Classes with Computer-Mediated Feedback. In *CSCL 2003*, pp. 119-123.
- [4] Angelo, T. A., & Cross, K. P. (1993). *Classroom Assessment Techniques: A Handbook for College Teachers*. San Francisco: Jossey-Bass Publishers.
- [5] Bell, T., Witten, I., & Fellows, M., “Computer Science Unplugged”. June 8, 1998. <http://unplugged.canterbury.ac.nz/>
- [6] Berque, D., Johnson, D., & Jovanovic, L. Teaching Theory of Computation Using Pen-Based Computers and an Electronic Whiteboard. In *Proc. of ITiCSE '01*, pp. 169-172.
- [7] Bonwell, C. & Eison, J. Active Learning: *Creating Excitement in the Classroom*. ASHE-ERIC Higher Education Report No. 1. Washington, D.C.: 1991.
- [8] Buckalew, C. & Porter, A. The Lecturer’s Assistant. In *Proc. of SIGCSE '94*, pp. 193-194.
- [9] Dahl, R., *Charlie and the Chocolate Factory*. 1964.
- [10] Davis, R., Landay, J. et al. NotePals: Lightweight Note Sharing by the Group, for the Group. In *CHI '99*, pp. 338-345.
- [11] eInstruction <http://www.einstruction.com/>
- [12] Golub, E., <http://www.cs.umd.edu/~egolub/AVIAN/BIRD>
- [13] Iles, A., Glaser, D., Kam, M., & Canny, J. Learning via Distributed Dialogue: Livenotes and Handheld Wireless Technology. In *CSCL 2002*, pp. 408-417.
- [14] McConnell, J. Active Learning and its use in computer science. In *Proc. of ITiCSE '96*, pp. 52-54.
- [15] Ratto, M., Shapiro, R. B., Truong, T. M., & Griswold, W. G. The ActiveClass project: Experiments in encouraging classroom participation. In *CSCL 2003*.