COMPUTER SCIENCE OUTREACH IN AN ELEMENTARY SCHOOL*

Lynn Lambert
Christopher Newport University
Newport News, Va 23606
(757) 594-7826
lambert@pcs.cnu.edu

Heather Guiffre
Virginia Commonwealth University
Richmond, Va 23284
(804) 827-1467
hnguiffre@vcu.edu

ABSTRACT

To encourage interest in computer science, we visited three fourth grade classes once a week for thirty minutes each (three content sessions and a pre-test and post-test session) in March-April 2007. The basis for the activities was Computer Science Unplugged. Pre and post tests indicate improved confidence and interest in Computer Science and Math.

1. INTRODUCTION

In order to counter current trends in computer science, Christopher Newport University has initiated outreach programs, with Girl Scouts, PTA programs, science nights, science clubs, etc. We also have visited schools during the school day. In Spring 2007, we visited three fourth grade classes at Riverside Elementary School in Newport News, Virginia once a week for thirty minutes for five sessions to do computer science activities, many of them based on Computer Science Unplugged (www.csunplugged.org). We held three content sessions (binary numbers; fractions and division; and parity and the Internet) a pre-test session and a post-test session. Results indicate improved interest and confidence.

In the rest of this paper, we argue why we need to begin computer science outreach in elementary school; what Computer Science Unplugged is; who went and what we did; how we evaluated the project and what our results are; problems that we encountered, and how we might address those; and our conclusions.

2. THE NEED FOR EARLY COMPUTER SCIENCE OUTREACH

Both industry and government indicate that the number of jobs in areas related to computer science is increasing and will continue to increase**.** 7 of the 19 fastest growing jobs in *Fortune* magazine's list of the *"Fastest-Growing Professional Jobs"* are in software and hardware. [For05]. Furthermore, salaries for these jobs are projected to remain high.

In the meantime, the number of people who are qualified for these jobs is shrinking. Computing Research Association reports a 50-70% drop in majors from 2000 to 2005 [Fos05,Veg06]. Last year in the ACM programming competition, only one school from the United States finished in the top 10 (MIT, finishing 4th) [Bay07]. Additionally, interest in computer science in the high school age group is waning. Computer Science is the only AP exam that is experiencing a decline in the number of people taking it [CSTA07] (though 2008 had a hopeful increase). Recently, ETS has announced that it will discontinue the AB [Vi08] exam (and will have the A only).

There are programs to encourage involvement at all levels, but often students become disinterested before these programs begin. Early exposure to computer science is especially important for young girls. Girls are as interested and involved with computers as boys in elementary school [Joh05]. However, that interest decreases as girls get older. The time to reach future computer professionals is in elementary school.

3. OUR PROGRAM

We chose to target fourth graders based on previous successes with that age group. They understand the idea of sequencing, can read and understand addition and subtraction. They also are still interested in trying new ideas, and are not yet jaded. Four Christopher Newport University (CNU) departmental majors helped lead each session. A psychology student developed and administered a cognitive competency and interest test in computer science and math. Initially, we had planned six content sessions, but that was reduced to two computer science and one math session by the principal who was worried about covering state standards content. We developed the content based on activities that the lead was familiar with from previous outreach and interaction with the Computer Science Unplugged group. In the remainder of this section, we talk about the content of the sessions beginning with an overview of Computer Science Unplugged.

3.1. Computer Science Unplugged

Computer Science Unplugged, a book by Tim Bell, Ian Witten and Mike Fellows [BWF98] describes of a set of kinesthetic, fun activities that cover many core areas of computer science including searching, networks and finite state automata. Recommended in ACM's *A Model Curriculum for K-12 Computer Science* [ACM03], the Unplugged book describes error correction via the "magic trick" of parity bits; demonstrates pixels and image representation using spray paint; shows how binary number work, and examines user interfaces. The activities are intended for young audiences, but are appropriate for ages 5-adult. The book and activities have been used in the United States, Asia and Europe (and has been translated into Korean, Chinese, and Swedish). The book formed the basis for many of the activities.

3.2. The Riverside Project

We visited three fourth grade classes (each with 20-25 students) in succession performing the same activity for each class. This section discusses each activity.

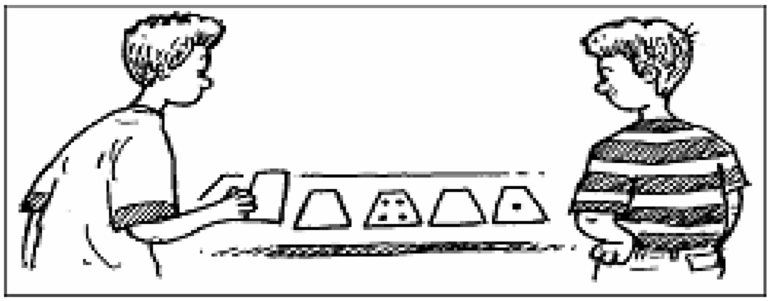

Figure 1: The binary number 5 using CS Unplugged Cards (Figures 1, 2, 4 and 5 are from the original Computer Science Unplugged book [BWF98] published in 1998 and available at www.lulu.com)

3.2.1. Activity 1: Representing Numbers, Text, Sounds and Images

Students were told that computers use the binary system to represent all information, and that 0s and 1s can be used to represent the binary system. To explain how numbers are represented using 0s and 1s, children are given 4 cards with 1 dot, 2 dots, 4 dots, and 8 dots. They then make the numbers 5 and 9 by turning some cards over and leaving some cards face down (see Figure 1), and explore using the cards to represent numbers. To describe how text is represented, children were given an additional 16 dot card, and a secret code with A=1 (00001), B=2 (00010), etc., up to Z=26 (11010). Children translated a message from binary into letters and words, then created a binary message of their own. Children enjoyed the idea of having a secret code for messages to their friends.

The next step in the binary activity was translating the 0s and 1s into sounds. The professor "sang" the message they had just translated (high note for 1, low note for 0). This was intended to be, and was, funny to the students, and they sang the messages later also. The final activity was making the transition from binary numbers to images. Figure 2 was given to show how black and white images can be represented where each black pixel is a 0 and each white pixel is a 1. So, with 0s and 1s, the first row in Figure 2 would be:

0	0	0	0	0	0	1	1	0	0	1	1	0	0	0	0	0	0

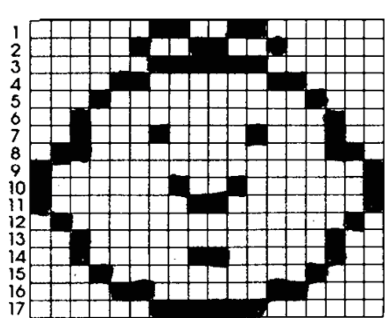

Figure 2: Translating Binary Numbers into Images

Children then filled in 0s and 1s for the next few rows. Of all of the activities, this was the children's least favorite. In other outreach, we have extended this to show that color can be represented when more than 1 bit/pixel is used. For example, Figure 3 shows how to use 24 bits to represent a blue pixel in RGB colors in Microsoft Word (0 0 255, or 00000000 00000000 11111111 in binary). It is easy to get all red (255 0 0, and all yellow, 255 255 0, if there is time for color adding explanations). With these concepts, children understand the basics of how computers represent numbers, text, sound and images.

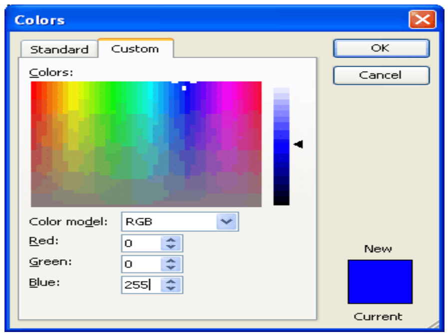

Figure 3: Microsoft Word Custom Color Dialog Box Showing RGB values for Blue

3.2.2. Second Activity

For the second activity, we went over two math concepts that the fourth graders were struggling with: fractions and long division. We explored fractions and equivalent fractions in different ways, including asking a fraction of students in a row to stand. Students shaded a paper roll that was on a) table in front of them. When 1/5 of the students in a row stood, the shading was 1/5 of the way across the table; when 3/5 were standing, the shading was 3/5 of the way down the table. When additional rows were added behind the first, students could see that 3/5 and 9/15 were the same in terms of where the shading. They then compared fractions using this same technique to see that 8/10 of the students standing was greater than 3/5.

For long division, children first calculated simple division. For 12 divided by 3, students distributed 12 beads into three different cups (take 3 at a time, and put one in each different cup). Next, the students calculated 14 divided by 3 to find remainders (if there are not 3 beads to grab, the rest is a remainder). Then we did this activity with students. Take 12 students, 3 at a time, and put them at 3 tables (then 14 students at 3 tables). This was extended to long division using a "divide/multiply/subtract" series of steps. The students loved both of these activities, and understood every part of them.

3.2.3. Third Activity

The last activity covered sending data over the Internet in two subtopics. First, we performed "the magic parity trick" from the Computer Science Unplugged book. In this activity, students are given 25 squares, each with one white side and one black side. They create a 5x5 grid of white or black squares (they choose without the computer science teacher/magician looking). An apprentice (one of the undergraduates) then adds a parity row and column to make each row and each column have an even number of whites (see Figures 4A and 4B).

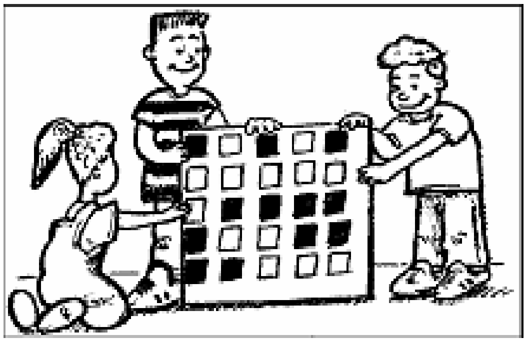

Figure 4A: 5x5 grid

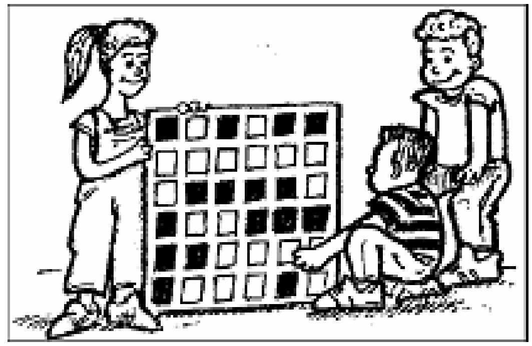

Figure 4B: Column/Row Parity Added

Finally, a student is picked to reverse one card. At this point, the computer science professor looks at the squares, finds the row and column that have an odd number rather than an even number of whites, and chooses the one square in that row and column as the square that has been shifted (see Figure 5):

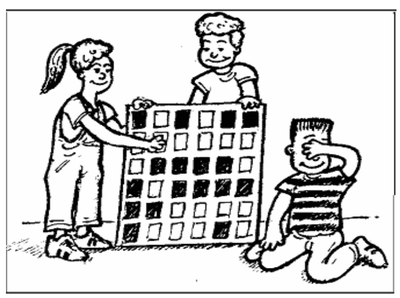

Figure 5: Fourth row, Third column Square is changed

Students love this, ask how it works so they can show their parents. We then explain that computers use parity, that bits can get flipped in transmission, and that computers can correct small errors like the one in the grid.

The second part of the third activity day is a simulation of how Transmission Control Protocol (TCP) works on the Internet. Each table where the students are sitting is labeled. Each table writes a message to another table. The students at the sending table then tear the message into 4 pieces and label each piece with Table Origin Number, Table Destination Number, Order that this part of the message is, and how many total parts there are. The students then pass the message parts to an adjacent table (not all parts go to the same table). That table looks at the parts of messages as they come in, and passes them closer to the destination, or keeps the message if it is for that table. When all tables get their message, they put the message back together and read it. Students love this, but it can get noisy as the packets get passed around. This activity is not from the Computer Science Unplugged book.

4. RESULTS

Students were given a pre-test one week before the first activity, and an identical post-test one week after the last activity. We matched each student's pre and post-test and measured the difference. The questions measured how confident students were in general regarding their cognitive skills, their interest in computer science, their interest in mathematics, and their math anxiety. The scale consisted of 15 simple questions answered on a 5 point Likert scale (from Strongly Agree to Strongly Disagree). Sample questions include: "I would do well in a computer science class" and "I understand math less quickly than other people that I know". Questions were developed to fit into one of four different constructs: computer science interest; cognitive competence (confidence in all academic areas); math anxiety (or lack of confidence); math interest. Results indicate that students were more interested in computer science, had significantly higher cognitive competence, and were significantly more confident about math ($p<0.05$ for all), but not significantly more interested in math.

5. PROBLEMS

Although the students received the interventions enthusiastically, there were some problems. In future outreach, more emphasis will be given to training the undergraduates to make a sustainable program that the undergraduates can conduct more independently. Public schools are, in general, not enthusiastic about computer science interventions since there are few or no computer science state standards tests at any grade level (those that do exist are usually technology, not computer-science based). Worry about state standards testing reduced this intervention from a planned 6 computer science sessions to 2 (and a math session). Future effort will include finding more partners, and working with them to help with covering curriculum in state standards. We will partner with a private school in Fall 2008 which does not have the same restrictions on the curriculum. In terms of the evaluation, we will evaluate the undergraduate experience next time. Finally, determining if the improvements in confidence and interest are long lasting would make this intervention more convincing. It is likely that occasional subsequent intervention would be necessary to maintain interest and awareness in the field.

6. CONCLUSION

The elementary school students thoroughly enjoyed the activities, and seemed to understand each one. The CNU students also benefited, both in learning content and in helping the children. This outreach has been expanded into a 1 credit elective computer science course that was offered Fall 2007 and will be again Fall 2008. In general, the authors discovered that the children had (before the intervention) surprisingly limited knowledge or interest in computer science, especially given their intense interest in technology. For example, although all students had used the Internet, all of them thought a copy of the Internet existed in each separate machine. Thus, the visits made us feel even more strongly that early intervention is both needed and effective.

7. REFERENCES

[ACM03] "*A Model Curriculum for K-12 Computer Science: Final report of the ACM K-12 Task Force Curriculum Committee,*" Allen Tucker, Chair, Computer Science Teachers Association, October 2003.

[BWF98] *Computer Science Unplugged* by Tim Bell, Ian Witten and Michael Fellows, 1998. available at lulu.com. Later editions available at www.csunplugged.org.

[CSTA07] "AP Report Shows Slight Improvements" ,http://blog.acm.org/archives/csta/2007/03/ap_report_shows.html, Computer Science Teachers Association web site, March, 2007.

[For05] "Fortune 75: 20 Fastest-Growing Professional Jobs," *Fortune Magazine*, March 21, 2005.

[Fos05] "Student Interest in Computer Science Plummets", *Chronicle of Higher Education*, May 27, 2005 by Andrea Foster, Vol 51, Issue 38, page A31.

[Joh05] "Handheld Computers: No Child Left Behind's (NCLB's) Digital Divide Equalizers?" by Doris G. Johnson, "Journal of the Research Center for Educational Technology," Fall 2005.

[Veg06] "Drop in CS Bachelor's Degree Production," by Jay Vegso. *Computing Research News*, March 2006.

[Vi08] "AP Language, Computer Courses Cut", By Daniel de Vise, *Washington Post*, Friday, April 4, 2008; Page A04, http://www.washingtonpost.com/wp-dyn/content/article/2008/04/03/AR2008040303925.html".