Rex has been dognapped!
How do you befriend a meal?

USING STORYTELLING TO MOTIVATE PROGRAMMING

Using the Storytelling Alice programming environment to create computer-animated movies inspires middle school girls' interest in learning to program computers.

By CAITLIN KELLEHER and RANDY PAUSCH

Student interest in computer science is falling in the U.S., as well as in other countries worldwide. The Higher Education Research Institute found that between 2000 to 2005 the number of college freshman listing computer science as their probable major dropped by 70% in the U.S., and the Taulbee Survey found that computer science enrollment at research universities dropped by 50% [10]. Meanwhile, we have found that presenting computer programming as a means to the end of storytelling motivates middle school girls (ages 11 to 15) to learn to program, potentially increasing the number and diversity of students studying computer science.

There is also still a strong and growing need for computer scientists in many countries. The U.S. Bureau of Labor Statistics predicted in 2005 that 65% of job

Student programs created with Storytelling Alice. (Carnegie Mellon University, Pittsburgh, PA.)

openings from 2004 to 2014 in science and engineering will be in information technology [3]. But the computer science community must do better than simply fill available jobs. Computer science promotes progress across a range of disciplines, including all the basic sciences, medicine, business, and communications. With such broad influence, it is critical for the U.S., as well as for any given national economy, that computer science attract the brightest minds. Other disciplines, including biology and chemistry, have greatly increased their talent pool in recent years by drawing from both genders. The 2005 Taulbee survey found that 84.9% of bachelor's degrees in the U.S. in computer science were awarded to men [12]. In addition to increasing the size of the talent pool, creating a more diverse computer science community will help ensure that new computing-based technologies meet the diverse needs of the global population.

Figure 1. Scene created in Storytelling Alice.

Although several factors likely contribute to the loss of student interest in computer science, one particularly important one is that students often find their first course uninspiring. Typical assignments, like "sort a list of numbers" or "generate the sum of the first 1,700 integers," fail to engage many students. There is now a growing interest among computer science departments and industry and government funding sources in using computer and video games to draw undergraduates into the field and in creating video games majors. The motivation behind these programs is a belief that today's students are likely to be much more engaged by computer games than they are by a more traditional introduction to computer science. In 2000, fewer than a dozen game-related degree programs were available in North America [8]; by 2005, there were more than 100 [8].

There is also a potential trade-off. Using games to motivate students to study computer science may help increase the number of computer science majors but further decrease the percentage of women in the field. One widely quoted 2005 survey found that the gaming population in the U.S. is 43% female [2]. This is in fact a misleading statistic because it includes casual game play (such as games like solitaire). It is likely that gaming-based majors will be more appealing to hardcore gamers than to the occasional player of solitaire and bejeweled. Who are the serious gamers? In 2006, the publisher of *Electronic Gaming Monthly* reported its readership was 92% male [11]. Netshelter, a provider of marketing information, found through a Web site survey that the community of hardcore gamers—those devoting significant capital to games-related purchases—was 97.5% male [7].

3D Movies

It is critical for preventing the further erosion of interest in the field that computer science professionals and educators find a way to motivate more female students to pursue computer science. Research into girls' educational choices has shown that to have the greatest potential effect on increasing the number of women who choose to study computer science, it is important to intercede no later than middle school, when girls typically decide whether or not to pursue math- and science-related studies [1]. Today, relatively few middle school girls learn to program computers.

If you walk into a classroom of middle school girls in the U.S. and ask how many of them want to learn to program, few hands are likely to go up. If you ask how many of them want to learn to make animated movies like those from Pixar and Dreamworks, you are likely to get a very different and very positive response. Creating an animated movie and learning to program a computer can be fundamentally the same activity.

Learning to program is recognized as a difficult endeavor. There is a long, rich history of attempts to ease the process of learning to program and make programming accessible to a broader population of people [5]. Storytelling Alice is based on an existing programming environment—Alice 2.0, www.alice.org—that allows novice programmers to create interactive 3D virtual worlds. In Alice 2.0, users construct programs by dragging and dropping code elements, thus removing the possibility of making syntax errors. Programs in Alice are animations

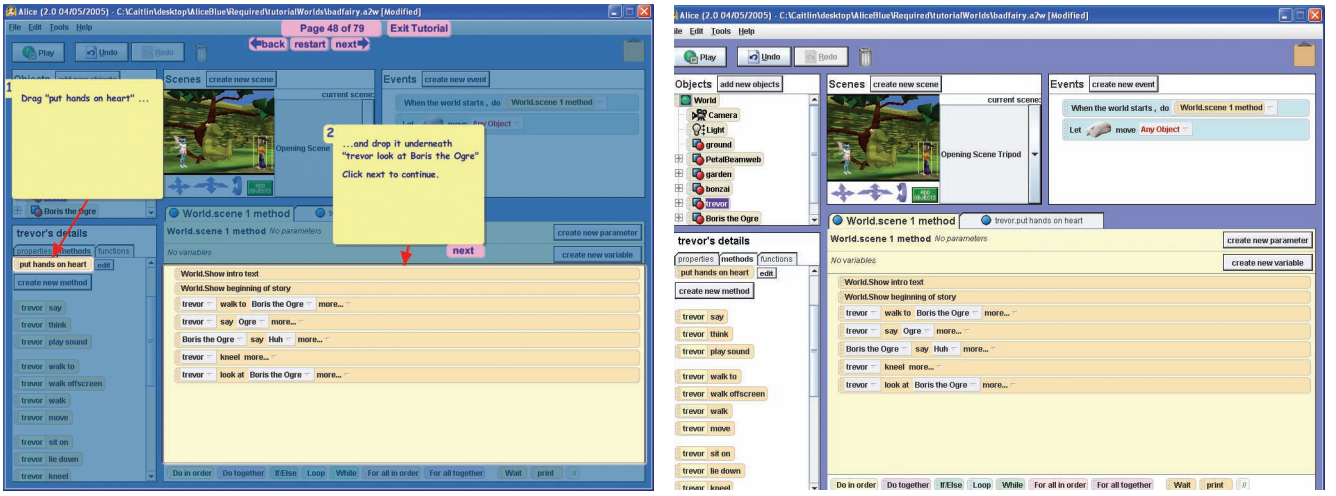

Figure 2. Views of Alice interface with and without a Stencils-based tutorial.

that allow users to see their mistakes as they occur. Alice allows students to gain experience with a number of programming concepts, including looping, conditionals, methods, parameters, variables, arrays, and recursion.

The mechanical supports Alice provides can help broaden the pool of computer science majors. A 2004 study found that using Alice 2.0 increases the academic success and retention of at-risk college students—freshmen intending to major in computer science who enter college with no programming experience and/or are not prepared to enroll in calculus as freshmen [4].

Storytelling Alice

No matter how easy doing something may be, most people still need a reason to want to do it. While making the process of learning to program less needlessly frustrating may help retain students already interested in computer science, it is not sufficient for attracting new students into the field. Storytelling Alice introduces computer programming to middle school girls as a means to the end of creating animated movies [4]. From here forward, we refer to Alice 2.0 as "Generic Alice" and the version of Alice we modified to better support storytelling as "Storytelling Alice" (see the table here).

Storytelling Alice	Generic Alice
Say, think	Move
Play sound	Turn
Walk to, walk offscreen, walk	Roll
Move	Resize
Sit on, lie on	Play sound
Kneel	Move to
Fall down	Move toward
Stand up	Move away from
Straighten	Orient to
Look at, Look	Point at
Turn to face, turn away from	Set point of view to
Turn	Set pose
Touch, Keep Touching	Move at speed, turn at speed, roll at speed

Animations in Storytelling and Generic Alice compared.

More than 200 middle school girls (most were Girl Scouts, ages 11 to 15, from the greater Pittsburgh area) participated in the formative evaluation that informed the design of Storytelling Alice (see Figure 1). Over 18 months in 2004 to 2005, we created and tested 15 different versions of Storytelling Alice. In trying to improve it, we were guided by the girls' storyboards of what they wanted to build, observed problems and listened to questions that came up during user testing, and recorded logs of the actions users took interacting with Storytelling Alice.

Based on this testing, we made three major changes to the Alice system:

Added high-level animations and support for creating multiple scenes. By analyzing the storyboards the girls created, we identified a set of high-level animations that would enable them to make better progress on their stories while providing natural motivation to use such programming constructs as methods, loops, and parameters;

Created a library of 3D characters and scenery that helps spark story ideas. One determining factor in girls' motivation to learn to program in Storytelling Alice is whether or not they have a story they want to tell. We found that the library of 3D characters with custom animation (such as a robot character with an animation called "crazy go nuts") inspired stories with topics ranging from too much parental pressure to romantic relationships gone awry that culminated with the robot going crazy; and

Created a story-based tutorial we present through a new interaction technique we developed called "Stencils." The Stencils-based tutorial is spatially overlaid on top

Figure 3. Scene from the Sims 2.

of the running Alice application (see Figure 2). The technique allowed us to create substantially richer, detailed tutorials, helping underscore that the system is useful for storytelling [4].

The Generic Alice tutorial was designed around examples that would demonstrate concepts as simply as possible. We found that the Storytelling Alice tutorial had to introduce concepts within the context of stories similar to the ones the girls imagined creating. Stencils moderates the additional complexity of stories by presenting instructions one at a time in the context of the application and prevents users from accessing user interface components that are not necessary for the current step.

An initial fear we had about using storytelling as an end goal to motivate programming was that girls would mostly want to tell stories requiring only a simple sequence of instructions. However, we found that while many girls do start with such sequences, the activity of storytelling provides a graceful, gradual transition to more complex programming concepts and constructs.

To provide insight into the kinds of programming constructs girls' stories motivate using, we analyzed storyboards and programs created by 23 girls representing seven Pittsburgh Girl Scout troops. They created their storyboards, as well as their programs, in a single four-hour session. Participants had never seen or used Storytelling Alice prior to the user-testing session. On average, the girls in the study spent 1–1.5 hours on their storyboards and 2–2.5 hours learning to use Alice and write their programs.

3D Characters and OO Programming

Storytelling is a natural way to introduce the basic concepts behind object-oriented programming. Characters are reified objects that know how to perform a variety of methods. Objects have distinct sets of methods; a teacher may have a "scold (character to scold)" method, while a student does not. One challenge in teaching programming to beginning programmers is how to motivate the use of methods. We found that the need for multiple scenes in stories provides natural motivation to use methods. The girls in the study wanted to be able to work on the action for their second scene without having to watch their entire first scene in order to do so; 87% of girls' storyboards included multiple scenes, naturally motivating the use of methods. Many of the actions study participants wanted their characters to perform provide natural motivation for creating methods that take

parameters. Examples include multiple students jumping through a window and a character skateboarding into and knocking over various pieces of furniture; 52% of the girls' storyboards included methods that needed parameters.

The detailed gestures users want their characters to perform often involve repeating motions that naturally motivate loops. Specific examples of actions that involve loops drawn from girls' storyboards include: bouncing a basketball, making a fairy fly by having her flap her wings, and petting a dog; 48% of the girls' storyboards included actions that require looping.

Covering all these topics would typically take several weeks in a programming class. While the study participants did not become expert programmers in two hours, they did make progress. All of them created a sequential program in Storytelling Alice; 87% created a program with multiple methods; and, in their first two hours of programming, several were using loops and/or creating methods that take parameters.

Creating an animated movie and learning to program a computer CAN BE FUNDAMENTALLY THE SAME ACTIVITY.

In 2003, the ACM K–12 Task Force Curriculum Committee released a model curriculum for K–12 students to serve as a guide for teaching computer science-related skills to pre-college students [9]. The curriculum recommends that ninth- and tenth-grade students be introduced to five basic programming topics:

- Variables, data types, and representation of data in computers;
- Managing complexity through top-down and object-oriented design;
- Procedures and parameters;
- Sequences, conditionals, and loops; and
- Tools for expressing design.

Most are naturally introduced through storytelling. In our experience, those that do not arise effortlessly—conditionals, variables, and data representation—can be motivated through assignments. In traditional programming courses, all topics are typically motivated through assignments. Storytelling Alice has enabled us to introduce most of the programming concepts to students as early as fifth grade, possibly enabling the introduction of more complex programming and core ideas of computer science before they finish middle school.

To determine the effect of the storytelling focus on girls' motivation to learn to program computers, we conducted a study comparing girls' performance with and interest in learning to program using Storytelling Alice and Generic Alice. By comparing versions of Alice with and without storytelling support, we were able to isolate the effect of the storytelling focus and prevent performance differences caused by the mechanical support for programming provided by Alice.

We found that girls using Storytelling Alice were more motivated to program, as evidenced by the fact that they were three times more likely to sneak extra time after we had called "time up" to continue working on their programs. Moreover, the girls who used Storytelling Alice expressed a stronger interest in taking a future Alice course. Informal user testing suggests that storytelling provides similar motivation for boys. Storytelling works for three main reasons:

It gives girls an opportunity for self-expression. Nearly all the girls we've worked with in both the development and evaluation of Storytelling Alice find a story idea they are motivated to work on. Girls who have a vision for their stories worked through obstacles they encountered rather than give up;

It gives girls an opportunity to think through issues they encounter in their own lives. Forming an identity independent of their parents and friends is an important activity for all adolescents. By creating stories in which they experiment with a variety of roles, girls are able to explore social status, romantic relationships, and their relationships with the authority figures in their lives; and

It allows girls to share stories with friends and classmates. Many of the first programs we assign to beginning programmers are difficult for nonprogrammers to appreciate. Few people are excited by a demonstration of the canonical first program "Hello World." The stories girls create can be appreciated by anyone, including nonprogramming peers and adults. More-

STORYTELLING IS A NATURAL WAY TO INTRODUCE THE BASIC CONCEPTS behind object-oriented programming.

over, where computer games are typically best experienced individually, stories can be viewed and enjoyed by a whole classroom of students simultaneously.

Based on our success motivating girls to learn to program, in combination with anecdotal tests with a broad age range of male, as well as female, students, convinces us that storytelling can help motivate a broad audience of people to learn basic computer programming. Alice 3.0, scheduled for release in 2008, will take a storytelling approach through the characters from the popular Sims 2 PC and video game (see Figure 3). The Sims 2 was donated for use in Alice 3.0 by Electronic Arts (www.ea.com) to help inspire students to pursue computer science. The professional production values of the Sims 2 characters will further enhance students' feelings of accomplishment when using the Alice system.

Conclusion

In searching for ways to make computer science more appealing to students, educators must think about more than simply the numbers of students they enroll. When creating curricula based on games, they need to be sure that the games they choose really are able to motivate students of both genders. Yet games like the Sims that appeal strongly to both genders are far too complex for first-semester students. As we search for games that are of reasonable scope for beginning students in a game world that is not gender-neutral, we risk choosing games (such as first-person shooters) that disproportionately appeal to male students. Unless we take care in the way we design our gaming curricula, demonstrating broad appeal among the larger gaming community, we risk further reduced diversity in computer science. How could we possibly consider it a success if we doubled the number of students majoring in computer science while further reducing the percentage of computer science majors who are women?

References

1. American Association of University Women Educational Foundation. *Girls in the Middle: Working to Succeed in School.* Washington D.C., 1996; www.aauw.org.
2. Entertainment Software Association. *Essential Facts about the Computer and Video Game Industry: 2005 Sales, Demographics, and Usage Data.* Washington, D.C, 2005; www.theesa.com/files/2005EssentialFacts.pdf.
3. Hecker, D. Occupational employment projections to 2014. *Monthly Labor Review 128,* 11 (Nov. 2005), 70–101.
4. Kelleher, C. *Motivating Programming: Using Storytelling to Make Computer Programming Attractive to More Middle School Girls.* Carnegie Mellon University Ph.D. Dissertation, Sept. 2006; www.cs.cmu.edu/~caitlin/kelleherThesis_CSD.pdf.
5. Kelleher, C. and Pausch, R. Lowering the barriers to programming: A taxonomy of programming environments and languages for novice programmers. *ACM Computing Surveys 37,* 2 (June 2005).
6. Moskal, B., Lurie, D., and Cooper, S. Evaluating the effectiveness of a new instructional approach. In *Proceedings of SIGCSE* (Norfolk, VA, Mar. 3–7). ACM Press, New York, 2004, 75–79.
7. NetShelter. *NetShelter Reaches Enterprise Decision Makers and IS/IT Professionals, Technology Enthusiasts, Consumers, and Hardcore Gamers.* Thornhill, ON, Canada, 2004; www.netshelter.net/media_kit/audience/.
8. Schiesel, S. Video games are their major, so don't call them slackers. *New York Times* (Nov. 22, 2005), A21.
9. Tucker, A., Deek, F., Jones, J., McCowan, D., Stephenson, C., and Verno, A. *A Model Curriculum for K–12 Computer Science: Final Report of the ACM K–12 Task Force Curriculum Committee* (Oct. 2003); www.csta.acm.org/Curriculum/sub/k-12modelcurr2nded.pdf.
10. Vegso, J. Drop in CS bachelor's degree production. *Computing Research News 18,* 2 (Mar. 2006), 5.
11. Ziff-Davis Publishing Holdings, Inc. *Readership Facts: Electronic Gaming Month*; www.ziffdavis.com/products/print/egm.
12. Zweben, S. Ph.D. production at an all-time high, with more new graduates going abroad; Undergraduate enrollments again drop significantly. *Computing Research News 18,* 3 (May 2006), 7–17.

Caitlin Kelleher (caitlin@cs.cmu.edu) is a post-doctoral fellow in computer science and human-computer interaction at Carnegie Mellon University, Pittsburgh, PA.

Randy Pausch (pausch@cs.cmu.edu) is a professor of computer science, human-computer interaction, and design at Carnegie Mellon University, Pittsburgh, PA, where he is a co-founder of Carnegie Mellon's Entertainment Technology Center and director of the Alice Project (www.alice.org).

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, to republish, to post on servers or to redistribute to lists, requires prior specific permission and/or a fee.

© 2007 ACM 0001-0782/07/0700 $5.00