Using Game Creation for Teaching Computer Programming to High School Students and Teachers

Mohammed Al-Bow[2], Debra Austin[4], Jeffrey Edgington[2] Rafael Fajardo[1,3], Joshua Fishburn[3], Carlos Lara[2], Scott Leutenegger[2], Susan Meyer[1]

[1]Art and Art History, [2]Computer Science, [3]Digital Media Studies, [4]Education

ABSTRACT

In this paper we describe a two-week residential summer game camp for rising 9th and 10th grade students and a four-week high school teacher professional development course. We present survey results that indicate our approach results in increased computer programming knowledge and self-confidence for both students and teachers. Our project aims to use a holistic game creation approach to increase student interest in computer science by directly teaching to students in a summer camp, instructing the teachers during a 4-week professional development course, and finally by supporting teachers as they use our curriculum in their high schools.

Categories and Subject Descriptors

K.3.1 [**Computers and Education**]: Computer Uses in Education, K.3.2 [Computer and Information Science Education]: Computer Science Education, Curriculum

General Terms: Design, Human Factors, Languages

Keywords: Games, High School Teacher Professional Development, Introductory Programming

1. INTRODUCTION

Capitalizing on youth's interest in games, we have developed a curriculum that uses the *creation* of computer games to integrate computer science, art, and design instruction in a project-based learning model. Our curriculum is designed for high school 9th and 10th graders (14 and 15 year olds) and is intended as a first course in programming. In our course students design and create their own computer games; learning programming concepts such as variables, control statements, classes and objects, coordinate systems, loops, and arrays. We have directly delivered our curriculum during a residential summer game camp, taught our curriculum to high school teachers through a professional development course, and indirectly delivered our curriculum to students in Denver metropolitan high schools.

The well documented drop in incoming computer science majors, especially women, is troubling. One possible approach is to increase high school teacher comfort and competence with teaching computer science. By increasing the competence of high school teachers and students through the use of our game-creation based curriculum we hope to increase the number of students entering Science, Technology, Engineering, and Mathematics (STEM) disciplines. We have created the Pixels, Programming, Play & Pedagogy (*P4Games)* project to leverage student interest in games toward teaching students and teachers introductory programming in a project-based inter-disciplinary learning model. Our project has three major components:

1. **Residential Game Camp For 9th and 10th grade students.** The *P4Games* curriculum was developed and tested in a residential summer program for high school students. Full scholarships are supported by the NSF. The camp also functions as training lab for high school teachers participating in the Teacher Game Institute.
2. **Teacher Game Institute (TGI).** TGI offers 120 hours of professional development for up to 20 Colorado teachers per year. TGI includes the same curriculum taught to the students and is augmented with a pedagogy component to improve knowledge in project-based learning and performance-based assessment theory as related to our interdisciplinary game development pedagogy. TGI instructs high school teachers to adapt the *P4Games* curriculum to create learning opportunities appropriate to their individual school environments. Teachers also have the option to also enroll for graduate credit in our College of Education.
3. **School Implementation.** The P4games project provides graduate students to help teachers implement the curriculum into their classrooms. Using a "framework" approach, the curriculum provides teachers with the flexibility to implement a full game development course or embed game development exercises into existing courses including those in other disciplines. In 2007 – 2008 4 sections of approximately 30 students each were delivered in two high schools. During the first semester of 2008 – 2009 our curriculum was delivered to about 300 additional students at 4 Denver metro high schools.

In this paper we give a brief overview of the P4games program and present survey results from our 2008 Summer Women's Game Camp and our 2008 summer Teacher Game Institute.

2. RELATED WORK

Games are currently being used in many college classes as an engaging way to teach various computer science topics including introductory programming, graphics, software engineering, artificial intelligence, and networks. The number of papers suggesting this approach is already quite large and we refer the reader to recent proceedings of conferences such ACM SIGGRAPH Sandbox, Future Play, Conference on Game Development in Computer Science Education, Foundations of Digital Games, and SIGCSE.

To our knowledge little has been done to reach out to high school students using game creation nor to deliver high school teacher professional development courses in programming via game

ITiCSE'09, July 6–9, 2009, Paris, France.

creation. Early work on using game creation for teaching introductory programming to school aged children was done by Kolling and Henriksen [6]. In the summer of 2006 we piloted a 9-day mixed gender game camp. We used Flash and Actionscript for game creation. A description of the initial camp is found in [4]. Subsequently we obtained NSF funding and expanded the camp to also include a professional development course with targets of 45 students and 20 teachers per summer.

Beginning in summer of 2007 we switched to using Java within the Greenfoot [5,6] programming environment. We chose Greenfoot because we believe it is a solid tool that provides many of the needed constructs for creating 2D computer games at a level that is especially appropriate and fun for novice programmers. During the two weeks, we teach variables, if statements, classes and objects, 2D coordinate systems, loops, and arrays. Our goal is to get students excited about programming and the possibility of studying computer science by taking subsequent courses in high school and/or college. In a recent paper [1] we describe our specific programming approach and programming topics covered in more detail. In 2008 we shifted our focus and targeted our camp marketing toward young women. A conference presentation describing the camp and its focus on young women occurred at the 2008 Women in Games Conference [2]. In this paper we present survey results for both the 2008 women's camp and the 2008 professional development course.

3. GAME CAMP CURRICULUM

Our two-week residential summer game camp is for rising 9th and 10th graders (14 and 15 year old students). The camp serves three purposes: 1) to directly reach students and encourage them to consider STEM disciplines; 2) to serve as a curriculum model for high school based offerings; and 3) to serve as a training lab for high school teachers.

The camp has 10 days of instruction. Each day is divided into four 2-hour classes. During the first 6 days the 4 classes each day are: Programming, Pixels (art), Play (game design) and period of outdoor time or game lab time. During the art sessions, students are introduced to visual asset development using traditional and digital studio practices. The curriculum focuses on the figure and the scene. First, students make three-dimensional wire figures. Using these figures, two-dimensional space is addressed and gesture drawing introduced, resulting in picture planes loosely depicting the illusion of space through line weight, proportional shifts, placement, and horizon line. Gesture drawings of the human figure follow, as a live model enacts short, multi-step action poses. A move to the digital realm results in jointed characters ready to be programmed in an animated sequence. Students then focus on the scene using construction paper and scissors to create narrative compositions taken from "Picture This" [3]. Relying on shape, placement, proportion and color to depict narrative, the collages offer a model easily transferred to the digital environment. Instruction on the basics of one-point, two-point and three-point perspective follow. Studio work concludes with an introduction to value, light and shadow.

In the game design sessions, we employ the rapid creation and iteration of paper-based games to teach fundamental design skills. Students are given a working definition of games taken from Salen and Zimmerman [7] and then challenged to create a game in the first thirty minutes of the first design session. They are then introduced to the concept of critical and reflective play as a way to assess the quality of the play experience. Students are encouraged to develop helpful critical vocabularies by play testing and commenting on each others' games. Students then are given opportunities to refine their games and have them replayed. Students are given a simple and clear set of quality standards so that "good" games could be identified. We also model and described Humane Games in order to encourage the broad exploration of different game types.

In the programming sessions students start with playing a pre-made game to get them comfortable using the Greenfoot IDE and then create a simple game by adding one step of complexity at a time. The Greenfoot API includes an Actor class and World class that provide many needed methods including getLocation and setLocation, creating/removing objects, object intersection tests and keyboard/mouse control. We introduce computer programming concepts in conjunction with logical next steps in building a game. We first create sprites and move them using Greenfoot API methods. Next, we make the sprites move back and forth reflecting at boundaries thus necessitating if-statements and adding data members for current direction and/or velocity. We then solidify understanding of the coordinate system by having students place sprites using code to spell out block letters using for-loops. We next add keyboard control of sprites and use of the Greenfoot provided intersection test for collision detection. Next, we add simple patrolling behavior AI for non-player-characters. Finally, object members are added to keep track of game counters such as timers and the number of objects remaining in the game. A more detailed description of our programming approach and some utility classes we added to Greenfoot to make game programming a bit easier can be found in [1].

During the last 4 days students have all day workshop time to build their own game. The majority of students actively seek us out for help rather than us having to push them to work, thus demonstrating the power of a constructivist learning model when the task is something the students want to do.

4. TGI CURRICULUM

The Teacher Game Institute is a four-week professional development course. The curriculum during the first two weeks is nearly identical to the Game Camp Curriculum. In addition, the 4th two-hour period each day is spent exploring related pedagogical issues. The book "How Computer Games Help Children Learn" [8] is used as a primary text and augmented by web-based articles and resources. Covered topics include: Learning Via Game-play; Technology Standards & Policy; STEM Education and STEM Career Resources; Designing Learning Environments; Digital Age Students; Project-Based Learning; Performance Based Assessment; Creativity; Intellectual Property; Innovation & Design; and Humane Games.

During the second two weeks teachers are divided into groups to create Humane Games. Having the teachers create a more substantial game solidifies the concepts learned during the first two weeks. By having the teachers create Humane Games they come to believe they can do the same with their students. The final two weeks of the Teacher Game Institute also overlap with the student game camp allowing the teachers to observe the curriculum as it is being taught and learn first hand the issues that are likely to occur in their classrooms. During the second week of camp teachers spend two hours per day helping students.

5. SURVEY RESULTS

In this section we present results for our 2008 game camp and 2008 professional development course. The 2008 game camp was for young women only. The ethnic mix included two African Americans, one East Indian, eight Latinas, and sixteen Caucasians. We had 26 students take surveys before and after the camp. During the teacher game institute, 8 teachers completed both surveys. The pre-survey occurred before any instruction and the post-survey occurred on the last day. The pre-survey and post-survey consisted of identical questions. We first discuss the results for the students and then for the teachers.

In the pre-survey 5 girls stated they intended to major in a computer/technology/programming related field where as in the post-survey the number increased to 9. The near doubling of interest is an indication that our approach is working. We next explore specific computer science knowledge gained by students during the camp. The computer science knowledge questions were as follows:

The first three questions assume the following block of code:

```
int num1 = 3 ;
int num2 = 8 ;
int num 3 ;
num3 = num1 * num 2 ;
num2 = num1 + num2 ;
```

- *Trivial variable:* "After the above code num1 contains" {0, 3, 8, 11, Don't Know}
- *Variable addition*: "After the above code num2 contains" {24, 3, 8, 11, Don't Know}
- *Variable multiplication*: "After the above code num3 contains" {24, 3, 8, 11, Don't Know}

The next question assumes the following block of code:

```
int num = 0 ;
for (int i = 6 ; i < 10 ; i++)
{
 num = num + 1 ;
}
```

- *Understand for-loop:* "After the above code num contains": {0, 3, 4, 5, Don't Know}

The remaining questions were:

- *Simple Object Oriented Concepts:* "Member variables hold the data associated with an object" {true, false, Don't Know}
- *Simple Object Oriented Concepts:* "An object is a particular instance of a class" {true, false, Don't Know}
- *General If-statements:* If statements allow a program to make decisions based on current conditions" {true, false, Don't Know}
- *Coordinates and Movement:* "Describe the motion of a greenfoot actor with the below act( ) method:

```
Public void act( )
{
        if ( getX() > 20 )
            setLocation( 1, getY( ) ) ;
        else
            setLocation( getX() + 1, getY() ) ;
        if ( getY() > 40 )
            setLocation ( getX() , 1 ) ;
        else
            setLocation( getX(), getY() + 1) ;
}
```

- *Write Own for-loop* Assuming System.out.println(num) prints the contents of variable 'num', write a for-loop to print the following 6 lines of output:

  3
  6
  9
  12
  15
  18

  Note, when grading the answers to this question we did not mark "off-by-one-errors" as wrong.

In Table 1 we summarize the survey results. The "DK" in the table header sections is an abbreviation for "I Don't Know". As can be seen there was a substantial improvement. Especially encouraging is the reduction in the number of "I Don't Know" answers, even if the answers were wrong, as this shows more confidence in student ability to try to answer the question. At this level it is arguably more important to increase student confidence and excitement concerning technology rather than getting all the answers 100% right. We note that the coordinate system question appears to have been too difficult.

We next turn to the student learning in Art and Design. We believe that our approach's promise is due to a combination of using games as subject matter and the integration with Art and Design. Thus, it is equally important to look at the Art and Design learning to understand why the computer science thinking has improved. In the surveys students were asked to self-assess their knowledge by answer the following questions with Strongly Agree (5); Agree (4); Neutral (3); Disagree (2); Strongly Disagree (1). In Table 2 we present the survey results. As can be seen there is a substantial increase in student knowledge and confidence. We believe these increases also partially explain the increases in computer science abilities.

We applied a paired t-test to the data resulting from the questions in Table 2. Three questions produce $p > .05$ ("I know what a game is", "I can provide an affirmative definition of a game" and "I think I can use shape to tell a story"). The question "I know what design is" resulted in a value of $p < .05$. All other questions resulted in values of $p < .01$.

We now present the 2008 teacher results. The teachers were first asked the same questions as the students and then asked additional questions about the utility of game creation for teaching various concepts. Our sample size was only 8 teachers thus it is hard to draw statistically sound conclusions but the raw data is still telling. In Tables 3 and 4 we present results from the same questions as used for the students. Given that five of the eight teachers had already taught programming it was not surprising to see that they already knew most of these elementary questions in the pre-survey. Despite the high scores on the pre-surveys there was further improvement presumably for the three teachers who had not previously taught programming. The Art and Design topics on the other hand were new to most of our teachers and there was a substantial improvement in knowledge and self-confidence in these areas. For all but one of our 2008 teachers, who was a graphic arts teacher, the idea of teaching game design and art was intimidating. Thus, it is very important to instill confidence in them before they take the curriculum to the classroom. Our survey results in Table 4 indicate that this confidence building did occur.

In Table 5 we present survey results concerning the pedagogy of game creation. We asked the teachers if they agreed with various statements about the efficacy of game creation for teaching a wide range of topics. The specific questions are reproduced in the table. Given that the teachers had already self-selected to take this professional development class it is not too surprising to see that they entered the experience with a positive attitude towards the utility of games, but even so they left with a significantly higher opinion of the promise of using game creation to teach a wide variety of knowledge.

Table 1: 2008 Camp Student Programming Assessment

Question Topic	Pre-Survey	Post-Survey
	Right,Wrong, DK	Right,Wrong, DK
Trivial Variables	14, 1, 11	17, 4, 5
Variable Addition	3, 12, 11	5, 13, 8
Variable Multiplication	3, 9, 14	4, 12, 10
Understand for-loop	1, 11, 14	10, 7, 9
Simple OO concepts (methods)	4, 2, 20	15, 3, 8
Simple OO concepts (Object instance)	7, 0, 19	18, 3, 5
General if-statements	10, 1, 15	12, 2, 12
Coordinate system and movement	0, 2, 24	6, 11, 9
Write own for-loop	0, 0, 26	3, 5, 18

Table 2: 2008 Camp Student Self Assessment on Art and Design

	Pre-Survey	Post-Survey
I know what a game is	4.42	4.46
I can provide an affirmative definition of a game	3.50	3.73
I know what a humane game is	2.81	4.0
I know what design is	3.92	4.31
I know what game design is	3.58	4.31
I can design a paper-prototype of a game	2.65	4.0
I think I can draw	3.35	4.08
I think I can use shapes to tell a story	4.04	4.27
I think I understand how to represent depth or space visually	3.19	4.23
I think I understand the emotional power of color	3.88	4.38
I think I know how to make size matter	3.00	4.0
I understand what makes a good visual composition	3.23	4.0

Table 3: 2008 TGI Teacher Programming Assessment

Question Topic	Pre-Survey	Post-Survey
	Right, Wrong, DK	Right, Wrong, DK
Trivial Variables	5, 3, 0	7, 1, 0
Variable Addition	5, 3, 0	6, 2, 0
Variable Multiplication	1, 7, 0	3, 5, 0
Understand for-loop	7, 1, 0	6, 2, 0
Simple OO concepts (methods)	4, 4, 0	6, 2, 0
General if-statements	6, 0, 2	6, 2, 0
Coordinate system and movement	3, 1, 4	4, 2, 2
Write own for-loop	4, 1, 3	6, 2, 0

Table 4: 2008 TGI: Teacher Art and Design Self Assessment

	Pre-Survey	Post-Survey
I know what a game is	4.375	4.875
I can provide an affirmative definition of a game	3.75	4.75
I know what a humane game is	3.625	4.875
I know what design is	4.25	4.875
I know what game design is	4.125	4.75
I can design a paper-prototype of a game	3.75	5.0
I think I can draw	3.5	4.75
I think I can use shapes to tell a story	3.875	4.75
I think I understand how to represent depth or space visually	3.625	4.625
I think I understand the emotional power of color	4.0	4.5
I think I know how to make size matter	3.625	4.625
I understand what makes a good visual composition	3.75	4.5

Table 5: Teacher Survey on Advantages of Game Approach Over Traditional Approaches		
I think that teaching game creation is more effective than traditional pedagogies for:		
	Pre-Survey	Post-Survey
Encouraging background research to gain knowledge of a topic	4.0	4.375
Developing creativity with technology skills	4.75	4.75
Integrating instruction in art, technology and mathematics	4.625	4.875
Teaching design methodology including the design, create, and revise lifecycle	4.625	4.75
Teaching computer programming	4.375	4.75
Improving student understanding of coordinate systems	4.0	4.625
Improving student understanding of elementary mathematical concepts, including variables	4.25	4.625
Improving student understanding of elementary mathematical concepts, including equations	4.0	4.5
Improving student understanding of elementary mathematical concepts, including mathematical models	4.0	4.125
Improving computer literacy	4.625	4.75
Improving student confidence in computer skills	4.75	4.875
Teaching logic as demonstrated in game rules	4.5	4.625
Developing student creativity and innovation	4.75	4.875
Teaching critical thinking	4.5	4.75
Teaching problem solving	4.5	4.75
Teaching decision making	4.375	4.75
Developing global citizenship skills	4.25	4.5
Demonstrating the relevance of math	4.375	4.5
Demonstrating the relevance of art	4.5	4.75
Demonstrating the relevance of design	4.625	4.75
Improving student confidence in art skills	4.125	4.625
Encouraging student reflection	4.125	4.625
Developing communication skills	4.25	4.375
Developing social responsibility	4.25	4.375
Encouraging student interest in STEM disciplines	4.25	4.875

6. SUMMARY AND FUTURE WORK

We believe that our game creation approach to teaching introductory programming shows great promise for engaging high school students in programming and increasing interest in computer related fields of study. Student survey results from our 2008 summer game camp demonstrate a significant improvement in computer programming knowledge, and, more importantly, self - confidence. We believe teaching game creation as a holistic discipline integrating programming, art, and design makes the approach compelling and fun for the student. Thus, it is essential that teachers feel competent in the game creation process, not just programming. Teacher survey results form our 4-week professional development course indicate a strong increase in knowledge, confidence and belief in our approach. In the 2008-2009 school year are currently working with 8 teachers to deliver variations of our curriculum to 350 – 500 Denver metro students. We plan to use and evaluate a new rubric for game education and share our results in the future.

7. ACKNOWLEDGEMENTS

Our thanks to the National Science Foundation, Colorado Council for the Arts, University of Denver School of Art and Art History, and Electronic Arts and Colleen McCreary for their continued support. This work is supported under NSF grant number ESI-0624767.

8. REFERENCES

[1] Al-Bow, M., Austin, D., Edgington, J., Fajardo, R., Fishburn, J., Lara, C., Leutenegger, S.T., Meyer, S., "Using Greenfoot and Games to Teach Rising 9th and 10th Grade Novice Programmers", Proc. of ACM SIGRAPH SANDBOX 2008, August, 2008.

[2] Al-Bow, M., Austin, D., Edgington, J., Fajardo, R., Fishburn, J., Lara, C., Leutenegger, S.T., Meyer, S., "Motivating Young Women in Game Development Via the Pixels, Programming, Play & Pedagogy Project", presentation at the Women In Games Conference, Warwick England, September 2008

[3] Bang, Molly**, Picture This**, SeaStar Books. San Francisco, CA. 1991.

[4] Fajardo, R., and Leutenegger, S.T., "Programming, Pixels & Play : A University Summer Game Camp To Attract Under-represented Populations to Game Development and Computer Science", Proc. of Future Play, October, 2006, London, Ontario Canada

[5] Greenfoot: DOI= www.greenfoot.org .

[6] Kolling, M., and Henriksen, P., "Game Programming in Introductory Courses With Direct State Manipulation", Proc. of ITiCSE'05, Lisbon, Portugal, 2005.

[7] Salen, K., Zimmerman, **Rules of Play**, MIT Press, Cambridge, MA, 2004.

[8] Williamson Shafer, D., and Gee, J.P., **How Computer Games Help Children Learn**, Pallgrave Macmillian, 2008.