COMPUTER GAMING AT EVERY AGE: A COMPARATIVE EVALUATION OF ALICE

By

CHERYL D. SEALS*, YOLANDA MCMILLIAN**, KENNETH ROUSE**, RAVIKANT AGARWAL***,
ANDREA WILLIAMS JOHNSON**, JUAN E. GILBERT****, RICHARD CHAPMAN*

*Associate Professor, Computer Science and Software Engineering Department, Auburn University.
**Doctoral Student, Computer Science and Software Engineering Department, Auburn University.
***Assistant Professor, Computer Science, Hope College.
****T-Sys Distinguished Professor, Human Centered Computing Lab, Computer Science and Software Engineering, Auburn University.

ABSTRACT

This research has two thrusts of teaching object oriented programming to very young audiences and of increasing student excitement about computing applications with the long-term goal of increasing involvement in technology classes, in the use of computer applications and interest in technology careers. The goal of this work was to provide challenging interactive activities for young students that integrate their courses with computer technology. The authors utilize game development and interactive storytelling as a motivator for introductory programming training. The authors identified that many of their young second through fifth grade students showed much promise and ingenuity in programming when using visual programming environments. The hypothesis of the study was that the young students would fare as well as introductory college students when completing introductory programming tasks. The environment utilized for this experiment was Alice 3D and the authors hope was that the youngsters would perform at levels complimentary to those of the college students to illustrate that there is no age limit on ingenuity, when the proper training and tools are provided. This work discusses the outcome of a college introductory assignment that would be given to both elementary school and college students enrolled in an introductory computer science course.

INTRODUCTION

In the American school system many students feel that they are not appropriately challenged. Many students do not have the motivation to do the work and in many cases they stop attending school. In addition to the students that simply drop out, the ones who are high achievers do not feel stretched to meet their full potential for a multitude of reasons. The reasons include the "no child left behind" clause, where more focus is being placed on those students who are struggling. This practice leaves more advanced students bored and neglected. This issue highlights the necessity for a more challenging curriculum, instead of just meeting the minimum criteria. In order to uphold the state requirements, teachers have to be focused on all their students passing the assessment exams to graduate on to the next grade rather than imparting education to stimulate their minds. This is an issue that need to be addressed at a fundamental level.

There is a needs to be truly innovative in education to address the needs of all students and give them a curriculum that will motivate them to excel. Therefore, greater resources are required to support creative and challenging curricula with provisions for enhanced materials, teacher support and training. One method is to take advantage of intrinsic motivation. "Intrinsic motivation, also known as self-motivation, refers to influences that originate from within a person, which cause a person to act or learn" (Bomia et al., 1997).

From their local school district, the authors investigated and found similar trends to the ones mentioned above. To alleviate some of these problems and to introduce students to technology, a computer based program Auburn University Computer Science & Software Engineering STARS Computer Club (STARS Computer Club) was started. This included using visual programming software called Scratch, Squeak, Lego Mindstorms and Alice 3D. In this program the authors incorporated concepts from physical science, computing and reading comprehension.

The STARS Computer Club was adopted as an innovative method to challenge the exceptional kids and give marginal students (i.e. those disinterested in school) motivation and a foundation to get back to work and be directly involved in improving their educational future. To assist students at all levels, the authors introduced them to multiple environments (i.e. from struggling through gifted) which helped them to develop a stronger foundation and passion for learning. The aim of the program was to motivate students to be more actively involved in their education by developing their own intellectual capacity for learning (Papert 1980). When designing the initial program, the authors brainstormed methods to target student interest, learning styles, and cognitive needs while developing enrichment activities to have the greatest impact on student motivation and their acquisition of knowledge.

Background

The issue of falling enrollment rates in computing based majors has become a prominent problem. Student interest in computer science is falling worldwide and between 2000 and 2005 incoming freshman interest in computer science dropped by 70% in the U.S. (Kelleher, 2007). The Taulbee survey found that computer science enrollment at research universities dropped by 50% (Vegso, 2006). It also indicated that 84.9% of bachelor's degrees were awarded to men in computer science. As a result, literature was explored to provide tools and ideas that could be integrated and applied to take a step towards providing a solution. In order, to keep attracting the brightest minds in a more diverse context, the computer science community must ensure that new computing-based technologies and curricula meet the diverse needs of the global population and find ways to attract more diverse student groups.

There is growing interest by youth in video gaming. There was a 15% increase in computer game sales in 2006 (Brightman, 2006), and the hope of the authors is that they can leverage this phenomenon to draw more undergraduate students into the field. In 2008, with an ailing economy "Video game sales keep bucking recession trend… as an exception to the worldwide gloom". There is a theory that the gaming industry "may be recession-proof" (Terdiman 2008). With such strong interests in video gaming, in many cases this may prove as a great technique for recruiting students into the field of computer science. In searching for ways to make computer science more appealing to students, educators must think fundamentally about what culturally relevant innovations can be used to enrich computer science. Making changes to the methods of student's first introduction to computer science can greatly affect future enrollments. When creating and modifying curricula identifying ways to motivate students of both genders and a wider variety of backgrounds is required.

In background review, the authors investigated other approaches to technology use in the classroom. Many elementary students struggle to read, write and comprehend in the classroom (Geary, 2006; Guensburg, 2006) and many programs have been created to help them with basic reading proficiency. Some programs have looked at solutions of utilizing computers to aid students in improving their reading literacy (Williams et al., 2009). It has been documented that programs need to better take advantage of the hours that students spend playing video games (Jones, 2003). Interactive educational games like video games can encourage students to be more responsive and are of greater benefit than inactive games (Bates, 2004). Students are highly interested in learning to design and create their own video games. As a result, many books have been created to take advantage of this trend toward game design study and computer gaming as a serious discipline that can improve student's intrinsic motivation for programming (Fullerton, Swain, & Hoffman, 2004; Rabin 2005). The authors have also utilized these texts in creating undergraduate and graduate classes and students created their own educational video games. Interaction with and creating video games has proven to greatly increase student motivation in the classroom (Gosha et al. 2008; Seals, Hundley, & Strange 2008).

The authors have studied these approaches and have found that an integration of computing with basic skills acquisition can be beneficial to students at all levels. The tactic of stealth of adding educational concepts to fun situations like game programming can be a way to provide support for a broad array of concept areas. Williams (2009) performed studies in the classroom with Squeak and SimBuilder, where young children in grades 2-4 improved their reading comprehension and learned to program video games that depicted lessons from physical science classroom content (e.g. interactive ecosystem development). Seals (2002) engaged students with AgentSheets in the classroom where K-12 teachers and their students were able to create interactive simulation microworlds with minimalist programming instructions.

For the last three years at Auburn University (AU) the authors have been facilitating a computer science after school program. In this program, the AU CSSE (Computer Science & Software Engineering) STARS Computer Club, the authors been utilizing introductory computer gaming and digital story creation to reinforce content matter dissemination with the added benefit of improving the computer efficacy of K-12 students and their teachers. Our teachers have been part of the process to support classroom management and help to reinforce content matter. One of the goals of the Computer Club team is to scaffold student's improving their problem solving skills, computer science knowledge, basic programming skills and to provide more opportunities for K-12 students and their teachers to have great access to educational software. A second goal of the program is to improve current and future college enrollments in computing and related fields. Many students are highly motivated by computer video games and the authors hope to graphically entice students into a long term affair with computer technology and in the long run increase their computer science enrollments.

The authors were inspired by “children's narrative development through computer game authoring” (Robertson & Good, 2005). This narrative looked at the benefits of children ages 12-15 in creating stories using medium interactive 3D virtual reality computer games. It used an authoring tool requiring very little prior programming knowledge and they discovered that this created a strong motivational influence for the young students.

The "Computer Club" Paradigm

The STARS Computer Club model to invigorate education was created to address our local battle to improve education and student interest. The state of Alabama has a difficult battle to improve education with the state's graduate rates dropping to 60%. High attrition rates and many other factors are very discouraging to educators that are faced with their educational initiatives not being successful. Many of our constituency no longer want to invest in the future of education or to improve the level of education in the state. This has resulted in lower general wages and a lower tax base in the state. The authors want to challenge students to achieve more in school and there hope is that intrinsic motivation that can be gained by this model can reduce dropout rates by getting students excited about learning.

The author’s plan was to introduce STARS Computer Clubs into all Auburn City Schools. In the first two years of the project, the authors focused their activities on Auburn Elementary Schools and in the upcoming year plan to incorporate Drake Middle and Auburn Junior High to provide enrichment activities to get students excited about education. Auburn University's department of Computer Science and Software Engineering started computer clubs in the elementary schools at the 4th and 5th grade level and through this study they found that these experiences have a positive effect of getting students excited about learning to utilize new technology, and excited about demonstrating their understanding of science concepts by creating digital simulations with visual programming techniques. The increased excitement and motivation caused by involvement in computer clubs has improved the grades of some students in other areas as well. This program has a long term benefit to the state to invigorate education and increase future graduation rates with hopes to interest more students in technology related fields.

The author's plan was to develop a model curriculum for teaching computational thinking from grades 4 through high school. This curriculum will be based on an experimentally verified cognitive hierarchy of computational thinking skills. The Computer Club program began three years ago to provide more exposure for the local youth to educational computer technology and improve their computer literacy with structured educational activities. The authors impression was that extracurricular and enrichment activities promised to provide great educational benefit to the partnering schools in their area. Based upon the experience, they plan to develop a methodology and lesson plans that can be exported and reused by any school which is interested in teaching a computer based curriculum. The authors intend to use web sites, videos, pod casts and publications to widely disseminate the "Computer Club" model as broadly as possible. The "Computer Club" methodology will improve basic problem solving and stimulate computation thinking through various visual programming experiences to enhance logic based reasoning, reading comprehension, and clarity of science concepts. The projects include Scratch visual programming, Alice visual programming, LEGO robotics and two programming competitions.

During year one of the program, the after school programs used the Squeak programming environment. Their projects were to improve their reading achievement with "Reading like a scientist" activities to stress the importance of reading as the requisite tool to become a proficient scientist. After learning about the phenomena each student designed a physical science lesson. The next step was to integrate reading materials into the project and design requirements into the creation of an animated or interactive physical science model. The culminating event of year one's activities were for the students to display their worlds as exhibits at Engineering Day. Engineering Day, better known as E-Day, is Auburn University's annual open house for the school of engineering. The impact of the initial program was that student's written communication by 4th graders increased average scores by 10% and 5th graders increased average scores by 77%. Also all students had an improvement to their reading comprehension & writing skills assessments. Self-efficacy of students improved in the areas of computing and problem solving. 5th graders felt that they learned new computer skills and had improvement in problem solving skills. Based on the results of the initial program the authors strongly believe that the "Computer Club" program will improve the level of educational motivation in their local area, the state and their country.

In year two of the program, the authors were equally impressed with the worlds created and the accomplishment of the participating Elementary school students. During year two of the program, they used the Alice 3D programming environment in an effort to provide more interest by broadening from 2D world to 3D worlds. The students were introduced to Alice3D with tutorials for three weeks and then supported for three weeks in order to create their own 3D interactive stories. Many of the students interacted with each other and they found that in many cases the elementary student's level of creation was very sophisticated and they had grasped some object oriented programming concepts; and, with the support of visual programming the students were performing in many cases equal to the undergraduates who were taking introductory programming classes. This paper intends to investigate the performance of these two groups; group one consisted of elementary students and group two consisted of students taking an introduction to programming class with no prior programming experience.

Computer Training for Activities

The students that were chosen for this were students that may have had instruction on using Squeak, but not Alice. Thus they all should have come into this with the same lack of experience as the undergraduate students, who were enrolled in ENGR 1110, Introduction to Engineering. The authors were told by the instructor of ENGR 1110 that no in-class instruction would be given to the students for using Alice but that they would be required to go through the 4 tutorials that are available for Alice. The elementary students were told that they would be given an assignment that college kids would be given. Likewise, they were started on the tutorials. To be sure the tutorials were actually completed; the ENGR 1110 students received grades for the work. The elementary students were asked a few questions about the topics to ensure the tutorials were completed. All students were required to complete the four tutorials in succession before they received the Alice assignment.

For the elementary school students, the authors used the first 4 weeks to go through the tutorials and then gave them a 50 minute session for the next two weeks to complete the assignment. The only down side is that all students were not able to come all 6 weeks, so some were behind the others and did not get to start on the assignment with the same amount of time as the others. Some students only had time to get the first part of a two part assignment completed. Another issue is that once the student started the assignment, the Auburn University CSSE assistants only gave minimal help to the students. This required the students to either figure it out on their own or go back to the tutorials. This was done deliberately, as the motivation of this was that the authors conjectured that even the average kids were not being challenged to the degree that they could be and the authors were going to test this by this study.

Detailed Instruction

The college students received very little formal training in Alice. The following is an example of their first introduction to the environment. At the beginning of the course, each college student was asked to experiment with Alice by creating a very simple world. The world consisted of a table set of four. Each setting on the table had a plate and mug. The students were given the picture in figure 1 and instructed to also add a cookie to each plate. The main concept of this assignment was to notice the different perspectives in Alice. In a particular view, everything appears to be in order, but switching to another view may show something unexpected, like cookies floating in the air that once appeared to be sitting on plates. Once this concept was mastered, each student was required to complete all four of the tutorials that are included with the Alice software program, as mentioned previously.

After initial training all students were then given the same Alice assignment. This assignment asked each student to create an island scene in Alice, starting with the water world template. They were also required to add an island object, fish object, and flamingo object with specific instructions on where to place the objects (see figure 2). Additionally, the students were asked to add a cloud object and set the lightning variable to invisible. Once the objects were set, the following behaviors were needed. The cloud was to move slowly into view, stop, and flash the lightning once. The flamingo was to turn and face the cloud quickly, and then turn slowly back around. The cloud should then move forward quickly, come close to the

Figure 1. Alice World - Table Setting

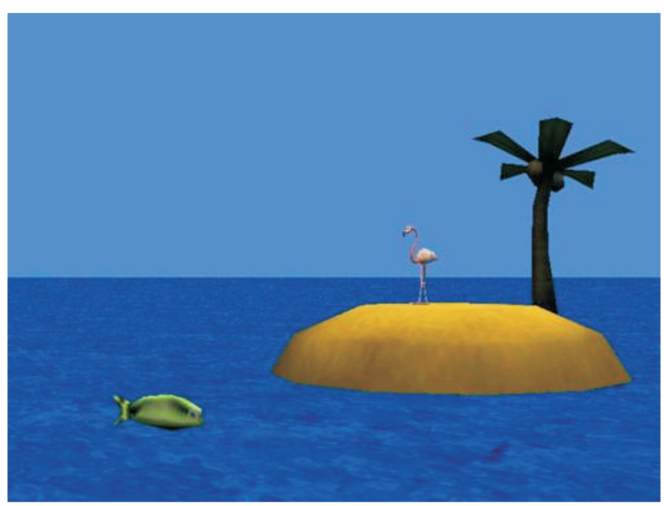

Figure 2. Alice World - Island Scene

flamingo, and flash lightning once more. The flamingo then was to move quickly away, but stay on the island. The fish was to make a ¼ revolution to its right and move quickly out of the camera's view.

Computer Gaming at Every Age

The college students and elementary students completed the assignment with few questions to their teaching assistants. There were generally no interactions in the lab environment between the students and the teaching assistants with regard to this assignment. Likewise, the students had little interaction between themselves with regard to questions about the assignment. The teaching assistants expected the students to do well on the assignment as there were few questions and it didn't appear that the students were having difficulty in understanding the specifications of the assignment or the Alice environment.

To compare the Alice worlds of both the college students and the elementary school students, the authors randomly selected 14 college students' worlds and 10 elementary school students' worlds. The authors renamed the worlds using numbers to make sure, there was no way to tell which student group a particular world came from. The researchers then graded the worlds on a 100-point scale using very simple criteria. The criteria were taken directly from the problem statement. Students received roughly half credit for including all of the objects in the world. For each missing object, points were deducted from the total score. Each behavior was assigned about 5 points. For each missing behavior, the points were deducted from the total score.

Evaluation

During the experience, the aim of the study was to motivate K-12 students and increase their interest about computing technology. This was an opportunity for a realistic introduction to programming which was of the same level of difficulty given to introductory engineering students. With both groups, all the students were being introduced to programming for the first time with great success. The results of the study indicated that many students from both Engineering 1110 and the 5th grade class were able to reach the same level of proficiency with the exercise and received 100% as their grade. The average grade of the ENGR 1110 scores was 92% and the average of the Cary Woods students that had time to complete the activity was 87.5%. The researchers attribute this difference in average in some cases to the difference in class room setting and time scheduling, as some of the 5th grade students did not have sufficient time to complete the exercise which was limited by their allotted time during class. As a result some of the 5th grade students' grades were lower, but for the group that had ample time (i.e. students that attended all planned sessions) many completed with great proficiency. The authors feel that this result is very impressive that nine year old students were able to fulfill the requirements of a basic programming assignment with near equal proficiency as 18 and 19 year old college freshman. In prior semesters working with elementary students, the authors have had opportunities to work with students from all performance levels and we have even worked with students labeled as under achievers and when you set high expectation they will often reach beyond our expectations and their own expectations to give the motivations to improve in all course work.

This work gives us a vision for developing and exposing more K-12 students to technology at younger ages and K-12 faculty to more educational software. The software application that the researchers utilized for this study is intended for high school students, but for this study the hypothesis was that elementary students can make great strides at programming give the proper support. With this tool children did not have to worry about the cognitive baggage of having to learn an entire programming language and to remember all the syntax which is normally required of most traditional programming languages. This study showed that utilizing tools that support visual direct manipulation programming are a mechanism to provide students the tools to express their imaginations through simulations.

Acknowledgments

This material is based in part upon work supported by the National Science Foundation under Grant Numbers CNS-

0837580 and CNS-0540523. Any opinions, findings, and conclusions or recommendations expressed in this material are those of the author(s) and do not necessarily reflect the views of the National Science Foundation.

References

[1]. **Brightman, J. (2006).** U.S. Video Game Sales Up 15.5% in April. *GameDaily Biz*. (May 16, 2006). Retrieved October 8, 2007 from http://biz.gamedaily.com/industry/feature/?Id=12700&rp=49.

[2]. **Bomia, L., Beluzo, L., Demeester, D., Elander, K., Johnson, M., & Sheldon, B. (1997).** T*he impact of teaching strategies on intrinsic motivation.* Champaign, IL: ERIC Clearinghouse on Elementary and Early Childhood Education. (ERIC Document Reproduction Service No. ED 418 925).

[3]. **Bates, B. (2004).** *Game Design* (2nd Ed.). Boston: Thompson Course Technology.

[4]. **Claypool, K. and Claypool, M. (2005).** Teaching software engineering through game design. In *Proceedings of the 10th Annual SIGCSE Conference on innovation and Technology in Computer Science Education* (Caparica, Portugal, June 27-29, 2005, ITiCSE '05) ACM Press, New York, NY, 123-127.

[5]. **Coleman, R., Krembs, M., Labouseur, A., and Weir, J. (2005).** Game design & programming concentration within the computer science curriculum. *In Proceedings of the 36th SIGCSE Technical Symposium on Computer Science Education* (St. Louis, Missouri, USA, February 23-27, 2005). SIGCSE '05. ACM Press, New York, NY, 545-550.

[6]. **Fullerton, T., Swain, C. & Hoffman, S. (2004).** *Game Design Workshop: Designing, Prototyping, and Playtesting Games.* CMP Books.

[7]. **Geary, P. (2006).** Every child a reader: a national imperative. Reading Improvement: *Magazine for teachers of reading and language arts.* Retrieved September 18, 2008 from http://goliath.ecnext.com/coms2/summary_0199-6227739_ITM. Publication Date Dec. 22, 2006.

[8]. **Gosha, K., *Eugene, W., Gilbert, J., *Hamilton, C. *McClendon, J., *Rogers, G and Seals, C. (2008).** Financial Responsibility Learned Playing Street Legal Customs, *Georgia Tech Graduate Technical Symposium*.

[9]. **Guensburg, C. (2006).** Why Johnny (still) can't read. *Edutopia,* 2(l), 35-36.

[10]. **Jones, S. (2003).** *Let the games begin: gaming technology and entertainment among college students.* Retrieved July 10, 2003, from http://www.pewinternet.org/Pdfs/PIP_College_Gaming_Reporta.pdf.

[11]. **Papert, S. (1980).** Mindstorms: *Children, Computers, and Powerful Ideas,* 1980.

[12]. **Rabin, S. (2005).** *Introduction to Game Development (Game Development Series).* Charles River Media; 1st Edition, June 2005.

[13]. **Robertson, J. and Good, J. (2005).** Story creation in virtual game worlds. *Commun. ACM* 48, 1 (Jan. 2005), 61-65. DOI= http://doi.acm.org/10.1145/1039539.1039571.

[14]. **Seals, C., Hundley, J., & Strange, L. (2008).** The Gaming Approach to Creative Educational Technology. *EISTA 2008 International Conference on Education and Information Systems, Technologies and Applications*, on CD-ROM.

[15]. **Seals, C. D., (2002).** Learning and Reuse in Visual Programming Environments: Simulation Builder for Teachers. *ACM OOPSLA 2002,* November 2002, pp. 120-121.

[16]. **Terdiman, D.(2008)** Video game sales keep bucking recession trend. *CNET News: Gaming and Culture, Geek Gestalt with Daniel Terdiman*. Retrieved December 16, 2008. Publication Date December 11, 2008 from http://news.cnet.com/8301-13772_3-10121667-52.html.

[17. **Tsai, M., Huang, C., and Zeng, J. (2006).** Game programming courses for non programmers. *In Proceedings of the 2006 International Conference on Game Research and Development (Perth, Australia, December 04-06, 2006). ACM International Conference Proceeding Series,* Vol. 223. Murdoch University, Murdoch University, Australia, pp. 219-223.

[18]. **Vegso, J. (2006).** Drop in CS Bachelor's Degree Production, *Computer Research News,* Vol. 18, No. 2. Retrieved July 2008 from http://www.cra.org/CRN/articles/

march06/vegso.html. March 2006.

[19]. Williams, A., Rouse, K., Seals, C. & Gilbert, J. (2009). Enhancing Reading Literacy in Elementary Children Using Programming for Scientific Simulations. *International Journal of E-Learning. 8(1), 57-69.*

ABOUT THE AUTHORS

Cheryl D. Seals, Ph. D. is an Assistant Professor in the Computer Science and Software Engineering Department at Auburn University. She conducts research in Human Computer Interaction with an emphasis on visual programming of educational simulations with end user programming, intelligent agent, usability evaluation, computer supported collaborative work, and minimalism. She is additionally involved in software engineering projects.

Yolanda McMillian, M.S. is a Ph.D. Candidate in the Computer Science and Software Engineering Department at Auburn University. She performs research with the Human Centered Computing Lab, under the direction of Dr. Juan E. Gilbert. Her research focus is with Spoken Language Systems.

Kenneth Rouse, M.S. is a Ph.D. Candidate in the Computer Science and Software Engineering Department at Auburn University, where he is the coordinator/instructor of Intro to Programming for Engineering Students. His area of research is in Voice Biometrics and his specific area of interest is Speaker Classification. Along with this research he is a member of the Human Centered Computing Lab, under the direction of Dr. Juan Gilbert where he is participating in other voice related projects.

Dr. Ravikant Agarwal is an Assistant Professor in the Department of Computer Science at Hope College. His main research areas include software engineering, usability study, modeling and simulation of software process, agent directed simulations and project simulations.

Andrea Williams Johnson M.S. recently received her Master's degree from Auburn University. Before attending Auburn, she graduated from Spelman College with honors receiving a Bachelor's degree in Computer Science. Her research is in the area of human computer interaction with emphasis in usability studies.

Dr. Juan E. Gilbert is the T-SYS Distinguished Associate Professor in the Computer Science and Software Engineering Department and a Fellow in the Center for Governmental Services at Auburn University where he directs the Human-Centered Computing (HCC) Lab. He is also a National Associate of the National Research Council of the National Academies, an ACM Distinguished Speaker and a Senior Member of the IEEE Computer Society.

Dr. Richard Chapman is an Associate Professor at Auburn University with research in the area of VLSI CAD tool design; formal methods; high level synthesis; history of computing; wireless security. He is a major professor for the "Better Tools and Better Interfaces for Hardware-Software Codesign" and "Efficient, Formally Verified Tools for High Level Synthesis and Hardware-Software Codesign" research projects. His research also includes Ubiquitous Computing with small wireless networked gadgets and Unmanned Aerial Vehicles. This research aims at replacing the cables with a wireless network. Funded by US Army Unmanned Systems Initiative. This project attempts to deliver information by ad hoc wireless networks to the users in the field without routing the information through the remote controller.