INTRODUCING PROGRAMMING TO MIDDLE AND HIGH SCHOOLS USING GAME-BASED APPROACH

Emmanuel Udoh

Indiana University – Purdue University, Fort Wayne, Indiana; Email: udohe@ipfw.edu

1. INTRODUCTION

Abstract: In response to the declining enrollment in undergraduate computer science (CS) program, and as an attempt to make CS major more popular to the middle and high school audience, a summer program – SPICE, was established at the IPFW Computer Science Department. The program supplements the K-12 technology curriculum by generating interest in computer science early as an attractive career field. SPICE introduces Java programming language using fun-filled game strategy. Games have a number of desirable pedagogic features, which are easily understood by a generation of students exposed to Nintendo and Sony PlayStation. Participants easily grasp the project's objectives and therefore devote more time to the creative aspects of visual design and strategy. The program is currently at its fourth year and runs for two weeks every summer. This paper documents the design and implementation of the program. It is an essential step that could help stem the tide of declining enrollment in computer science, since it targets middle schools, where most students begin to make decision about career. However, programming has to be introduced early into the K-12 curriculum just like mathematics, in order to boost enrollment and underline the importance of computer science in the emerging information age.

Computer technology plays a crucial role in our lives today, as can be noticed in all fields of business, science, medicine, engineering, research and entertainment. Ironically, there has been declining enrollment in computer science program, especially by women and minorities (Vegso, 2005). Some of the enrolled students even lack adequate preparation for CS program. This has led to a persistently low retention rate in introductory programming classes and discouraged many from continuing in the program. Surveys indicate that many middle and high schools do not have any programming curriculum except basic introduction to Microsoft office tools (Vegso, 2005). This tallies with our observation in the project area, and probably one of the reasons for the ill-preparation of first year CS students. Most experts agree that the best way to address these issues is to strengthen and supplement the K-12 technology curriculum (Sabin, et al., 2005).

Another important issue about the retention of first year CS students is the use of object-oriented languages, e.g. Java, as the introductory programming language. This issue generates intense discussion in CS academic community. There is the school of thought that feels that an object-oriented language makes programming less transparent to

beginners, and eventually leads to frustration or dropout of some first year students from CS program. Our own experience with the first year students regarding Java programming confirms these observations. Java, like other object-oriented programming languages, is however a language of choice in the academia and industry, and has to be mastered by CS students. This situation motivated our decision to set up for the neighboring middle and high schools, a summer program in intensive computer education – SPICE. The thinking is that a one year exposure to Java programming language for these students will give them enough time to master the Java stuff before arriving at the campus for CS career. This will make the first year classes less intimidating, since they are merely consolidating their Java knowledge, and will have ample time to excel in other courses. Moreover, we confined the SPICE program to programming, since programming skill is necessary in several other CS courses. Thus, SPICE departs from other summer CS program that covers multiple areas of CS. The main objective of SPICE is to simplify Java programming language and expose the vast choices of rewarding opportunities in CS to middle and high school students.

The remainder of this paper is organized as follows. Section 2 introduces the modality of the SPICE program. Section 3 discusses the game-based approach that could make programming an interesting endeavor, while section 4 concludes with lessons learned.

2. THE SPICE PROGRAM

SPICE is an intensive, fun-filled, two-week summer program in modern software development offered by the IPFW Computer Science Department for middle and high schools. It provides the opportunity to sharpen basic skills in Java, one of the most widely used languages for Internet and modern application programming. Students meet other students with strong programming interests, prepare for the AP CEEB examination, and perhaps explore a computer-based career. SPICE takes cognizance of the recommendations made by the ACM/IEEE report on high school computer science curriculum (http://www.acm.org/education/hscur/appendixab.html#a).

The program is currently at its fourth year. In the first year, we had 6 participants, but currently have 20 students. This increase is an encouraging development, although we target 40 students. The program is funded with support from the department and area companies. Our students are from Fort Wayne schools. Activities are only on weekdays (Monday to Friday), and students arrive daily to campus and leave at noon following the schedule as in Table 1.

Table 1: SPICE Daily Schedule

Time	***Activity***
8:00 a.m. – 9:00 a.m.	Class session
9:00 a.m. – 9:15 a.m.	Snack Break
9:15 a.m. – 10:15 a.m.	Class Session

10:15 a.m. – 10:30 a.m.	Snack Break
10:30 a.m. – 12 Noon	Lab Session

Students participate in lectures, laboratory demonstrations and hands-on computer science projects. We offer two classes using a single Java textbook (Liang, 2005) as follows:

SPICE I. The content of SPICE I is truly basic Java, and intended for absolute beginners. The topics are introduction to Java and JBuilder, primitive data types and operations, control statements, methods, arrays, strings, objects and classes.

SPICE II. The content of SPICE II is fairly advanced, and designed for students with prior Java background. The topics include GUI programming, event driven programming, user interfaces and abstract classes, inheritance and polymorphism, exceptions, input, output, and threading.

Students are encouraged to work in teams, and develop Java application that implements a computer game. They are also exposed to various career opportunities by different faculty members. At the end of the program, attendance certificates and several gifts are awarded. The participants are further encouraged to participate in Advanced Placement (AP) examinations and other programming competitions. We also encourage students to evaluate the program and computing in general through series of pre- and post-program surveys. At the end of the program, students take a trip to an area software company where they can experience computing at work. Details of the SPICE program are available at http://users.ipfw.edu/udohe/spice.htm.

In the past three years, 44 students participated in SPICE 1. SPICE II began in 2005 with only 5 participants. Due to the participation of 7-12 graders in SPICE, it will take some time to obtain accurate statistics on enrollment in colleges. However, based on the post-program surveys, about 50 to 60 percent of the participants indicated interest in computer science. The enrollment figures we obtained from IPFW and the parents of SPICE participants, show that about 10 students are now in colleges, with about 5 participants enrolling in computer science and engineering.

3. GAME-BASED PROGRAMMING APPROACH

Pedagogically, SPICE adopts the "fundamentals-first and objects-late" strategy, proceeding at a steady pace through all the necessary and important basic concepts, then moving to object-oriented approach to build interesting GUI applications. At every ACM-SIGCSE conference prior to 2005, the objects-early approach was trumpeted and the voice for the fundamentals-first approach was muted (Liang, 2005). This has been changed when some former proponents of "objects early" began to air their frustrations and declared that "objects early failed", possibly one of the reasons for the frustration of the first year CS students or the high dropout rate from CS program. Based on this

philosophy, we concentrate on Java fundamentals in SPICE I, while SPICE II focuses on objects and GUI.

We adopted game strategy to introduce Java programming language in SPICE I and II. Simple games are introduced in SPICE I with GUI-based ones in SPICE II. Games have a number of desirable pedagogic features, which are easily understood by a generation of students exposed to Nintendo and Sony PlayStation (Pivec, et al., 2003,: Pollock, et al., 2004) Participants easily grasp the project's objectives and therefore devote more time to the creative aspects of visual design and strategy (Huang, 2001). At the beginning of each day session, we introduce a game for the day. The Java material that will be covered each day should be sufficient to program the game in the lab session. In this vein, we introduce games that can be programmed with few lines of code. It ensures the participants understand the code and can discern how the game application is developed. We noticed a lot of excitement among the participants as we strategize and attempt to design the game. During this design phase, we introduce some Java statements that can solve the problem, and then expatiate on them during lecture session.

In SPICE I, students build both applet and stand-alone applications, but develop low level games such as Name-Guesser. This is the level of game that is standard for SPICE 1. The Name-Guesser game requests a user to input a first name. The program then prompts for the last name. Thereafter, the game proceeds to manipulate the names and output some results. This is a simple game program that entails basic functions available in many applications, and therefore useful in the educational process (Werner, et al., 2005). We also immerse SPICE 1 participants in graphical programming that develops simple graphics software. We noticed that designing simple graphics is a thriller for the participants, and helped enkindled interest in the program. On a lighter note, a participant in SPICE 1, who had no prior background in Java before enrolling in SPICE, won the Indiana Java Programming contest in 2004.

In SPICE II, students build more sophisticated game programs such as table-tennis, slot machines and strategy games. We build upon the success achieved in SPICE I. This tallies with our enrollment, since SPICE II participants are former SPICE I students. We focus on games that allow the students to also draw more graphics such as table-tennis. This game offers the students several aspects of programming as well as tasks to solve. The projects help students to understand data structures and algorithms as well as manage several lines of code. In the process of designing projects in teams, students learn to discuss ideas and coordinate software systems.

5. CONCLUSION

The combination of a small group in lab setting, collaborative group projects on fun-games and enthusiastic faculty and students, were shown by surveys to have a positive impact on the participants' perception and interest in computer science. We strived to make the projects doable and the Java language discernible. The focus on games provided a way to make programming concepts clearer and interesting. The enthusiasm generated

by academic success can provide the impetus to pursue a career in computer science. The SPICE program helped to advertise computer science to the community.

One pointed lesson learned is the stunning absence of a computer science/programming curriculum in many area schools. Curriculum change is essential for greater awareness of computer science in the secondary schools. Without significant influx of computing expertise and resources, the situation may not improve. Lastly, programming has to be introduced early into the K-12 curriculum just like mathematics, in order to boost enrollment and underline the importance of computer science in the emerging information age.

REFERENCES

Huang, T. (2001). Strategy Game Programming Projects. *Journal of Computing Sciences in Colleges*, **16**, (4), 205-213.

Liang, D. Y. (2005). *Introduction to Java programming*. Prentice Hall, Upper Saddle River.

Pivec M., Dziabenko O. and Scinnerl I. (2003). Aspects of game-based learning. *International Conference on Knowledge Management*, Graz, Austria, 216-225 pp.

Pollock L., McCoy K., Carberry S., Hundgopal N. and You X. (2004). Increasing high school girls' self confidence and awareness of CS through a positive summer experience. *35th SIGCSE technical symposium on compute r science education*, Norfolk, Virginia, 185-189 pp

Sabin M., Higgs B., Riabov V. and Moreira A. (2005). Designing and running a pre-college computing course. *Journal of Computing Sciences in Colleges*, **20**, (5), 176 -187

Vegso J. (2005). Interest in CS as a major drops among incoming freshmen. *Computing Research News*, **17**, (3), 6-1.

Werner L., Campe S. and Denner J. (2005). Lifelong student engagement in IT: Middle school girls + games programming = information technology fluency. *6th Conference on information technology education SIGITE*, Newark NJ, 301-305