App Inventor Summer Camps Increase Student Interest in Programming

Barbara J Ericson
College of Computing
Georgia Institute of Technology
801 Atlantic Dr
Atlanta GA 30332

ericson@cc.gatech.edu

James K Greenlee Jr
Business and Computer Science Dept
St Pius X Catholic High School
2974 Johnson Rd NE
Atlanta GA 30345

jgreenlee@spx.org

Abstract

The Institute for Computing Education (ICE) is a partnership between the Georgia Department of Education and the College of Computing at Georgia Tech. ICE challenges the stereotypes surrounding modern computing by offering summer programming camps aimed at high school students. These camps use App Inventor to introduce participants to programming concepts and computational thinking, in a way that is both creative and engaging. Participants work individually or in small groups to create Android apps of their own design, utilizing a variety of media and programming techniques. Typical projects involve image manipulation, games, communication with external devices, or use of sensors. Pre- and post- surveys are conducted to measure how participants' attitudes toward computing are changed by the experience. This talk will explore some results and conclusions from conducting these camps, and will offer some suggestions and strategies for increasing interest in computing among high school students.

1. Introduction

As part of its outreach efforts, the Institute for Computing Education (ICE) at Georgia Tech offers annual summer programming camps for local students in grades 9 through 12. The purpose of these camps is to expose students to computing, increase their knowledge and understanding of programming and programming concepts, and improve students' impression of computing as a professional activity. In particular, a major goal of the programming camps has been to introduce computing in a way that directly addresses and contradicts negative stereotypes about programmers and programming. In particular, the primary focus of the camps is to show that programming can be engaging and social, as well as providing an outlet for creative expression. A secondary focus is to attract young women and underrepresented minorities to the discipline.

One of the more popular camps from the past few summers has been a week-long programming experience where participants are introduced to Android programming with App Inventor. Daily instruction for the camp takes place from 9:00 AM to 3:00 PM, in a

dedicated programming laboratory that is provided by Georgia Tech's College of Computing. Each individual camper is provided a laptop and Android device to use for the entire week. Instruction is conducted by an experienced high school or college teacher who uses App Inventor regularly and is knowledgeable about programming. The teacher is assisted by several counselors; typically these are undergraduate Computer Science students from Georgia Tech. Female and minority counselors are common, and the usual student-to-staff ratio for these camps is no more than 6:1.

2. Demographics and Curriculum

A pre-survey is administered to determine overall group demographics, as well as perceptions about computing. A typical breakdown of students by ethnicity, age, and gender is shown in Tables 1 through 3.

Ethnicity/Race	
Asian American	33%
African American	39%
Hispanic	0%
Native American	0%
Caucasian	28%
Two or more of the above	0%

Table 1

Age	
13 years	11%
14 years	56%
15 years	22%
16 years	6%
17 years	6%

Table 2

Gender	
Female	28%
Male	72%

Table 3

Campers receive approximately 5 contact hours of instruction each day. The first two days alternate between instructor-led hands-on demonstrations that create basic functional apps, followed by periods of self-directed effort where participants are allowed to either modify existing features of these apps, or add more capabilities of their own choosing. The instructor often solicits suggestions from the students on how to improve the app, and students are encouraged to come up with their own ideas as well. No assumptions are made about participants' previous experience with programming, and they are free to add as many new features as they can complete in the available time, dependent only on their own level of interest, confidence, or skill. Participants are also encouraged to work in small groups to solve problems collaboratively. At the end of each lab period, the instructor conducts a group discussion where all campers are given an opportunity to reflect on various solutions to the problems that they encountered.

The main goal of the first two days of the camp is to introduce the campers to as broad a range of apps as possible, including graphics applications, games, and apps that utilize sensors or communicate with other devices. A number of programming concepts and techniques are covered, including the purpose and use of variables, conditionals, loops, lists, and procedures. A thorough discussion of user interfaces and event-driven programming is also included. Participants are provided with full Internet access via their laptops, and are free to explore various on-line App Inventor resources to get additional ideas for their projects. After working on several types of apps, the focus shifts to general program development and design, and a brief discussion of algorithms and the software life cycle is conducted.

At mid-week, campers are asked to start designing an original app that they can complete by the end of the week. No constraints are placed on the kind of app that the participants undertake; they may start a brand new design or continue working on one of the previously demonstrated apps if they wish to explore further. The remaining two days of the camp are spent developing these original apps, and on the afternoon of the final day, the campers present their apps to each other and their parents. Post-surveys are conducted to determine how the experience has changed the participants' perceptions of computing.

3. Summary

The post-survey questions attempt to capture the campers' confidence and enjoyment of app programming. Table 4 briefly summarizes some results of the surveys. The scale is 1 through 5, where 1 indicates "Strongly Disagree" and 5 indicates "Strongly Agree"

Assertion	**Pre**	**Post**
Computers are fun	4.48	4.61
Programming is hard	2.90	2.50
I will be able to get a good job if I learn how to use a computer	3.90	4.28
I will stick with a computing problem until I have a solution	3.80	3.94
Girls can do just as well as boys in computing	4.52	4.72
I can be creative with computing	3.95	4.28

Table 4

In addition to the ranked responses, students are given an opportunity to provide free responses. Some sample written responses from a recent survey are shown in Table 5.

The best thing about this event was:
App Inventor was very helpful to get the feel of programming. It was fun.
Being able to make your own app however you want to.
Creating apps from scratch without having knowledge about programming
Meeting /working with other people and making an app from scratch
Making our own apps and the robot battles.

Table 5

4. Conclusions

Based on student responses from the surveys, there are several keys to increasing students' interest in programming:

- An easy-to-use programming environment with rapid prototyping capability allows students to develop and test sophisticated apps quickly.
- Students prefer having creative control over how their apps function, which gives them a sense of ownership in the final product.
- Collaboration and reflection allows students to gain additional insights into how to solve challenging problems.
- Students particularly enjoy tactile exercises that utilize sensors or external devices.