

Bridging the Gap: Storytelling Alice as a Precursor to Python

Computer Systems Lab 2009-2010

Abstract

The goal of my project is to determine whether or not Storytelling Alice is successful in preparing elementary school students for a career in computer programming. Essentially, I am testing to see how possible it is to bridge a gap between a visual and kid-friendly interface like Storytelling Alice and a non-visual, syntactically difficult interface like Python. If students are able to grasp concepts in Python after a foundational education in Storytelling Alice, it will show that this is a successful means of introducing students to computer science concepts.

Background and Introduction

Storytelling Alice was created by a student at Carnegie Mellon University named Caitlin Kelleher. Last year, two girls did a research project at Cardinal Forest Elementary but focused on Scratch programming. This year, Mr. Allard decided to add Storytelling Alice as another language for students to study and I have been put in charge of the mentoring program for this language. This is the first year we will be testing Storytelling Alice at Cardinal Forest Elementary. However, it is not the first time Storytelling Alice has been used to teach young children to program.

Kelleher was interested in developing Storytelling Alice for many reasons. In her PhD dissertation, she recognized how important diversity is in the field of computer science. She felt storytelling Alice could help attract middle school girls to computer science and geared her language toward that group. Kelleher mentioned a study that found that boys' and girls' ideal technologies were significantly different. Because of this, having a more equal ratio of girls and boys in the field could drastically change the course of technological development.

One article I read about Alice programming dealt with girls and their ability to use Storytelling Alice to learn programming. The article stated that girls have the same ability as boys do to program. This was encouraging for my project because not only am I a girl interested in programming as a career, I hope to spark an interest in programming in the girls I mentor. These researchers also stated that there were many reasons why girls were less likely to choose to pursue a career in programming. Not only are there social norms that encourage boys to program and not girls, at the middle school level, girls confidence in their abilities relating to math and science becomes deflated. This is actually a very promising fact for our study. Because we are working with Elementary School students, not only are we able to encourage programming for girls before they reach middle school level and lose confidence in science-related subjects, we are also working with them at an age when they are unlikely to know about social norms in programming. Hopefully, Alices easily understood set up will help give girls confidence and Alices storytelling nature will be attractive to girls.

Another article I read simply studied Storytelling Alice and game making. Both articles mentioned that Alice is very successful in teaching algorithmic thinking and basic programming language and syntax to children. A third article I read discussed the difference between teaching programming with Storytelling Alice and other gaming centered programming languages. Because the gaming community is predominately male, a programming language with a gaming nature would be more attractive to males than females. Storytelling, however, is attractive to both males and females and, if taught at an early age, could help equalize the number of boys and girls interested in programming.

There are other benefits to using Storytelling Alice. Concepts that are hard for first-time programmers to understand like variables, parameters, and loops have to be introduced through assignments in other languages. In Alice, however, they are incorporated into the foundations of the language. Kelleher noted that many problems that beginning programmers encounter, like invisible state and syntax errors are not an issue for users of Storytelling Alice. She also stated that in her work with college level students she found that the retention rate almost doubled after a short introductory Alice class.

Name: Paula

Write about what you learned in this lesson! Hint: What did this teach you about programming? Also, what was your favorite part?

My favorite part is when I win the game. It taught me how to make a game and how to use different loops. I think it was very cool. I was soooooo much fun how to play.

Development and Discussion

Each Tuesday, I work with a class of fourth graders. There are only three boys in the class (the majority of the students are female). We have completed four projects together in class so far this year. The first was an introduction to the Alice interface. It taught the kids how to create a world in Alice and allowed them to get comfortable with the new environment. The second lesson focused on creating methods and the third on creating stories in Alice. Each lesson targeted to teach certain programming concepts and over the year so far the students have learned about different control statements, variables and methods. The most recent lesson was a program called "Guess that Number". It centered less on the visual aspect of Alice and more on the coding. This lesson will be the first we complete using Python so that the concepts are the same but the language and environment are the only differences. I also have changed my methods of assessment over the course of this year. I have given three quizzes and allowed the kids to write a journal and complete a worksheet. See results and conclusions for the data acquired. I found that many students are threatened by the concept of the quiz and I am able to get more accurate data through other means of assessment.

Results and Conclusions

I have collected five sets of data from different assessment methods throughout the year so far. There were five perfect papers (out of sixteen) on the first quiz. This was a promising start to the year. However, the second quiz asked the students to code on their own without prompting and without the help of the drag-and-drop interface. Many were too intimidated by this concept to do well on the quiz and therefore the results were not good. The average score was about a 48%. I do not believe that this was because of a lack of understanding, because the students were receptive to the information in class. Therefore, I rethought my process of classroom assessment. After the fourth lesson, I decided to give the kids a journal assignment where they could write about what they thought about the lesson and what it taught them. One example is shown in the figure above. I found that I was able to see how the children felt and how confident they were with the coding without the prompting or intimidation that comes with questioning. However, the downside to this method is the inability to look at this data in terms of numbers to assess the students' understanding as a whole. We then did a comments worksheet in class that served as both a learning and an assessment tool. The results of this were more promising than the previous quiz. The average score was an 88%. The final quiz was an accumulative exam and all students did very well.