Intelligent Tutoring Systems and Alice

Yoon Jae Nam[1], Stephen Cooper[2], Si Luo[3]
[1]Department of Computer Science, Stanford University [2]Department of Computer Science, Stanford University [3]Department of Computer Science, Purdue University

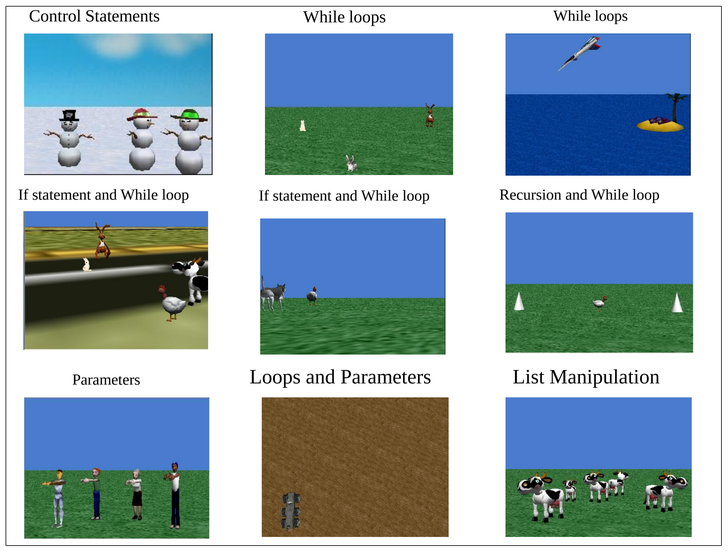
INTRODUCTION

An Intelligent Tutoring System (ITS) is a computer system that helps students learn without human interventions. There are many intelligent tutoring systems currently in practice, but few of them are related to computer science education.

In this research, we began to develop an intelligent tutoring system in Alice, a software program designed to teach introductory computer programming.

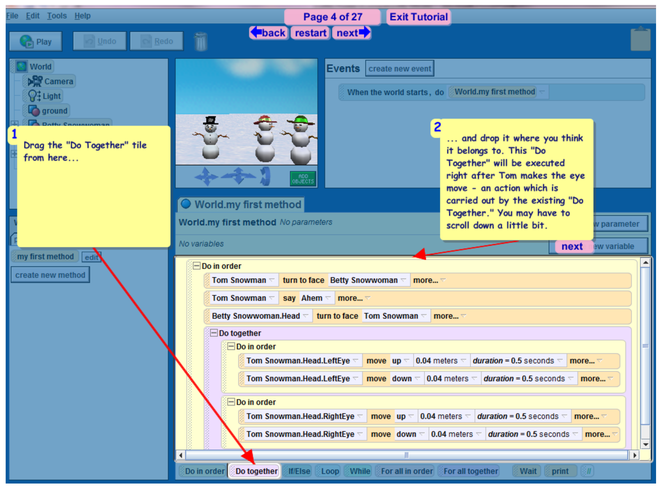
Our ITS will consist of two parts: one part that detects student struggles, and another part that offers corrective action for the student.

This research focuses on offering students corrective actions -- using stencils where students learn by making mistakes, which are intentionally designed and incorporated into the tutorial system.



(Not a complete list of tutorials)

Stencils



Stencils is an interaction technique that can help students follow a tutorial correctly. Stencils let the students drag or modify only certain parts of the screen in each step. This way, the ITS can direct the students' attention to the current objects of interest and prevent them from making errors by modifying other parts of the screen.

When a student makes a mistake, the stencil detects the error, lets the student know that he/she made a mistake, and makes the student go back to the previous stencil.

Teaching Alice

We created a total of thirteen tutorials that teach basic programming concepts with which novices often have difficulties, including parameter passing, nesting of control statements, while loops, and list manipulation.

For each tutorial, an activity was designed to measure the extent to which the student learned the concept presented in the tutorial.

For each tutorial, we created a storyline and stated the goal of the tutorial, in order to make the tutorials engaging.

Approach

Although the tutorials prevent the students from going on to the next step if the students do not follow the instructions carefully, the tutorials do let the students make mistakes so that the students can learn from them. In almost every tutorial, we introduced many errors that students often make. By having the student go through a mistake and see what happens for themselves, we made our tutorials educational and interactive.

Results

We tested our tutorials and activities on nine undergraduates with little or no prior knowledge in computer science. All of them were able to complete the tutorial with no or minimal help. On average, each student spent 20 minutes on a tutorial and 11 minutes on an activity.

In 68% of the trials, the participant answered that the instructions in the tutorial were very clear. In 73% of the trials, the participant answered that they learned a lot from the tutorial.

In 60% of the trials, the participant answered that the tutorial was easy to walk through. In 30% of the trials, the participant answered that the tutorial was of medium difficulty.

With little or minimal help, all the students were able to finish the activities.

Future Work

Building the detection system
- A logging system that can detect off-task behaviors has been prototyped. It needs to be made more robust and then logs can be analyzed to detect when students are struggling on various concepts.

Testing the resulting ITS

This work is partially supported by NSF award DUE 1021975 - Adding an Intelligent Tutoring System to Alice. Any opinions, findings, and conclusions or recommendations expressed in this material are those of the authors and do not necessarily reflect the views of the National Science Foundation.