

Successful Approaches to Teaching Introductory Computer Science Courses with Python

Special Session

David Ranum
Bradley Miller
Computer Science
Luther College
Decorah, IA
ranum@luther.edu
bmiller@luther.edu

John Zelle
Computer Science
Wartburg College
Waverly, IA
john.zelle@wartburg.edu

Mark Guzdial
Computer Science
Georgia Institute of
Technology
Atlanta, GA
guzdial@cc.gatech.edu

Categories and Subject Descriptors

K.3.2 [Computer and Information Science Education]:
Computer Science Education

General Terms

Languages

Keywords

CS1/CS2, Curriculum, Python

1. TOPIC

Teaching the Introductory Computer Science Sequence Using Python

2. NATURE OF THE PROPOSAL

Special Session: series of presentations related to a specific theme.

3. BRIEF OVERVIEW

This session will focus on ways that Python is being used to successfully facilitate introductory computer science courses. After a brief introduction, we will present three different models for CS1 and CS2 using Python. Attendees will then participate in a discussion/question-answer session considering the advantages and challenges of using Python in the introductory courses. The presenters will focus on common issues, both positive and negative, that have arisen from the inclusion of Python in the introductory computer science curriculum as well as the impact that this can have on the entire computer science curriculum.

4. CONTACT

David Ranum
Department of Computer Science
Luther College
Decorah, Iowa 52101
563-387-1410 (phone)
563-387-1088 (fax)
ranum@luther.edu

5. OVERALL OBJECTIVE

We believe that introductory computer science students can benefit from using Python as the primary programming language in their introductory courses. As Python gains in popularity, different models are emerging for successful implementation of CS1 and CS2 courses. The objective of this session is to show and discuss a variety of course and sequence models that use Python.

6. OUTLINE OF THE SESSION

6.1 David Ranum: Introduction and Statement of the Problem

As students progress through the introductory computer science sequence, we want them to focus on aspects of problem solving, algorithm development, and algorithm understanding. Unfortunately, many modern programming languages require that students jump into more advanced programming concepts at a point that is too soon in their development. This sets them up for possible failure, not because of the computer science but because of the language vehicle being used. We believe that Python is an exceptional language for teaching introductory computer science students. Python has many advantages that can help to minimize the failure scenario noted above. In this session we will explore these advantages and give specific examples of introductory courses and course sequences that have been successfully implemented.

6.2 Mark Guzdial: Python for CS1 and Non-Majors

We have been using Python in an introductory course for non-CS majors. The course, "Introduction to Media Com-

putation,” has students learn CS1 topics through manipulation of media, such as creating image filters, splicing sounds, generating HTML from databases, and creating videos and animations with digital special effects. Around 300 students a semester take the course, with a very high success rate (around 90% in some terms). We have developed a Java version of the course that we use with high school teachers in a summer workshop, but find that Python students can be doing interesting things sooner because of the less complex syntax of Python.

6.3 John Zelle: Python for CS1

We have been using Python as a first language and as one of the primary languages in our CS curriculum. We teach a two course introductory sequence where the first course is Python only and the second class uses both Python and Java. Both languages are used extensively in our upper level courses. Our experience with Python in the early courses has been almost universally positive. Our students are writing more programs, becoming much more proficient problem solvers, and having more fun in the first class. The transition to Java in CS2 is smooth, and our students are at least as proficient as they were when both courses were taught with the same language (C++ or Java). In upper-level classes, significant experience with a scripting language is a big advantage when students are doing experimental programming to reinforce conceptual material. We’ve actually seen that some of our upper-level projects now seem too easy for students proficient with Python.

6.4 Bradley Miller: Python for CS1 and CS2

We have implemented a three course introductory sequence using Python for CS1 and CS2. Our goals for the first course in computer science are to introduce students to basic problem solving approaches. Programming is presented as a notational means for expressing solutions to these problems. As students progress from the first course into the second we begin to focus on classic algorithms and data structures that recur in the solution to many problems. Even though these ideas are more advanced, we still assume that the students are beginners. They may be struggling with some of the ideas and skills from their first course and yet they are ready to further explore the discipline. Our goals for the second course are similar to the first. We want to continue the exposure to algorithms, data structures, and problem solving. We want students to understand how to build analysis frameworks to compare and contrast solutions. Most importantly, instead of learning a new programming language or a new software development technique, we want students to continue to learn computer science. Our experience with Python has been extremely positive with a very high degree of success for student projects. We utilize Java in our third semester introductory course. We believe that Java is an excellent programming language for industrial strength software development and in this course we emphasize many modern programming practices that students need to understand to be successful in upper level courses.

6.5 Discussion

Following the individual presentations, a set of common questions will be addressed by each participant. These questions will serve as motivation for a larger discussion involving all of the attendees.

7. EXPECTATIONS

We believe that many of the SIGCSE attendees continue to struggle with finding ways to address the complexities of their introductory computer science courses. This session shows that Python is a viable language for teaching the introductory sequence. It can be used to facilitate a number of different approaches. It is important to continue consideration of strategies that maximize the success of these students. Audience members will leave this session with an appreciation for the benefits of Python as an introductory language and an understanding of the possible challenges that can arise. In addition we hope that others who are using Python in their computer science courses will be able to share their experiences. Finally, a summary website (<http://www.cs.luther.edu/Python>) will be established to serve as a repository for ideas, information, and recommendations pertaining to the use of Python in the computer science curriculum.