Using Competitive Programming to Teach Computer Science

Initial Report: September 14, 2007

Scott Wallace
School of Engineering and Computer Science
Washington State University Vancouver
Vancouver, WA 98686
360 546-9112
wallaces@vancouver.wsu.edu

Jason Margolis
College of Education
Washington State University Vancouver
Vancouver, WA 98686
360 546-9091
margolis@vancouver.wsu.edu

Project Overview:

Our project seeks to gain a better understanding of the role that competition plays in the Computer Science classroom. To do this, we are building a set of curricular modules which target traditional computer science teaching objectives within the framework of competitive programming. Each module consists of an assignment that can be evaluated automatically in real-time using a purpose built software infrastructure. By varying the type of feedback students receive, we expect to be able to gain a deeper understanding of how competition and rapid feedback contribute to a student's level of engagement in their coursework.

Project Objectives:

As proposed, the project contains four high-level objectives:

1) Design and implement the Rapid Feedback Software Infrastructure
2) Develop and field test three curricular modules
3) Evaluate the impact of competitive programming curriculum
4) Disseminate teaching resources and resulting knowledge

Each objective is discussed below with respect to assessment criteria set forth in the project proposal. We comment on both the progress made to date and expected near-term accomplishments.

<u>Design and implement the Rapid Feedback Software Infrastructure</u>

The proposed assessment criteria for this objective were: 1) create a design document to specify the requirements of the software infrastructure; 2) implement the infrastructure so as to conform to the design document; and 3) field test the software in at least one course.

Over the past four months, the PIs have identified and documented the requirements for the software, meeting criteria (1). Implementation was started during the summer and is making good progress. Due to the competing obligations of our student programmer, the infrastructure is not yet fully complete. As a result, we have switched our approach to follow a two stage implementation strategy.

The first phase (currently ongoing) is to implement the base functionality required for this semester's use of the rapid-feedback software platform. This phase is in its final stages; the software is stable and mainly requires additional testing. We are on schedule for a phase one release date in the first week of October which will fulfill criteria (2). In addition, the phase one platform will be sufficient to meet (3) by mid December.

The second phase will be to extend the base platform for more generalized use. This effort will begin immediately following the phase one release in October. The main goals will be to support arbitrary programming languages and a large spectrum of curricular modules. We will also make improvements based on our in-class experience with the phase one platform. The phase two effort will likely continue into early 2008.

Develop and Field Test Three Curricular Modules

Proposed assessment criteria for this objective were to develop, implement, document and use the curricular modules in one WSUV course. We have targeted the current senior level *Design and Analysis of Algorithms* course offered by Dr. Wallace. Because the course is required for all Computer Science majors, we will ensure a large cohort of students for our study. In addition, since this is a standard course, other institutions will likely be able to profit from the curriculum we design. Currently, our curricular projects are in the planning stages. Course time has been set aside to introduce students to the Python programming language and three competitive programming projects have been added to the syllabus and are scheduled throughout the upcoming months (October through December). We are well on track for accomplishing the objective by the end of the semester.

Evaluate the impact of competitive programming curriculum

The proposed assessment criteria set forth for this objective were: 1) issue at least three surveys and one focus group to measure the impact of the course software and competitive programming paradigm; and 2) collect enough data to submit a report on the project and its findings to a national conference or journal. The project PIs have developed an initial survey to measure student predisposition to the use of competition in the classroom. This was given to our class during August, and will serve as a baseline for further surveys. The next survey is planned for after the first competitive programming assignment is offered. We are currently awaiting Institutional Review Board approval which will allow us to hold focus group meetings. We expect approval within the next month, and are well on track for meeting the all criteria for this objective within the current semester.

Dissemination

In May, we created a project website which will host all grant related resources. We have also performed a background literature review that has resulted in one paper submitted to the annual ACM SIGCSE Technical Symposium on Computer Science Education. We will continue to disseminate project related resources and results as they are completed.

Spending:

Currently, we have spent roughly half of the total budget (approximately $4700). The bulk of this amount ($4000) has supported the two PIs during the design and initial implementation phases of the project. The remainder has paid for a student programming assistant to help implement the software framework. The remaining funds are expected to be used mainly to support travel and additional student programming assistance.