Program Assessment via a Capstone Project

John Galletly
American University in Bulgaria
Blagoevgrad 2700
Bulgaria
+359 73 888 466
jgalletly@aubg.bg

Dimitar Christozov
American University in Bulgaria
Blagoevgrad 2700
Bulgaria
+359 73 888 443
dgc@aubg.bg

Volin Karagiozov
American University in Bulgaria
Blagoevgrad 2700
Bulgaria
+359 73 888 456
vkaragiozov@aubg.bg

Stoyan Bonev
American University in Bulgaria
Blagoevgrad 2700
Bulgaria
+359 73 888 419
sbonev@aubg.bg

ABSTRACT
This paper describes an approach that has been adopted at the American University in Bulgaria in order to assess the Computer Science degree program for accreditation purposes.

Categories and Subject Descriptors

K.3.2 [**Computers and Education**]: Computer and Information Science Education – *Curriculum, Self-assessment*

General Terms
Management, Measurement, Standardization

Keywords
Program assessment, student learning outcomes, accreditation, rubrics, primary trait analysis, capstone project

1. INTRODUCTION
Accreditation is the primary process for assuring and improving the quality of higher-education institutions. A university or program of study, such as computer science, that has successfully completed an accreditation review is considered to have the required instructional and supporting services to help students to achieve their educational goals. An accredited university means that students can expect that the university or program will live up to its name. It means that a student can be assured that his/her degree has value.

The American University in Bulgaria (AUBG), situated in Blagoevgrad, Bulgaria, offers an undergraduate American-style, liberal arts education. AUBG is subject to two accreditation agencies – one for American accreditation (via the New England Association for Schools and Colleges – NEASC [1]) as AUBG is an American institution; and the other for Bulgarian accreditation (via the National Evaluation and Accreditation Agency – NEAA [2]) as AUBG is situated in Bulgaria and is a Bulgarian institution, also. Earlier papers [3, 4, 5, 6, 7, 8] shared the experience of aspects of teaching computer science at a liberal arts university, including an examination of the accreditation process by these two different agencies, at the institutional and program levels.

BCI'12, September 16–20, 2012, Novi Sad, Serbia.

Local Proceedings also appeared in ISBN 978-86-7031-200-5, Faculty of Sciences, University of Novi Sad.

Accreditation, as a means to assure and improve higher-education quality, uses a set of standards that have been developed by accreditation agencies such NEASC and NEAA [9]. As part of the accreditation process, institutions and programs must show that they meet the standards that require them to provide quality education.

One standard is an assessment of student learning in each program of study, i.e. the expected student learning outcomes (SLOs). Assessment is based on clear statements of what students are expected to gain, achieve, demonstrate, or know by the time they complete their academic program. The program must implement and provide support for the systematic and broad-based assessment of what, and how, students are learning in the academic program. Assessment plays a key role in evaluating student learning, and making improvements to the program [10].

Assessment is not just about examining students; it is much more. The idea is that each program conducts an assessment of student learning each year, and uses the results to improve student learning by making appropriate changes to the program when and where necessary. Basically, each program should

(a) Identify and develop a set of student learning outcomes.

(b) Develop an assessment plan.

(c) Determine an assessment method.

(d) Develop assessment metrics or rubrics.

(e) Collect and analyze assessment data, and draw conclusions about collective student achievement in each outcome.

(f) When necessary, based on the above analysis, propose necessary changes to the program in order to improve student learning for any under-performing outcomes. In other words, close the loop.

Various approaches to this problem have been described in the literature [11]. This paper describes the method selected by the

Computer Science department at AUBG to assess the computer science program for accreditation purposes.

2. PROGRAM ASSESSMENT

2.1 Formulation of Learning Outcomes

The SLOs (or goals) for the computer science program were first developed from the program's mission statement and the (high-level) educational goals of the university.

The following are the key set of outcomes that the Computer Science department thought was necessary for our graduating students:

"The program is designed to enable students meet the following skill or competency-based outcomes and show mastery of computer science knowledge and skills, through a capability to

• Demonstrate an understanding of, and ability to apply, current theories, models, techniques and technologies that provide a basis for problem solving.

• Work as an effective individual and as part of a team to develop and deliver quality software.

• Have the ability to communicate effectively both orally and in writing.

• Be aware of key ethical issues affecting computer science and the responsibilities of computer science professionals.

• Learn new theories, models, techniques and technologies as they emerge and appreciate the necessity of such continuing professional development."

Having developed this set of learning outcomes, it was then necessary to decide how best they could be evaluated. The Computer Science department decided to approach this through the assessment of the computer science senior projects – evaluating senior projects based on the program's goals.

2.2 Assessment Method for Learning Outcomes

A number of assessment techniques have been suggested and used in practice [12]. Assessment methods are classified as being either direct or indirect [13]. Direct methods evaluate what a student has learned. Examples are capstone projects; tests and examinations; and portfolios of the students' work. Indirect methods, on the other hand, gather information through means other than looking at samples of student work. Examples are exit interviews; alumni surveys; and employer surveys.

Any assessment method should reflect the type of learning to be measured. Computer science is a practical discipline with emphasis, at AUBG, on quality software development. This led the Computer Science department to decide that the ideal assessment method is the computer science senior project – an existing compulsory capstone course [14] for graduating computer science students at AUBG. Completing the senior project successfully, broadly demonstrates a student's competencies as a computer professional.

The senior project requires the development of a substantial software package developed by each student, individually, over a semester-long period. As such, it provides evidence of how well our students integrate and apply principles, concepts, and abilities, learnt in preceding computer science courses, into this culminating project.

At the end of the semester, each project is evaluated via a public presentation of the project, along with a demonstration. Additionally, a detailed project report must also be submitted. The project is evaluated by a panel of Computer Science faculty.

2.3 Choice of Assessment Method

After some deliberation, the Computer Science department decided that a suitable method for assessing the projects was Primary Trait Analysis (PTA) [15, 16, 17]. ("Trait" here equates to a performance indicator, i.e. a measurable attribute that defines some learning outcome.) Primary trait analysis, an evaluation tool used extensively in liberal arts institutions, defines a number of specific criteria or traits to be evaluated along with specific measures of performance for each trait. To paraphrase Walvoord and McCarthy [16], the PTA method allows us to take what we are already doing, i.e. scoring the students' capstone projects, and translating that process into an assessment device. Using PTA for student evaluation provides the faculty with clear guidelines for student evaluation, and the students with a clear understanding of performance expectations.

The method makes use of a scoring grid (or matrix) which was developed from the computer science program learning outcomes, and, importantly, provides feedback to the faculty for any future curricular enhancements by indicating performance strengths and weaknesses in the given outcomes, i.e. it allows program assessment and improvement. An example partial grid for one student is shown in the Figure 1 below.

Outcome and Rubric	Score				
Show mastery of computer science knowledge and skills: the project was of sufficient complexity that a mastery of computer science knowledge and skills in a number of areas was necessary in order to develop an acceptable software solution.	1	2	3 X	4	5
Show understanding of, and ability to apply, current theories, models, techniques and technologies that provide a basis for problem solving: the project problem area was well researched and appropriate theories, models, techniques and technologies were selected and developed for constructing the software solution.	1	2	3	4 X	5

Figure 1: Example partial scoring grid for one student.

2.4 Development of Assessment Rubrics

Rubrics were developed from the student learning outcomes. Essentially, a rubric is a translation of each outcome in the context of the capstone project. For example, for the outcome "Work effectively … to develop and deliver quality software", the rubric developed was "the requirements for the software were thoroughly discussed with the client, analyzed, and a working software solution has been designed based on quality design goals, implemented and documented".

Rubrics aid both the students and faculty. For the students, rubrics let the students know the criteria by which their projects will be evaluated. For the faculty, rubrics allow all projects to be evaluated according to the same criteria.

2.5 Collection of Assessment Data

Each row of the PTA grid represents a trait, i.e. an outcome, plus its associated rubric; each column represents a score in the range 1 to 5, where 1 represents poor and 5 represents excellent for a given outcome.

For every student presentation, demonstration and project report, each member of the judging panel scores the student for each outcome, according to its rubric. The overall score (the aggregate of the judging panel's scores) for each student allows the assignment of a grade for that student for the project.

And also, importantly for program assessment, aggregating the scores for all students provides a quantitative, direct measure of student learning for each key program SLO.

2.6 Analysis of Assessment Data

The aggregated scores from the grids of all students for all outcomes are analyzed by the faculty to determine if there are any outcomes for which the students are collectively under-performing. If there are one or more such outcomes, then steps are taken to understand why this happening and apply corrective action in those courses that service those outcomes. The quantitative nature of the assessment allows faculty to focus on strategies for any improvement necessary in the program. An example of a partial grid with the aggregated score for 30 students is shown in Figure 2 below. The second row shows that in this example, students are under-performing in communication abilities.

Outcome	Score				
Work effectively to develop and deliver quality software. the software requirements for the software were thoroughly discussed with the "client", analyzed, and a working software solution has been designed based on quality design goals, implemented and documented.	1	2	3	6	18
Able to communicate effectively both orally and in writing. the oral presentation was professional, well-organized (giving an overview of the project and describing the analysis and design of the software solution) and was delivered in a strong, clear voice in a confident and timely manner. The project report was written in a clear, coherent style, well-organized and structured with appropriate textual and diagrammatic descriptions of the analysis and design of the software solution.	4	4	8	8	6

Figure 2: Example partial grid of aggregated scores for 30 students.

3. DISCUSSION

The PTA method of program assessment has been in use for a few years at AUBG. It has provided useful feedback to faculty on how student learning, related to each learning objective, is progressing. One early indicator was the one outlined above – some students were not performing well in communication, either writing or presenting. As a result, courses were updated to give more feedback to students, with, for example, better rubrics for report writing and presentation skills. More courses included student report writing and presentations.

Although the traits and rubrics served a purpose, experience has shown that current scoring matrix is too coarse – there needs to be finer detail of traits and accompanying rubric.

For example, for the outcome "Have the ability to communicate effectively both orally and in writing", the traits may be Preparation/Content and Presentation. The first trait may be broken down to Organization; Quality of Content; Quality of Conclusion; etc., and the second may be broken down to Style: pace, voice quality, mannerisms; Handling of questions; etc. Such break downs would require to be accompanied by more detailed rubrics.

Able to communicate effectively both orally and in writing.	1	2	3
Traits			
Preparation/Content			
Organization			
Quality of information			
Quality of conclusion			
Quality of accompanying media			
Presentation			
Style: pace, voice quality, mannerisms			
Use of media			
Handling of questions			
Etc.			

Figure 3: Example of adding finer detail to communications trait.

Examples of detailed rubric

Organization of presentation – How will we know?

3 - Clear organization, reinforced by media. Stays focused throughout.
2 - Mostly organized, but loses focus once or twice.
1 - Loses focus three or more times during presentation.

Ability to Answer Questions - How will we know?

3 - Handles all questions with relevant, correct information. Speaks confidently.
2 - Answers majority of questions, but does not expand on answers.
1 - Unable to answer majority of questions with correct information.

Figure 4: Example of detailed rubric for communications trait.

4. CONCLUSION

This paper has described how capstone projects in the computer science degree program at AUBG are assessed using the method of Primary Trait Analysis. This assessment is based on the expected student learning outcomes developed by the Computer Science department at AUBG. The approach also allows tracking of quantitative measures over time to provide a clearer view of student learning.

5. REFERENCES

[1] NEASC home page: http://www.neasc.org/

[2] NEAA home page: http://www.neaa.government.bg/en

[3] Christozov D., Galletly J., Miree L., Karagiozov V. and Bonev S , 2011, Accreditation – a Tale of Two Systems, *Sixth International Conference - Computer Science 2011*, Ohrid, Macedonia.

[4] Bonev S., Christozov D., Galletly J. and Karagiozov V., 2005, Computer Science Curriculum in a Liberal Arts Institution: Transition from ACM/IEEE Curriculum Model 1992 to 2001, *Second International Scientific Conference – Computer Science*, Halkidiki, Greece.

[5] Karagiozov V., Christozov D., Galletly J. and Bonev S. 2005, E-learning in a Liberal Arts Institution: An Open Source Solution –the AUBG Experience, *Second International Scientific Conference – Computer Science*, Halkidiki, Greece.

[6] Karagiozov V., Christozov D., Galletly J. and Bonev S., 2008, Facilities and Support for Teaching Computer Science at the American University in Bulgaria, 2008, *Fourth International Scientific Conference – Computer Science*, Kavala, Greece.

[7] Christozov D., Galletly J., Karagiozov V. and Bonev S., 2007, Learning by Doing – the Way to Develop Computer Science Professionals, *IEEE Conference: Informatics Education - Europe II*, Thessaloniki, Greece.

[8] Christozov D., Galletly J., Karagiozov V. and Bonev S., Cooperatve Learning – the Role of Extra Curricular Activities, 2009 *Fourth Balkan Conference in Informatics*, Thessaloniki, Greece.

[9] NEASC standards page: http://cihe.neasc.org/standards_policies/standards/standards_html_version

[10] Bailie F., Whitfield D., & Abunawass A., Assessment and Its Role in Accreditation, 2007, *International Conference on Frontiers in Education: Computer Science and Computer Engineering*, Las Vegas, NV

[11] Sanders K., McCartney R., Program Assessment Tools in Computer Science: a Report from the Trenches, 2003, *SIGCSE '03 Proceedings of the 34th SIGCSE Technical Symposium on Computer Science Education*

[12] Angelo, T. A., and Cross, K. T., *Classroom Assessment Techniques* (2nd Edition), 1993, Jossey-Bass Publishers

[13] See, for example: http://assessment.aas.duke.edu/documents/DirectandIndirectAssessmentMethods.pdf

[14] Berheide, C., Doing Less Work, Collecting Better Data: Using Capstone Courses to Assess Learning, 2007 *Peer Review*, No. 9, pp. 27–30

[15] Lloyd-Jones, R., Primary Trait Scoring, in C. Cooper and L. Odell (eds.), *Evaluating Writing: Describing, Measuring, Judging*, 1997, Urbana, Ill., National Council of Teachers of English

[16] Walvoord, B.E. and McCarthy, L.P., *Thinking and Writing in College*, 1991, Urbana, Ill., National Council of Teachers of English

[17] Walvoord, B.E. and Anderson, V.J., *Effective Grading: A Tool for Learning and Assessment*, 1998. Jossey-Bass Publishers