Applying GQM Approach towards Development of Criterion-Referenced Assessment Model for OO Programming Courses

Norazlina Khamis, Sufian Idris, and Rodina Ahmad

Abstract—The most influential programming paradigm today is object oriented (OO) programming and it is widely used in education and industry. Recognizing the importance of equipping students with OO knowledge and skills, it is not surprising that most Computer Science degree programs offer OO-related courses. How do we assess whether the students have acquired the right object-oriented skills after they have completed their OO courses? What are object oriented skills? Currently none of the current assessment techniques would be able to provide this answer. Traditional forms of OO programming assessment provide a ways for assigning numerical scores to determine letter grades. But this rarely reveals information about how students actually understand OO concept. It appears reasonable that a better understanding of how to define and assess OO skills is needed by developing a criterion referenced model. It is even critical in the context of Malaysia where there is currently a growing concern over the level of competency of Malaysian IT graduates in object oriented programming. This paper discussed the approach used to develop the criterion-referenced assessment model. The model can serve as a guideline when conducting OO programming assessment as mentioned. The proposed model is derived by using Goal Questions Metrics methodology, which helps formulate the metrics of interest. It concluded with a few suggestions for further study.

Keywords—Object-oriented programming, programming assessment, criterion-referenced assessment model, goal questions metrics.

I. INTRODUCTION

PROGRAMMING is one of the most important courses offered in Computer Science degrees. Over the past two decades, many aspects of programming have been investigated with the intention to make programming accessible to wide variety of people. This leads towards the development of programming environments, tools, and languages to support the process of learning programming [5]. Robins and Rountree [9] in their research stated that the demands and interest in programming have grown increasingly and this lead to the popularity of the programming courses being offered. Virtually all computer science educations include programming courses as one of the subject in the syllabus.

Object oriented programming (OOP) appears to be the most influential programming paradigm today, and it is widely used in education and industry. In the literature, OOP has been suggested as a solution to many problems and it is not surprising that most Computer Science (CS) degree programs offer object oriented-related courses. In Malaysia, virtually all CS program include OO related courses [17].

Previously, OOP has always been considered as an advanced course, but now more universities in Malaysia have started to introduce OOP during first year of programming course [17]. However, teaching OOP remains difficult. Learning new knowledge is hard and in general, people are reluctant to learn new things. Students learning OOP have problems not only in developing program writing, comprehension, and debugging skills but also in understanding basic OO concepts. The art of programming includes knowledge of programming tools and languages, problem-solving skills, and effective strategies for program design and implementation.

Most of OOP courses in Malaysian universities usually take one semester or 14 weeks to be completed. How do we assess whether the students have acquired the right object-oriented skills after they have completed the courses? What does it mean by OO skills? Current assessment techniques do not seem to provide answers to those questions. Thus we think it is important to have a framework for assessing object-oriented skills students should acquire after they have completed their OOP courses. In this paper, we describe our approach on developing criterion-referenced assessment model to support the assessment of object-oriented programming courses.

The structure of this paper will be as follows. In section II we describe the motivation of this study. In section III, a brief description on criterion referenced assessment model, which is the approach used to assess OO skills. We present the

Norazlina Khamis is with the Faculty of Computer Science & Information Technology, University of Malaya, 50603 Kuala Lumpur, Malaysia (phone: +60379676376; fax: +60379579249; e-mail: azlina@um.edu.my).

Sufian Idris is with Faculty of Information Science & Technology, Universiti Kebangsaan Malaysia, 46000 Bangi, Selangor, Malaysia (e-mail: sufian@ftsm.ukm.my).

Rodina Ahmad is with the Faculty of Computer Science & Information Technology, University of Malaya, 50603 Kuala Lumpur, Malaysia (e-mail: rodina@um.edu.my)

methodological approach employed in this study in section IV. We conclude in section V with some suggestions for further study.

II. MOTIVATION

This study focuses on the following questions: Do grades reflect students' performance on OO skills? What are educators' perceptions of object oriented programming skills? Specifically, this study tries to identify what constitute OO skills and how do we measure these skills using appropriate assessment method by developing guidelines for assessment of OO programming courses. Our aim is to improve grading systems used in OO programming courses by developing a criterion referenced assessment model. The model shall accurately reflect differences in students' performance and should be fair and clear to students so they can monitor their own progress.

Programming is one of the skills that every computer science student is expected to master [8]. In more recent years, research from CS education has focused primarily on students' performance in introductory programming courses as correlated with quantifiable factors, such as mathematical background, major or gender [2][8][10][6][13][14]. The frequent questions in the programming education literature concern whether students have the particular skills that they are expected to acquire after completing their programming courses. Many authors have reported that Computer Science undergraduate students in their second year do not have the ability to write programs even though they obtained good grades in OOP courses in their first year [8][4][16]. This shows that the programming assessments they took during their first year of studies are inadequate (based on grades) and do not reflect their actual programming skills. Students are always being assessed through programming assignments, lab practices and examination. They will be given a grade based on the correctness of the answers.

The question of whether grades reflect the true performance of students in OO courses is becoming a hot issue. Erickson and Strommer [18] highlight grade depends a great deal on values, assumptions and educational philosophy. Grades only represent the extent to which students have successfully met the university requirements and it is impossible to make inferences about what they know by looking at their grades. In Norazlina and Sufian [19], we found out that most of the Malaysian universities use grades as an indicator of students' performance. Grades were assigned to the students according to the marks they obtained based on the marking scheme. Using this approach, it is difficult to determine whether the students' have acquired the required OO skills. Most of the educators we interviewed agreed that there should be some sort of guidelines that will help them evaluate whether students' have acquired the necessary OO skills or not.

Another research conducted by McCracken, addressed how to assess the programming ability of a large population of computer science students in solving a common set of programming problems. The focus of his research was towards assessing the basic programming skills based on a framework for first year learning objectives. Students were tested on a common set of programming problems. The findings from his research suggest that the majority of students performed poorly than they were supposed to be.

Norazlina and Sufian [19] studied in their recent survey the issues and difficulties of object-oriented programming by conducting a questionnaire for educators among Malaysian universities. One of the most obvious results was educators found that there is a need for a model describing what are object-oriented programming skills which they can use as a guideline in designing assessment for their OO programming courses. Currently assessment of OO programming courses is based on the educators' experience in OO.

It is normally difficult to identify the OO skills that the students have acquired after they have completed their OOP courses. Most of the assessment in OOP focus on assessing the quality of the program produced by the students and not the skills of the students itself [15]. Through informal interviews conducted with educators, it is found that a student is able to explain and understand a programming concept, (for example, what does an object mean), but fails to use it appropriately in a program. It appears reasonable that a better understanding of how to assess these skills is highly needed. It will not only help Computer Science educators to evaluate each student's skills in OO programming, but will also lead to better ways of appropriate pedagogy to teach OO programming courses. It is even important in the context of Malaysia where there is currently a growing concern over the level of competency of Malaysian IT graduates in programming [22].

This study proposes to develop an OO criterion referenced assessment model for OO programming courses as mentioned. This model is based on object-oriented curriculum in Malaysian universities. By establishing complete standards, grades are assigned by comparing a student's performance to a set of standards. Students who meet the learning targets will receive higher grades than those who do not meet the targets. Goal Questions Metrics approach will be applied in the development of the model. This study also proposes a computer-based assessment tool to assist educators in assessing student's skills in object oriented programming based on the criterion referenced model.

III. CRITERION REFERENCED ASSESSMENT MODEL

Approaches to assessment can be categorized into norm-referenced assessment and criterion-referenced assessment [20]. Norm-referenced assessment "remains the dominant… (approach) within higher education and 'naturally' preferred by most markers" [21]. Biggs [3] acknowledges that one of the main reasons for implementing norm-referenced assessment is for administrative convenience, but asserts that there is "no *educational* justification for grading on a curve" (p. 69).

A criterion-referenced model is an assessment model where the purpose is to assess the extent to which a student has achieved the goals of a course. In this context, the assessment is carried out aligned with specified criteria. Results are expressed in terms of how well a given student's performance

matches set criteria. This model is usually independent of any other student result. The standards are set before teaching takes place. A grade is assigned on the basis of the standard the student has achieved on each of the criteria. Fig. 1 shows our criterion-referenced assessment model in a diagrammatic form. We apply Goal Questions Metric (GQM) approach when developing the model. GQM approach will be discussed in detail in the following section.
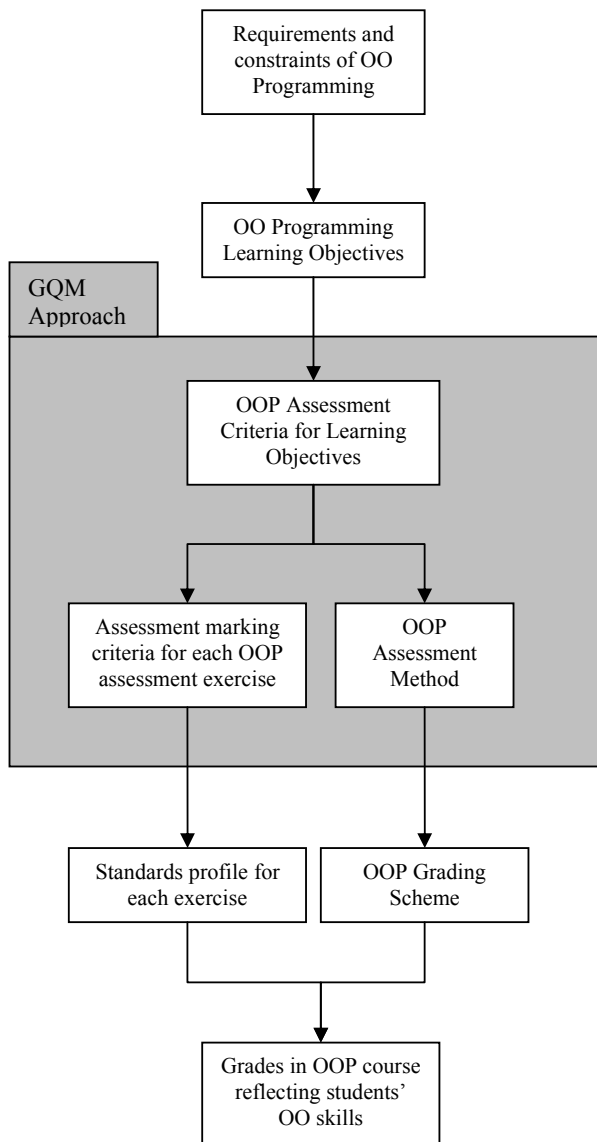

Fig. 1 Criterion-Referenced Assessment Model for Assessing Students' Skills in OOP

IV. GOAL QUESTIONS METRICS APPROACH

Goal Questions Metrics (GQM) is a paradigm for developing metrics program to support software development and maintenance. However, the basic concept of GQM is applicable whenever effective metrics are needed to assess satisfaction of goals. The GQM approach was initially developed by Victor Basilli and his colleagues in the 80s [11]. A summary of the GQM approach is given in Table I.

TABLE I
SUMMARY OF GOAL QUESTIONS METRICS

In our research work, we adapt this approach when developing a criterion referenced model for assessing OO programming skills. The next paragraph will describe in details how we applied the GQM approach in developing the model using one example of our current work. We begin by identifying the goals (G) of learning OO concepts that should be achieved by the students. This is then followed by the questions (Q) and the metrics (M).

A. Defining Goals (G) – Conceptual Level

The process of setting goals is critical to ensure successful application of the GQM approach [11]. We define the goals for developing the model based on the learning objectives for object-oriented courses. The learning objectives for OO programming courses from several universities in Malaysia were collected and studied. We also referred to learning objectives in ACM Computing Curricula 2001 Computer Science for OO courses [7]. We proposed to adapt learning objectives for object-oriented programming from the Computing Curricula 2001 and to combined them with a few learning objectives from several universities as standard learning objectives. The proposed learning objectives for OO programming are as follows.

1. Justify the philosophy of object-oriented design and the concept of encapsulation, abstraction, inheritance and polymorphism {G1}
2. Understand the concept of object interaction {G2}
3. Design, implement, test and debug simple programs in an object-oriented programming language {G3}
4. Describe how the class mechanism supports encapsulation and information hiding {G4}

5. Design, implement and test the "is-a" relationships among classes using class hierarchy and inheritance {G5}
6. Compare and contrast the notions of overloading and overriding methods in an object-oriented language {G6}
7. Describe how iterators access the elements of a container {G7}

Each of the learning objectives will be further detailed to ensure that educators are aware the topics involved in the OO syllabus. The first outcome from this phase is a set of goals based on detailed learning objectives for OO course. In this paper, we will limit our discussion onto one specific learning outcome, namely {G2}. According to Bennedsen, he found that understanding OO programming requires the understanding of object interaction goal [1]. Based on his model, we found that {G2} can be further detailed to sub goals as follows:

1. Students need to understand that an object-oriented program consists of several different objects {G.2.1}
2. Students need to understand object connection within certain structures {G.2.2}
3. Students need to understand the dynamics of these object structures {G.2.3}

B. Defining Questions (Q) – Operational Level

The next step is to define the questions for each goals/sub goals. For each goal/sub goals, we derive questions that define those goals as completely as possible. We show one example for learning objective {G.2.2} in Table II.

TABLE II
EXAMPLE FOR DEFINING QUESTIONS

GOAL	{G.2.2} Students need to understand object connection within certain structures
QUESTION	{Q.1} Do students know how to create a connection between objects?
	{Q.2} Do students understand connection between objects?

C. Defining metrics (M) – Quantitative level

Once the questions have been developed, we proceed to associating the questions with appropriate metrics. In this context, we will identify the appropriate measurement for each question. This will be done by identifying the assessment method that we can used to measure the goals.

TABLE III
EXAMPLE FOR DEFINING METRICS

GOAL	{G.2.2} Students need to understand object connection within certain structures
QUESTION	{Q.1} Do students know how to create a connection between objects?
	{Q.2} Do students understand connection between objects?
METRICS	{M1} Number of objects
	{M2} Output of interacting objects

Table III shows the example of defining metrics to the questions. Once the GQM model has been developed, we will identify and select the appropriate assessment methods, tool and procedures for each of the goals that we have identified earlier.

V. CONCLUSION

In this paper, we discussed our current work on the approach used to develop criterion-referenced assessment model. A method for developing a criterion referenced assessment model is presented. We have shown to what extent the results of the analysis of learning objectives on object-oriented programming courses can be used to prepare a criterion-referenced assessment model. An outline for a criterion-referenced assessment model based upon the Bennedsen's Competence model is also presented. For further work, we will focus on the development of the assessment method. We need to identify the appropriate assessment instrument to the validate model.

REFERENCES

[1] J. Bennedsen and E.M Carpersen, "A Competence Model for Object Interaction in Introductory Programming" in Proceeding of 18th Workshop of the Psychology of Programming Interest Group, 2006.
[2] J. Bennedsen and E.M Carpersen, "Teaching Object Oriented Programming" in 18th European Object oriented Workshop, 2004.
[3] Biggs, J. "What the student does: Teaching for enhanced learning", Higher EducationResearch and Development, 18(1), 57. 1999.
[4] C. Daly and J. Horgan, "Automatic Plagiarism Detection" Proceedings of the International Conference in Applied Informatics. pp 255-259, 2001.
[5] C. Kelleher and R. Pausch, "Lowering the Barriers to Programming: A Taxonomy of Programming Environments and Languages for Novice Programmers", ACM Computing Surveys, 37(2), pp. 83-137, 2005.
[6] E. Lahtinen and T. Ahoniemi, "Visualisations to Support Programming on Different Levels of Cognitive Development" in Proceedings of The Fifth Koli Calling Conference on Computer Science Education.
[7] Computing Curricula 2001 Computer Science, Final Report, December 2001.
[8] M. McCracken, V. Almastrum, D. Diaz, M. Guzdial, D. Hagan, D. Kolikant, C. Laver, L. Thomas, I. Utting, and T. Wilusz, " A multi-national, multi-institutional study of assessment of programming skills of first-year CS students." In ACM Sigcse Bullettin, 33(4), pp 125-140. 2001.
[9] S. Robins, J. Rountree, and N. Rountree, "Learning and Teaching Programming: A Review and Discussion", Computer Science Education, 13(2), pp. 137-172, 2003.
[10] F. Wei, H.S Moritz, M.S Parvez, and D.G Blank " A Student Mpdel For Object Oriented Design and Programming" in Journal of Computing Science in Colleges, Vol 20(5) p 260-273. 2005.
[11] V. R Basilli, Gianlugi. C and Rombach H. D "The Goal Question Metric Approach" 1996 Available on http://wwwagse.informatik.uni-kl.de/pubs/repository/basili94b/encyclo.gqm.pdf
[12] M. Kolling "The problem of Teaching Object-oriented Programming, Part 1: Languages," Journal of Object Oriented Programming, 11(8), 8-15, 1999.

[13] K.B. McKeithen**,** J.S. Reitman, H.H. Rueter, S.C. Hirtle, "Knowledge Organization and Skills Differences in Computer Programmers," *Cognitive Psychology*, 307-325, 1981.

[14] H.B Graham and L.M Verbyla, "An Online programming Assessment Tool," in *Proceedings of the Fifth Australasian Conference on Computing Education,* Vol 20, Adelaide, Australia, 69 – 75, 2003.

[15] J.F Patenaude, E. Merlo, M. Dagenais, B Lague, "Extending Software Quality Assessment Techniques to Java Systems," *IWPC*, Seventh International Workshop on Program Comprehension, 49, 1999.

[16] M.E Califf and M. Goodwin, "Testing Skills and Knowledge: Introducing a Laboratory Exam in CS1" in Proceedings of the 33rd SIGCSE Technical Symposium on Computer Science Cincinnati, Kentucky, 217 – 221, 2002.

[17] Norazlina Khamis and Sufian Idris, "Investigating Current Object Oriented Programming Assessment method in Malaysia's Universities", in Proceeding of ICEEI 2007, Bandung, Indonesia, 2007.

[18] Erickson, B. L., and Strommer, D. W. Teaching College Freshmen. San Francisco: Jossey-Bass, 1991.

[19] Norazlina K., and Sufian. Idris. "Issues with Object Oriented Programming Assessment in Malaysia's Univerisities" to be publish in Proceeding of 1st International Malaysian Educational Technology Convention, 2007.

[20] Dunn, L., Morgan, C., O'Reilly, M., & Parry, S. "The Student Assessment Handbook", London: Routledge Falmer. 2004.

[21] Rust, C., Price, M., & O'Donovan, B. "Improving students' learning by developing their understanding of assessment criteria and processes" *Assessment & Evaluation in Higher Education,* 28(2), 147.2003.

[22] Ungku Harun AlRashid Ahmad. "Meeting The Demands of Global Firms: Survey Finding" Presented at Workshop on Enhancing Graduates Employability in A Globalised Economy, MDC. 2004.