research 1

by Mukhlis Latief

Submission date: 31-Oct-2021 10:25AM (UTC+1100)
Submission ID: 1688574266
File name: Anti-cheating_software_tool.pdf (559.5K)
Word count: 3060
Character count: 16438

IOP Conference Series: Materials Science and Engineering

Anti-cheating software tool: Prototype of problem generator software for linear algebra introductory test

Anti-cheating software tool: Prototype of problem generator software for linear algebra introductory test

M S Tuloli*, M Latief and M Rohandi

Department of Informatics, Faculty of Engineering, Universitas Negeri Gorontalo, Gorontalo, Indonesia

*syafri.tuloli@ung.ac.id

Abstract. To prevent students from cheating in the test, one of the strategies is by making a question that differs for each student. This approach can be used by using question-bank or question generator, the first approach is prone to cheating especially because of the widely available communication technology, this means the question generator is the only hope. This strategy is very possible to be used especially in subjects whose material has a standard pattern, such as in mathematics (calculation/structure patterns) and computer programming (logic patterns). As a branch of mathematics and also subjects in the informatics engineering curriculum, linear algebra has the potential to use this approach. But this approach involves the creation of a huge number of questions (and calculation of its answer), this requires extra efforts, we need software that can help generate questions and calculate its answers automatically. This study proposes a question-answer (problem) generation model and uses a software engineering approach (software requirement, design, implementation, and testing) to test the model. This study shows that the model implemented in this software was able to meet the need for generating questions for selected material on introductory subjects in linear algebra, with some caveat.

1. Introduction

1.1. Problems

Cheating by students has become a common problem, studies have shown 61% of students have cheated on an exam [1]. This becomes a concern, especially because many studies have shown a correlation between cheating committed at school and cheating in the work environment [2].

Teachers nowadays must be willing to consider alternative approaches in conducting an exam. Approaches that can be taken to reduce cheating is to guard against the possibility of cheating such as increasing supervision [3] like increasing supervisory personnel or using technological assistance such as CCTV or even surveillance robots [4], or psychological approaches such as signing honor code [5] [2]. The prevention approach becomes more difficult because of highly advanced communication technology nowadays so this approach will get its limitations [6].

Another approach is to make cheating efforts become cannot be done or useless, by making different questions for each student. To implement this approach two common ways can be used, the first way is to use a question bank and the second way to use the question generator engine (program).

The approach of using a question bank has the advantage that it does not require complex development, it can even be done manually (written on paper), but has its drawbacks, the number of

question variations is limited to the number of questions in the question bank. With the growing development of the use of digital storage media, students can store questions and answers that have been faced/worked so they can use them if the same questions arise (which is very possible because of the limited number of questions in the question bank).

The approach of using a question generator engine is the ideal approach because it has several advantages: it does not require storing questions, does not require effort in making questions, and does not have limitations in the number of questions generated. But the drawback is that it involves the development of a question generation model that must be applied in an application form and needs analysis to maintain the same level of difficulty for each question which makes this approach considered impractical and less used in education.

Moreover, by using different questions, the grading work becomes more difficult, because teachers must also make all the answers to the questions. This shows the need for an automatic generation method for both the questions and also the answers. Generating questions with randomized parts may be done easily (by syntactic manipulation), but generating answers to those questions requires a more intricate way to produce the correct answer.

This study aims to design a model for generating questions and answer (problem) from a subject and apply these models to the generation of questions for introductory subjects in linear algebra. Research offers a problem-generation-model and uses a software development process (requirement analysis, design, implementation, and testing) to test the model. In chapter I the problem is explained, chapter II discusses the literature review on the problem of cheating, chapter III and chapter IV on the proposed model, results and discussion, the final chapter provides conclusions.

2. Literature review

There are many ways of cheating committed by students [7]: impersonating, using forbidden documents/tools (e.g. calculator), peeking, peer collaboration, outsider assistance, student-staff collusion. With so many ways that cheating can occur, teachers need to recognize and deal with these possibilities.

The cause of this behavior has been widely studied, the factors that influence this behavior can be many things such as status (students/non-students), age [8], peer influence and individualism/collectivism orientation [3], negligence by teachers and material not relevant to the subject and [5], and other causes.

Technological development, although it can help a learning process (e.g. for example with electronic exams), seems to also facilitate cheating, and is considered an easier medium to cheat [7]. The use of technology can sometimes be worse than the non-technological way, for example, the use of student exam guard robots actually makes students talkative (less discipline) in the exam [4].

The advancement of information/communication technology been facilitated the development of cheating practice into a business, for example, the Essay Mills service which makes contract-cheating easier [9]. These developments are difficult to control, so the most recommended approach is to focus teacher effort to improve student work verification [10].

3. Proposed methods

The proposed method is a model consists of three layers (Figure 1), namely the User Interface (UI Module) which is responsible for handling input from the user and displaying outputs to the user, the Problem layer (Problem Module) which handles the types of problem models to be generated, and the core layer (Core Module) which is a representation of objects in the problem domain along with the implementation of operations that can be performed on those objects.

In UI Module can be defined method and format of user input. The input method like a command line input, or graphical user interface interaction, or load from a file, will be implemented in this module. The input format (text sequence or control layout or text format) is also implemented in this module.

The problem module focused on the definition and implementation of the problem-model. For each problem model can be defined as a valid and invalid type of problem, a valid problem is a problem that has a defined answer, and an invalid problem is a problem that cannot be solved. The problem module controls the probability of emerging problem type, the ratio of problem type from the total problem collection.

The Core module as the representation of objects in the problem-domain focuses only on the actual structure and behavior/operation of the object it represents. In this module, a verification needed to assure the correctness and accuracy of the object (and its operation) with the real object it represents. This module needed to determine the validity of a question and also generate a valid answer to the question.

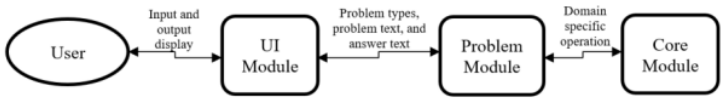

Figure 1. Proposed model.

4. Result and discussion

The implementation of the proposed model is carried out by creating static program models in the form of class diagrams and dynamic model programs in the form of sequence diagrams, these models are implemented into programs written in the Java programming language. The subject domain used as a case is in the introductory subject of linear algebra (matrix operations).

4.1. Class diagram

The static depiction of the program is illustrated in the class diagram (Figure 2). Because the implementation progress is still a prototype, the UI module that handles input and output formatting is still under development and is still using hardcoded input (directly to source-code), the focus of development is still on developing the problem generation module and core module.

4.1.1. Problem module. The class diagram is shown that the problem module is implemented as a Problem class. In this class, problem models are implemented, for example, matrix addition problems, matrix multiplication scalars, and other problem models. This class is responsible for generating text that is the problem and the text that is the answer to each problem model.

In almost every problem model has the possibility for operations (e.g. matrix operation) that can be done (valid) and operations that cannot be done (invalid), this is implemented in every method in this class. Parameters needed by problems can be stored directly in the method (e.g. for uniform types of problems) or can be opened as a method parameter so that it can be determined by the user (from the UI module).

4.1.2. Matriks class (Matrix representation class). Regarding the core module, because the case used is still limited to introductory subjects in linear algebra (especially in the matrix), it is only implemented into two classes, namely the Matriks class and the Bilangan class (Bilangan is Numeral in Bahasa). The Matriks class stores representations of matrix values and operations on the matrix, such as addition, multiplication, inverse, determinant operations, etc. Matrix operations are implemented into methods, some of which directly produce text and some of them produce Matriks class objects, the method that produces Matriks object is to accommodate further operations (operations more than once).

The value representation is implemented as an object of the Bilangan class. The number class is a representation of a value in the matrix, so it is prepared to handle integers, and also fractions (numerator and denominator). The Bilangan class also has been implemented basic arithmetic operations on numbers.

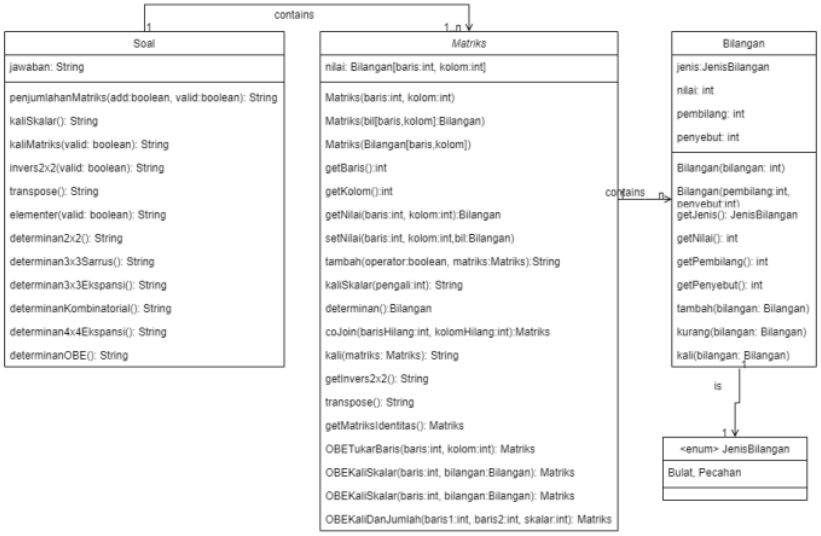

Figure 2. Class diagram.

4.2. Sequence diagram

Sequence diagrams show a dynamic picture of a program consisting of sequences of operations and messages sent between objects. To provide a general sequence description of program operations the sequence diagram of one of the problem generation operations (matrix multiplication) is shown in Figure 3.

The sequence of generating matrix multiplication problems start with the input given by the user which consists of: the type of problems to be raised, the number of problems needed (i.e. according to the number of students), and other parameters needed/provided by the program (e.g. matrix size, lower limit / for values, etc.). The UI module (Main class) accepts input, then uses random numbers to determine to generate valid multiplication or invalid multiplication problems. The type of problems (valid/invalid) will be used by the Problem Module (Soal Class, Soal is problems in Bahasa) to determine the dimensions (row and column size) of the matrix that will be used by the two matrices to be multiplied.

The dimensions of the two matrices to be multiplied (matrix A and matrix B) are used to create the Matriks class object. The Matriks class object will initialize its values by creating objects from the Number class. After the second matrix object to be multiplied has been created, a matrix multiplication operation is performed, by calling the kali method (kali is multiplication in Bahasa) on matrixA where the parameter is the matrixB object. The results of the method call are processed by the Soal class object and returned as an answer text of the Soal class. The question text generated by the Soal class object and the answer text resulting from the multiplication operation is returned to the main class object for formatting and displayed to the user so that it is easily understood by the user.

4.3. Testing

From the results of testing the program (Table 1), it was found that the implementation of the proposed model succeeded in fulfilling two requirements, able to generate problems with the number needed and also according to the parameters needed by the problem model (test-case # 1). Similarly, the program can generate a variety of valid and invalid problems for each problem-model (test-case # 3). But the program has a possibility to generate wide degrees of problem difficulty (test-case # 2), it is

still possible that very difficult and very easy problems arise in one group of problems. This can make students feel unfair because the time needed to work on very easy problems can be very different from the time to work on difficult problems.

This deficiency can be solved by making improvements to the definition of the problem model, by adding boundary parameters (lower and upper limits) for example for matrix dimension limits, cell value limits, etc. The addition of the parameter model of this problem can have a bad effect by making the problem module interface or UI module interface more complex, so it is necessary to analyze the best set of these parameters.

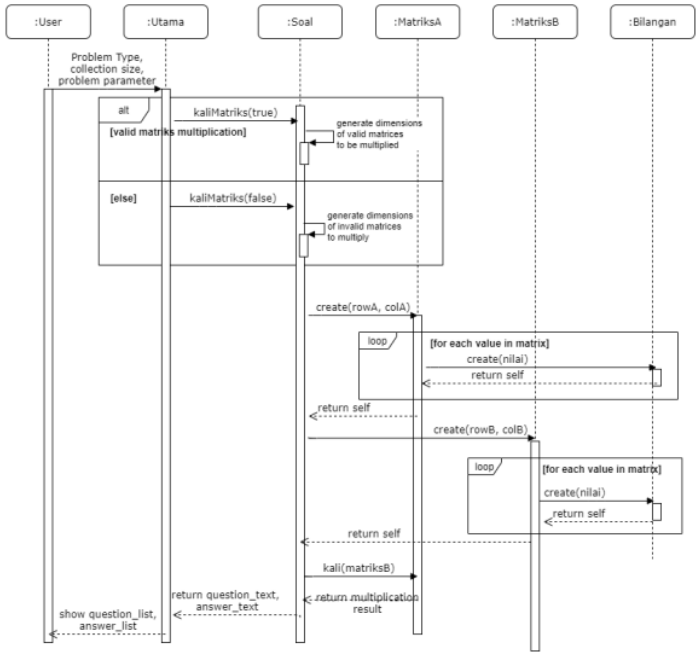

Figure 3. Sequence diagram of matrix multiplication problems.

5. Conclusion

The results obtained show that the proposed model for problem (question-answer) generation has been proven to meet the needs of the random problem generation. The proposed model allows modularization of components related to the display and input settings of users, from components that represent the problem model, and components that are representations of the structure and behavior of objects in the problem domains.

Implementation of the model by using cases in the introductory of linear algebra (matrices) is proven to be able to generate variations of problems both in the form of valid problems and the form of invalid problems. Questions generated in accordance with the amount specified and also according to the parameters required by the problem model. The application still has shortcomings, namely the probability of generating high variation in the difficulty level in a group of problems.

The future direction of research is on testing proposed models in other domains such as mathematics or programming to see the ability of the model to accommodate the specific needs of the domain. Development also needs to be done on the use of a graphical interface making it easier for common users to use the program.

Table 1. Testing result.

Test-Case	Objective	Expected output	Output	Conclusion
#1	To test the program's ability to generate a number of random problems automatically, based parameters specified by the user	The program generates a number of problems according to the parameters inputted (number of problems, limit values, dimensions, etc.)	The program generates a number of problems according to the parameters inputted	Acceptable
#2	To measure problems difficulty variation generated by the program	The program generates problems with a small variation of difficulty	The program generates problems with a high variation of difficulty	Needs repair
#3	To test program ability to generate a valid and invalid type of problem	The Program generates variations of valid and invalid problems for a problem model	The program generates valid and invalid problems	Acceptable

References

[1] Bucciol A, Cicognani S and Montinari N 2020 Cheating in university exams: the relevance of social factors *Int. Rev. Econ.* 1–20
[2] Graves S M 2008 Student cheating habits: A predictor of workplace deviance *J. Divers. Manag.* **3** 15–22
[3] Zhang Y and Yin H 2020 Collaborative cheating among chinese college students: the effects of peer influence and Individualism-Collectivism orientations *Assess. Eval. High. Educ.* **45** 54–69
[4] Mubin O, Cappuccio M, Alnajjar F, Ahmad M I and Shahid S 2020 Can a robot invigilator prevent cheating? *AI Soc.* 1–9
[5] Iberahim H, Hussein N, Samat N, Noordin F and Daud N 2013 Academic dishonesty: Why business students participate in these practices? *Procedia-Social Behav. Sci.* **90** 152–6
[6] Gobie W, Wesene G, Aynalem M, Amare Z and Melaku A 2020 Possible Reduction Mechanisms of Exam Cheating Practices for First Year Management Regular Students in Human Resource Management Course *Am. J. Educ. Inf. Technol.* **4** 19
[7] Chirumamilla A, Sindre G and Nguyen-Duc A 2020 Cheating in e-exams and paper exams: the perceptions of engineering students and teachers in Norway *Assess. Eval. High. Educ.* 1–18
[8] Fosgaard T R 2020 Students cheat more: Comparing the dishonesty of a student sample and a representative sample in the laboratory *Scand. J. Econ.* **122** 257–79
[9] Lancaster T 2020 Academic Discipline Integration by Contract Cheating Services and Essay Mills *J. Acad. Ethics* 1–13
[10] Amigud A and Dawson P 2020 The law and the outlaw: is legal prohibition a viable solution to the contract cheating problem? *Assess. Eval. High. Educ.* **45** 98–108

research 1

ORIGINALITY REPORT

11%	10%	9%	8%
SIMILARITY INDEX	INTERNET SOURCES	PUBLICATIONS	STUDENT PAPERS

PRIMARY SOURCES

1	Submitted to UIN Sunan Gunung DJati Bandung Student Paper	4%
2	earchive.tpu.ru Internet Source	2%
3	e-journal.uajy.ac.id Internet Source	2%
4	Sorayda Trejos, John Fredy Barrera, Roberto Torroba. " Erratum: Optimized and secure technique for multiplexing QR code images of single characters: application to noiseless messages retrieval (2015 085702) ", Journal of Optics, 2015 Publication	1%
5	Benedikt von Bronk, Alexandra Götz, Madeleine Opitz. " Locality of interactions in three-strain bacterial competition in ", Physical Biology, 2018 Publication	1%
6	T Abdillah, R Dai, E Setiawan. "Optimizing the Information Presentation on Mining Potential	1%

by using Web Services Technology with Restful Protocol", IOP Conference Series: Materials Science and Engineering, 2018
Publication

7	www.nature.com Internet Source	<1%
8	S C Wibawa, D E Dermawan, N G A G E Martiningsih, M Mashudi, C In. "Analysis validation of gamification fashion photography", IOP Conference Series: Materials Science and Engineering, 2021 Publication	<1%
9	www.lis.ac.cn Internet Source	<1%

Exclude quotes	Off	Exclude matches	Off
Exclude bibliography	On		

research 1

GRADEMARK REPORT

FINAL GRADE

/0

GENERAL COMMENTS

Instructor

PAGE 1

PAGE 2

PAGE 3

PAGE 4

PAGE 5

PAGE 6

PAGE 7