DOI: https://doi.org/10.59294/HIUJS.VOL.6.2024.631

Designing a knowledge assessment system for the Data structures and Algorithms course

Hoang Ngoc Long*, Mai Trung Thanh and Do Van Nhon
Hong Bang International University, Vietnam

ABSTRACT

The course of Data Structures and Algorithms is one of the fundamental and crucial subjects in the curriculum of the Information Technology field. This course serves as the foundation for understanding and applying data structures and algorithms in solving real-world and complex problems. However, teaching and learning Data Structures and Algorithms often face many challenges. Students often struggle with understanding and applying concepts, algorithms, and data structures to real-world problems. Moreover, assessing students' knowledge and learning performance in the course also requires accuracy and objectivity. In this context, designing and implementing a system to support testing and evaluating knowledge for the Data Structures and Algorithms course becomes truly necessary. Such a system not only helps students grasp knowledge effectively but also provides opportunities for instructors to monitor students' learning progress and provide feedback for them to improve their learning outcomes. The paper will present the design of techniques for building an application for testing and evaluating knowledge in the Data Structures and Algorithms course. This application will be a useful tool for both instructors and students, thereby enhancing the quality of teaching and learning in Information Technology-related subjects.

Keywords: *E-learning, intelligent software, intelligent tutoring system*

1. INTRODUCTION

Intelligent Tutoring Systems (ITS) are intelligent computer-based educational software systems designed to enhance the teaching effectiveness of instructors and the learning experience of students. These systems use Artificial Intelligence methods to make decisions in education, related knowledge in the field of teaching, student learning activities, and the assessment of student learning related to the subject knowledge [1]. Therefore, ITS is a popular type of educational system and is becoming a primary means of delivering education, leading to significant improvements in student learning [1] [2].

Testing and assessing knowledge related to subjects are essential components of Intelligent Tutoring Systems (ITSs). The objective of assessment is to assist students in gaining a deeper understanding of their level of knowledge in the subject, including areas of the subject matter that students have not yet mastered. In the curriculum of the Information Technology field, the course Data Structures and Algorithms is one of the important foundational subjects. The course helps students in the field of Information Technology to understand and learn how to implement data structures and algorithms to solve real-world and complex problems when building applications, as well as enhancing programming skills. However, learning data structures and algorithms is not easy for every student. The course requires learners to think abstractly and grasp the properties, implementation of algorithms, and data structures.

Currently, surveys in studies [1-4] have shown that Intelligent Tutoring Systems (ITS) can effectively support learners in acquiring knowledge in various fields. However, there is still no complete solution for a tutoring system in assessing subjects within the IT field. The study [5] presents the requirements for an intelligent educational system that serves two primary functions: querying course knowledge and evaluating learner proficiency through multiple-choice testing. The study also propose a solution to design the knowledge base, inference engine, and tracing system based on a knowledge model that integrates ontology and knowledge graph. Nevertheless, for the multiple-choice testing function, the proposed solution does not apply any testing theory to build exam questions or question

Corresponding author: MSc. Hoang Ngoc Long
Email: longhn@hiu.vn

bank. Although the solution designed in the paper is very useful for building intelligent systems in education, the paper does not mention the specific field in which the solution has been applied for knowledge testing and evaluation. The system for illustrating algorithms in study [6] presented a solution for intelligent automatic algorithm illustration for several subjects that heavily involve algorithms, such as data structures and algorithms, graph theory, etc. The system supported two user groups: end-users and knowledge administrators. The knowledge about algorithms was modeled, allowing the knowledge administrators to input various types of algorithms into the system. The second function was for users learning the algorithms, enabling them to view step-by-step illustrations of the algorithms based on specific problem data entered into the system. However, this system still lacks the functionality for assessing knowledge of the subject matter. The authors in paper [7] presented an automated knowledge assessment support system in the domain of high school mathematics. The paper showcased an application of expert systems in supervising and evaluating learners' competencies within the scope of content knowledge and thematic content in high school mathematics. Additionally, we can easily find commercial applications that support assessment of knowledge in general education subjects such as English (International English Test) [8] or Mathematics (VioEdu, HocMai) [9, 10]. However, there is not yet a specific application that supports the assessment of knowledge in Information Technology subjects, specifically Data Structures and Algorithms. There are also Learning Management Systems (LMS) that support knowledge assessment based on data added to the system, such as Elearning [11]. These applications only support testing but do not provide a system that offers detailed evaluation of students' or learners' test results. Currently, there is still no complete solution for developing a support system for testing and assessing knowledge in the Data Structures and Algorithms subject.

In this paper, a knowledge base model for assessing the Data Structures and Algorithms course will be presented. The system built on this model will support the creation of multiple-choice tests in alignment with the course's learning outcomes and the suitability level for students base on classical test theory. By using several analytical and statistical techniques, the system will automatically assess the user's capabilities by providing both qualitative and quantitative information. In addition to evaluating the learner's knowledge aspects, the system can also offer feedback and suggestions on the areas where the learner needs improvement. This system will help students accurately assess their abilities, reinforce their learning, and enhance related programming skills.

2. KNOWLEDGE MODEL FOR MULTIPLE CHOICE ASSESSMENT SYSTEM

In designing a support system for testing and assessing knowledge in the Data Structures and Algorithms course using objective multiple-choice format, building a knowledge representation model is a crucial part to ensure flexibility, efficiency, and diversity in the testing and assessment process. This knowledge representation model needs to: i) ensure the representation of concepts and knowledge related to the course; ii) have a structured organization for building a question bank; iii) include a model that links course content with the question bank.

2.1. Knowledge base model of the data structures and algorithms course

2.1.1. Modeling the knowledge base for the content of Data Structures and Algorithms course

The Data Structures and Algorithms course introduces and provides knowledge about data structures and algorithms in the field of Information Technology. The course equips students with knowledge on analyzing and designing computational algorithms for computers, as well as basic data structures and their applications. It helps students consolidate and develop their programming skills. After completing this course, students should be able to achieve the following course learning outcomes (CLO):

CLO1: Understand the concepts and roles of data structures and algorithms in the curriculum, the criteria for evaluating data structures and algorithms. Grasp the concept of algorithm complexity, basic computational techniques, and algorithm representation;

CLO2: Identify and state search problems, related factors, and constraints in the problem, model the problem, determine methods for solving the problem, and analyze the advantages and limitations of these methods;

CLO3: Identify and state sorting problems, related factors, and constraints in the problem, model the problem, determine methods for solving the problem, and analyze the advantages and limitations of these methods;

CLO4: Grasp basic data structures and be able to apply these basic data structures to write simple application programs;
CLO5: Grasp the implementation of data structures and algorithms, and be able to apply them to solve simple problems;

Definition 2.1: The model for organizing the knowledge base to represent the content knowledge of the Data Structures and Algorithms course has the form:

(T, Q, R, CLO)

In which, *T* is the set of topics in the course organized in a tree structure; *Q* is the set of multiple-choice questions in the course; *R* is the set of relationships between topics in the course and multiple-choice questions; *CLO* is the set of course learning outcomes. For each question $q \in Q$, it will have the structure:

(question_content, answers, correct_answers, difficult_level, discrimination_level, bloom_level)

In this model, *question_content* is the content of the question; *answers* is the set of possible answers; *correct_answers* is the set of correct answers *(correct_answers* $\in$ *answers)*;

difficult_level represents the difficulty level of the multiple-choice question, calculated according to classical test theory. The difficulty level of a multiple-choice question is the percentage of examinees who answer the question correctly out of the total number of examinees who attempt the question [12, 13];

discrimination_level represents the discrimination level of the multiple-choice question, indicating the question's ability to differentiate between groups of students with different abilities [12, 13];

bloom_level represents the cognitive level of the multiple-choice question according to the Bloom's Taxonomy scale of the course's learning outcomes [12, 13]. The paper utilizes 4 levels of the Bloom's Taxonomy scale: *Remembering (R), Understanding (U), Applying (AP), and Analyzing (AN)*.

This knowledge base model has been applied to organize a bank of multiple-choice questions for the Data Structures and Algorithms course. This question bank is used to generate a test suitable for the requirements and abilities of students, and the results of this test serve as the basis for diagnosing students' knowledge levels.

2.2.2. Modeling user of the system

The proposed system can track the student's learning progress through tests. The system collects content and results from the user's tests. Based on this information, the system can evaluate the student's knowledge development.

Definition 2.2: The user model in the system is structured as follows:

(PROFILE, TEST_LIST)

In which, *PROFILE* is the user's system information; *TEST_LIST* is the set of lists of tests that the user has completed. For each *test* $\in$ *TEST_LIST*, it will have the structure:

(t, cq, icq, m)

In this model, *t* is the test that the user has completed; *cq* is the set of questions in test *t* that the user answered correctly; *icq* is the set of questions in test *t* that the user answered incorrectly; *m* is the score of test *t*; questions *cq* and *icq* have structures similar to questions $q \in Q$.

2.3. Knowledge base model for multiple choice questions bank and course content linking

A test is usually designed by selecting questions from a question bank. A question bank is a collection of a relatively large number of questions, where each question is described with specific content areas and its parameters, such as difficulty level *(difficult_level)* and discrimination level *(discrimination_level)* according to classical test theory [12]. The question bank must be designed to allow operations such as excluding or modifying poor questions, and adding good questions, so that the quantity and quality of the questions continually improve. The steps for designing a standardized test and a question bank for the Data Structures and Algorithms course can be summarized as follows [12]:

(1) Build a knowledge matrix for the Data Structures and Algorithms course;

(2) Assign instructors to create a certain number of questions according to the requirements associated with the cells of the knowledge matrix as an example shown in Table 1;

(3) Organize reviews, edit, exchange questions among colleagues, and store the questions in computer databases. This step will result in a set of meticulously edited multiple-choice questions stored on the computer. However, this is not yet a question bank, as the questions have not been tested to determine their difficulty and discrimination parameters;

(4) Create trial tests and conduct pilot testing on groups of students representing the overall population to be assessed;

(5) Grade and analyze the trial test results. The process of statistical analysis and calibration of the multiple-choice questions will yield two types of results: one, providing the parameters of the multiple-choice questions, and two, identifying poor-quality questions.

(6) Handle poor-quality questions by either modifying them or discarding them if they are of such low quality that they cannot be corrected. The revised questions are then stored again. Through this step, a question bank begins to take shape. The process of pilot testing and refining the questions can be conducted multiple times, with each iteration resulting in the modification and improvement of some questions in the question bank, thereby expanding and enhancing the question bank.

(7) Once the quantity and quality of the questions in the question bank are assured, official tests can be designed for formal exams. The structure of an official test should be represented by a corresponding knowledge matrix.

Table 1. The knowledge matrix for a final exam at Level 1

Course Learning Outcomes	Data Structures and Algorithms Course Content	Number of Questions according to Bloom's Taxonomy Cognitive Levels				Total
		R	U	AP	AN	
CLO1	*Chapter 1. Overview of Data Structures and Algorithms*	5	3	1	1	10
CLO2, CLO3	*Chapter 2. Searching and Sorting*	4	3	1	1	9
CLO4, CLO5	*Chapter 3. Linked List*	4	3	1	1	9
CLO4, CLO5	*Chapter 4. Stack, Queue*	4	3	1	1	9
CLO4, CLO5	*Chapter 5. Tree*	4	2	1	1	8
CLO4, CLO5	*Chapter 6. Hash table*	4	1	0	0	5
Total		**25**	**15**	**5**	**5**	**50**
Percentages		**50%**	**30%**	**10%**	**10%**	**100%**

Definition 2.3: The model for the test structure is as follows:

$$(L, Q_R, Q_U, Q_{AP}, Q_{AN})$$

In this model, L is the difficulty level of the test; Q_R, Q_U, Q_{AP}, Q_{AN} are the set of questions at the *Remembering (R), Understanding (U), Applying (A), Analyzing (A)* levels of Bloom's Taxonomy correspondingly.

The difficulty of a test can be designed according to Bloom's Taxonomy, the difficult level (p) of the questions, and the discrimination level (d) of the questions as in Table 2. The difficulty level of a question can be categorized as follows: easy question *(difficult_level ≤ 0.7)* moderately difficult question *(0.6 ≤ difficult_level < 0.7)*, medium question *(0.4 ≤ difficult_level < 0.6)*, relatively difficult question *(0.3 ≤ difficult_level < 0.4),* and fairly difficult question *(difficult_level < 0.3)* [12, 13]. The difficult level of a question is computed by classical test theory [12, 13]. A good discrimination level for a question falls within the range of [0.2 – 1] and is also computed by classical test theory [12, 13]. If *discrimination_level* < 0.2, the question is considered poor and needs to be either removed or revised for improvement.

Table 2. Difficult level of the test

Difficulty Level of the Test	Proportion of Questions by Cognitive Levels According to Bloom's Taxonomy				Difficult level (p)	Discrimination level (d)
	R	U	AP	AN		
Level 1 (Easy)	50%	30%	10%	10%	$p \geq 0.7$	$0.2 \leq d \leq 1$
Level 2 (Moderate)	40%	30%	20%	10%	$0.6 \leq p < 0.7$	$0.2 \leq d \leq 1$
Level 3 (Medium)	30%	40%	20%	10%	$0.4 \leq p < 0.6$	$0.2 \leq d \leq 1$
Level 4 (Difficult)	20%	30%	30%	20%	$0.3 \leq p < 0.4$	$0.2 \leq d \leq 1$
Level 5 (Very Difficult)	20%	20%	30%	30%	$p < 0.3$	$0.2 \leq d \leq 1$

3. MODELING OF PROBLEMS AND ALGORITHMS

Based on definitions 2.1, 2.2, 2.3, the system supporting the assessment of knowledge in the subject of Data Structures and Algorithms needs to

address the following issues:

(1) Creating tests according to specific requirements: the system can generate tests based on the desired number of questions, the difficulty level of the test, and the specific topics of interest within the subject.

(2) Assessing students' knowledge through a particular test: based on the result of a particular test taken by students, the system can provide feedback on their performance and abilities in each topic or course learning outcome.

3.1. The problem and algorithm of generating a test

Definition 3.1: The problem of generating tests according to requirements is defined as follows:

(#question, test_level, topic, time) $\rightarrow$ *(L, Q_R, Q_U, Q_{AP}, Q_{AN})*

Where *#question* is the number of questions in the test; *test_level* is the difficulty level of the test as shown in Table 2; *topic* is the set of topics in the subject to be tested *(topic* $\subseteq$ *T)*; *(L, Q_R, Q_U, Q_{AP}, Q_{AN})* are defined in 2.3 and meet the requirements of *(#question, test_level, topic)*; *time* is the duration of the test, measured in minutes.

With the knowledge organization *(T, Q, R, CLO)* defined in 2.1 and the parameters for generating a test *(#question, test_level, topic, time)* as defined in 3.1, the algorithm of generating a test for the subject of Data Structures & Algorithms is described as in Figure 1.

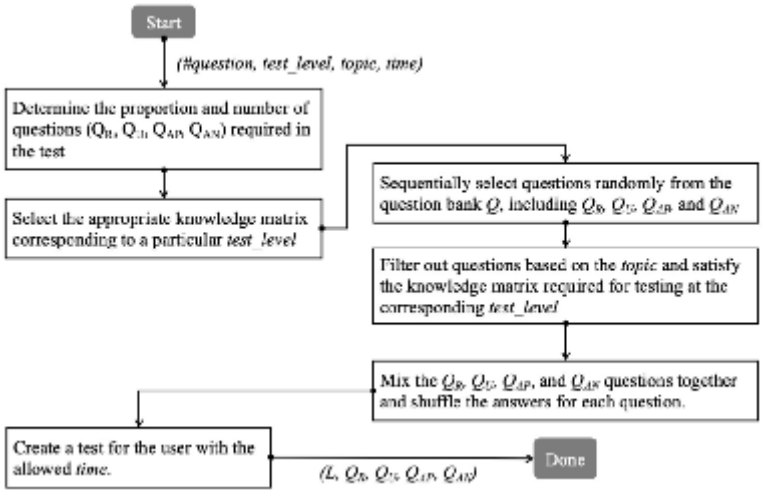

Figure 1. Algorithm of generating a test

3.2. The problem and algorithm of evaluating user knowledge through a test

3.2.1. Evaluating user's ability on topic

In this evaluation, the system must provide some statistical results about the topics that students have been tested on. The test can contain one or multiple topics and has a difficulty level of *test_level*. Based on each topic *t* in the test and questions about *t*, the system will assess the student's ability on topic *t* when taking the test at the difficulty level of *test_level*.

Definition 3.2: The knowledge gained by the student or learner on topic *t* in the subject of Data Structures and Algorithms, based on the test, is assessed according to the following formula:

$$P_t = \frac{r_t'}{r_t} \times m + \frac{u_t'}{u_t} \times n + \frac{ap_t'}{ap_t} \times p + \frac{an_t'}{an_t} \times q; \quad (1)$$

with 0< *m, n, p, q* < 1

Where P_t is the knowledge gained by the student on topic *t* in the subject of Data Structures and Algorithms based on the test the student has taken;

(m, n, p, q) are the proportions of questions corresponding to the levels of Bloom's taxonomy used in the paper, such as *Remembering*, *Understanding*, *Applying*, and *Analyzing*, on topic *t* in the test. These proportions are calculated based on the number of questions for each content area distributed in the knowledge matrix, as an example shown in Table 1;

r'_t, u'_t, ap'_t, an'_t are the numbers of questions on topic *t* that the student answered correctly in the test, corresponding to the cognitive levels *Remembering*, *Understanding*, *Applying*, and *Analyzing*, respectively;

r_t, u_t, ap_t, an_t are the numbers of questions on topic *t* in the test corresponding to the cognitive levels *Remembering*, *Understanding*, *Applying*, and *Analyzing*, respectively.

The knowledge gained by the student on topic *t* in the test is proposed to be evaluated by the system as follows:

(1) $P_t < 0.5$: the student is weak in topic *t* when taking the test in the subject of Data Structures and Algorithms at the *test_level*;

(2) $0.5 \le P_t \le 0.75$: the student demonstrates good knowledge of topic *t* when taking the test in the subject

of Data Structures and Algorithms at the *test_level*;

(3) $P_t > 0.75$: the student demonstrates excellent knowledge of topic *t* when taking the test in the subject of Data Structures and Algorithms at the *test_level*.

Definition 3.3: The problem of assessing students' or learners' knowledge in a subject of Data Structures and Algorithms based on the test is defined as follows:

(PROFILE, QUIZ) FEEDBACK

In which, *PROFILE* represents the information of the student or learner taking the test on the system; *QUIZ* is the set of questions in the test; *FEEDBACK* is the result and recommendation provided by the system to the student or learner based on formula (1).

With the knowledge organization *(T, Q, R, CLO)* defined in section 2.1, the test structure model as in definition 2.3, and formula (1), the algorithm of evaluating students' or learners' knowledge in the subject through a test is described as in Figure 2.

3.2.2. Evaluating user's ability on course learning outcome

A test can contain one or more learning outcomes and have varying difficulty levels. Based on each learning outcome (CLO) in the test defined in the knowledge matrix as example in Table 1, along with the questions related to these CLOs, the system will assess the students' or learners' ability to achieve the learning outcomes when taking the test with the specified difficulty level.

Definition 3.4: The learning outcomes in the subject of Data Structures and Algorithms that students or learners achieve through any given test are evaluated according to the following formula:

$$P_{clo} = \frac{r'_{clo}}{r_{clo}} \times m' + \frac{u'_{clo}}{u_{clo}} \times n' + \frac{ap'_{clo}}{ap_{clo}} \times p' + \frac{an'_{clo}}{an_{clo}} \times q'; \quad (2)$$

with $0 < m', n', p', q' < 1$

Where P_{clo} is the knowledge gained by the student on course learning outcome *clo* in the subject of Data Structures and Algorithms based on the test the student has taken;

(m', n', p', q') are the proportions of questions corresponding to the levels of Bloom's taxonomy used in the paper, such as *Remembering*, *Understanding*, *Applying*, and *Analyzing*, on course learning outcome *clo* in the test. These proportions are calculated based on the number of questions for each content area distributed in the knowledge matrix, as an example shown in Table 1;

r'_{clo}, u'_{clo}, ap'_{clo}, an'_{clo} are the numbers of questions on course learning outcome *clo* that the student answered correctly in the test, corresponding to the cognitive levels *Remembering*, *Understanding*, *Applying*, and *Analyzing*, respectively;

r_{clo}, u_{clo}, ap_{clo}, an_{clo} are the numbers of questions on course learning outcome *clo* in the test corresponding to the cognitive levels *Remembering*, *Understanding*, *Applying*, and *Analyzing*, respectively.

The knowledge gained by the student on course learning outcome *clo* in the test is proposed to be evaluated by the system as follows:

(1) $P_{clo} < 0.5$: students have not met the course learning outcome *clo* when taking the Data Structures and Algorithms course test at the *test_level*;

(2) $P_{clo} \geq 0.5$: students demonstrate good knowledge and meet the course learning outcome *clo* when taking the Data Structures and Algorithms course test at the *test_level*;

With the knowledge organization (T, Q, R, CLO) defined in section 2.1, the test structure model as in definition 2.3, and formula (2), the algorithm of evaluating students' or learners' ability on the course learning outcomes through a test is described as in Figure 3.

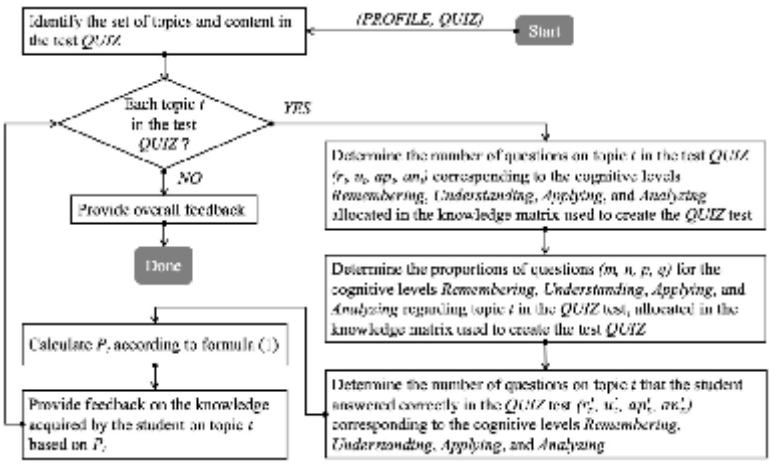

Figure 2. Algorithm of evaluating students' ability on topic

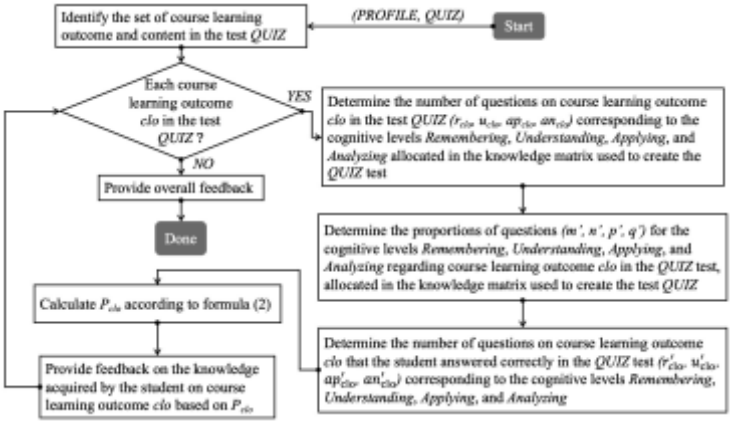

Figure 3. Algorithm of evaluating students' ability on course learning outcomes

4. EXPERIMENTAL RESULTS

Based on knowledge model for Data structures and Algorithms course in section 2 and model of problems in section 3, a knowledge assessment system for Data structures and Algorithms multiple choices test at Hong Bang International University has been developed. In this section, we present the architecture and knowledge base of the system, and our system evaluation.

4.1. Architecture and knowledge base of the system

The architecture for knowledge assessment system for the Data Structures and Algorithms is shown as in Figure 4. The system has two types of users: students and knowledge engineers. Students will use our program to support their studying by doing a test and following their developing knowledge. Knowledge Engineer can manage the knowledge base of the system through the user interface. The module (1) will classify the request from the user and distribute the request into other modules (3), (4), (5) approriately. The module (3) has functions to manage the knowledge such as insert, update, delete. The module (4) will evaluate the data structures and alogorithms knowledge through the testing processing. The module (5) is reponsible for generating a test according to specific requirements. The knowledge base stores the data structures and algorithms knowledge domain in SQL database.

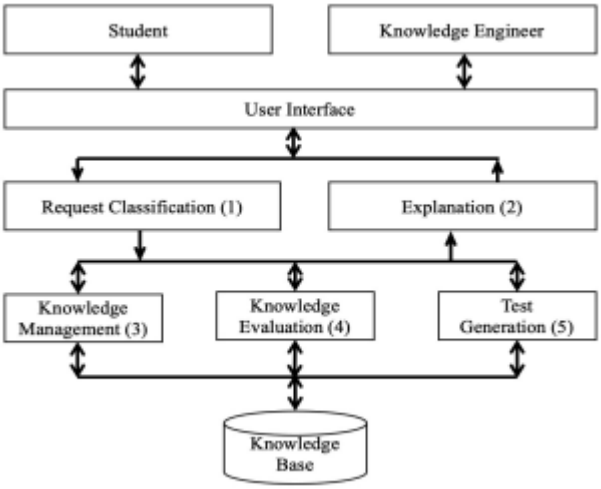

Figure 4. The architecture of knowledge assessment system

The topics of the subject of Data structures and algorithms are collected from textbooks and lectures used in the Data Structures and Algorithms course for Information Technology students at Hong Bang International University [14, 15]. Multiple questions are also developed from [14, 15]. These questions were divided into topics and determined their cognitive level according to Bloom's Taxonomy (*Remembering (R)*, *Understanding (U)*, *Applying (AP)*, *Analyzing (AN)*) by some of IT lecturers at Hong Bang International University. Currently, the reposistory of our system has about 225 questions. The number of questions in 6 chapters of our Data Structures and Algorithms course are presented in Table 3.

Table 3. The number of questions in 6 chapters of Data Structures and Algorithms course

	Data Structures and Algorithms Course Content	Number of Questions according to Bloom's Taxonomy Cognitive Levels				Total
		R	U	AP	AN	
1	*Chapter 1: Overview of Data Structures and Algorithms*	42	12	7	7	68
2	*Chapter 2: Searching and Sorting*	15	19	11	6	51
3	*Chapter 3: Linked List*	16	11	1	1	29
4	*Chapter 4: Stack, Queue*	10	3	5	6	24

	Data Structures and Algorithms Course Content	Number of Questions according to Bloom's Taxonomy Cognitive Levels				Total
		R	U	AP	AN	
5	*Chapter 5: Tree*	7	4	10	10	31
6	*Chapter 6: Hash table*	5	7	7	3	22
Total		**95**	**56**	**41**	**33**	**225**

4.2. System evaluation

Our knowledge assessment system can make a test according to specific requirements. This test satisfies requirements of students about the content, topics, and the level of hardness. Besides that, it can evaluate the level of the students' knowledge on each topic. From these results, the students will have a plan for self-studying to improve their weakness topics.

A. Functional Testing

1) Test-taking User Interface: As shown in Figure 5, this is the interface of our system when a student does a multiple choices test.

Figure 5. Test-taking User Interface

***2)* Test Result User Interface**: After finishing a test, our system will analysis the questions of this test by topics and show results of the diagnosing the ability of the student about each topic. As shown in Figure 6, the system shows student's assessment on the 6 topics of the Data Structures and Algorithms course.

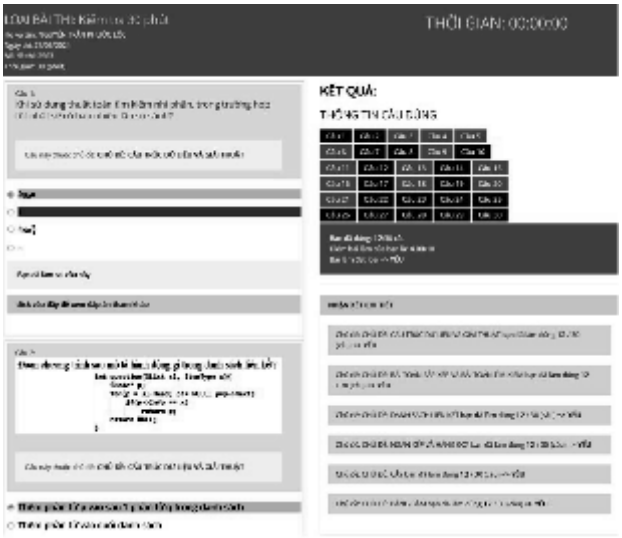

Figure 6. Test Result User Interface

B. Comparisions with Other Systems

Table 4 below shows a comparison between our system and other systems that support multiple-choice testing. The comparisons are based on the function about the generating a test, the evaluation to the student, and testing domain.

Table 4. A comparison between our system and other systems that support multiple-choice testing

	Program	The ability to generate test	The ability to give student assessment	Testing Domain
1	Intelligent Support System for the Knowledge evaluation in high-school mathematics [7]	- It can generate a test randomly with common parameters, such as: the time, the number of questions, the level of the difficulty. - It cannot create a test with the selected contents and topics.	- It can evaluate the level of the learner's knowledge on each topic through a test.	High-school mathematics
2	International English Test [8]	- Funtionalities are not public to user of the systems.	- It only evaluates based on the scores - It does not analysis how to get those assessments.	English
3	VioEdu [9]	- Funtionalities are not public to user of the systems.	- It can evaluate the level of the learner's knowledge on each topic through a test.	Elementary and High-school mathematics
4	HocMai [10]	- It has available tests which were made by experienced lecturers. - It cannot make a test automatically.	- It only evaluates based on the scores - It does not analysis how to get those assessments.	High-school mathematics

	Program	The ability to generate test	The ability to give student assessment	Testing Domain
5	Elearning [11]	- Lectures are able to create a test but they have to import or create questions manually for a particular test. - It cannot create a test with the selected contents and topics.	- It can only give the scores - It does not analysis how to get those assessments.	Multiple domains
6	Our system	- It can generate a test randomly with parameters such as: number of question, test level, topic, and time. - It can create a test with selected topics.	- It can evaluate the level of the learner's knowledge on each topic through a test.	Data structures and Algorithms

5. CONCLUSION

The paper presented the design of techniques to develop an application for assessing knowledge in the subject of Data Structures and Algorithms. The proposed system supports creating objective multiple-choice test according to the course content and the course syllabus of the Data Structures and Algorithms subject at Hong Bang International University. The system is able to evaluate level of students' knowledge about the subject through classical test theory. This system can also support the training through making a test satisfying the requirements from students. However, the impact of the proposed system on the learning process of students in the subject requires further experimentation and research. In the future, question banks for courses need to be reviewed and supplemented. Based on multiple tests that students have completed, the system can provide feedback on students' knowledge of specific topics in the course content through the entire process of multiple test-taking. We also need to develop a function to remove or revise the question or answer for improvement. Additionally, we needs to further research modern theoretical testing, also known as Item Response Theory [12], to have a more objective view of the difficulty of questions and the competence of students.

ACKNOWLEDGEMENT

This work is funded by Hong Bang International University under grant code GVTC17.31.

REFERENCES

[1] Lin, C. C., Huang, A. Y., & Lu, O. H., "Artificial intelligence in intelligent tutoring systems toward sustainable education: a systematic review", *Smart Learning Environments*, 10(1), 41, 2023.

[2] Kurni, M., Mohammed, M. S., & Srinivasa, K. G., "Intelligent tutoring systems". In *A beginner's guide to introduce artificial intelligence in teaching and learning* (pp. 29-44). Cham: Springer International Publishing, 2023.

[3] E. Mousavinasab, N. Zarifsanaiey, S. R. Niakan Kalhori, M. Rakhshan, L. Keikha, and M. Ghazi Saeedi, "Intelligent tutoring systems: A systematic review of characteristics, applications, and evaluation methods", Interact. Learn. Environ., pp. 1–22, Dec. 2018, doi: 10.1080/10494820.2018.1558257.

[4] B. D. Nye, "Intelligent tutoring systems by and for the developing world: A review of trends and approaches for educational technology in a global context," *Int. J. Artif. Intell. Educ.*, vol. 25, no. 2, pp. 177–203, Jun. 2015.

[5] Nguyen, H. D., "Intelligent System in Education: Requirements and Design Method", *Journal of Electronic Voltage and Application*, 4(2), 12-19, 2023.

[6] Hien D. Nguyen, Nhon V. Do, Thanh T. Mai, Vuong T. Pham, "*A method for designing the Intelligent system in learning of Algorithms*", *Proceedings of 18th International Conference on Intelligent Software Methodologies, Tools, and Techniques (SOMET 2019)*, Kuching, Malaysia, *Frontiers in Artificial Intelligence and Applications*, vol. 318, pp. 658 - 671, Sept. (2019).

[7] Thanh T. Mai, Hien D. Nguyen, Trung T. Le, Vuong T. Pham, "*An Intelligent Support System for the Knowledge evaluation in high-school mathematics by Multiple choices testing"*, The 5th NAFOSTED Conference on Information and Computer Science (NICS 2018), pp. 284-289, Ho Chi Minh city, Vietnam, Nov. 2018.

[8] International English Test, "International English Test". [Online]. Available: https://internationalenglishtest.com/. [Accessed 25 05 2024].

[9] FPT, "VioEdu". [Online]. Available: https://vio.edu.vn/. [Accessed 25 05 2024]

[10] Hệ thống Giáo dục HocMai, "HocMai". [Online]. Available: https://hocmai.vn/. [Accessed 25 05 2024].

[11] Hong Bang International University, "Elearning". [Online]. Available: https://elearning.hiu.vn/. [Accessed 25 05 2024].

[12] Lâm Quang Thiệp, "Đo lường trong giáo dục - lí thuyết và ứng dụng", NXB Đại học Quốc gia Hà Nội, 2010.

[13] Trịnh Thanh Đèo, "Tài liệu tập huấn Công tác xây dựng và đánh giá chất lượng ngân hàng đề thi/câu hỏi thi tại Trường Đại học Quốc tế Hồng Bàng", Khóa tập huấn Xây dựng và đánh giá chất lượng ngân hàng đề thi/câu hỏi thi, 03/2022.

[14] Đỗ Văn Nhơn, Trịnh Quốc Sơn, "Giáo trình Cấu trúc dữ liệu & Giải thuật", NXB ĐHQG TP. HCM, 2015.

[15] Hoàng Ngọc Long, "Lectures of Data Structures and Algorithms". [Online]. Available: https://www.youtube.com/playlist?list=PLrcfrbKhqmAWrIoc47jVx1eEzS9gJUdg_ . [Accessed 25 05 2024].

Thiết kế hệ thống hỗ trợ kiểm tra đánh giá kiến thức môn Cấu trúc dữ liệu và Giải thuật

Hoàng Ngọc Long, Mai Trung Thành và Đỗ Văn Nhơn

TÓM TẮT

Môn học Cấu trúc dữ liệu và Giải thuật là một trong những môn cơ bản và quan trọng trong chương trình đào tạo ngành Công nghệ thông tin. Môn học đóng vai trò là nền tảng cho việc hiểu và áp dụng các cấu trúc dữ liệu và thuật toán trong việc giải quyết các vấn đề thực tế và phức tạp. Tuy nhiên, việc giảng dạy và học Cấu trúc dữ liệu và Giải thuật thường đối mặt với nhiều thách thức. Sinh viên thường gặp khó khăn trong việc hiểu và áp dụng các khái niệm, thuật toán và cấu trúc dữ liệu vào các bài toán thực tế. Đồng thời, việc đánh giá kiến thức và hiệu suất học tập của sinh viên trong môn học cũng đòi hỏi sự chính xác và khách quan. Trong bối cảnh này, việc thiết kế và triển khai một hệ thống hỗ trợ kiểm tra và đánh giá kiến thức cho môn Cấu trúc dữ liệu và Giải thuật trở nên thật sự cần thiết. Một hệ thống như vậy không chỉ giúp sinh viên nắm vững kiến thức một cách hiệu quả mà còn tạo điều kiện cho giảng viên theo dõi tiến trình học tập của sinh viên và cung cấp phản hồi để họ có thể cải thiện kết quả học tập. Bài báo sẽ trình bày về thiết kế các kỹ thuật để xây dựng ứng dụng kiểm tra đánh giá kiến thức môn Cấu trúc dữ liệu và Giải thuật. Ứng dụng này sẽ là một công cụ hữu ích cho giảng viên và sinh viên, từ đó nâng cao chất lượng giảng dạy và học tập các môn học ngành Công nghệ thông tin.

Từ khóa: *giáo dục điện tử, E-learning, intelligent software, intelligent tutoring system*

Received: 15/05/2024
Revised: 11/06/2024
Accepted for publication: 13/06/2024