

Xây dựng ứng dụng kiểm tra đánh giá kiến thức môn học Cấu trúc dữ liệu và Giải thuật

Hoàng Ngọc Long^{*}, Đỗ Văn Nhon
Trường Đại học Quốc tế Hồng Bàng

TÓM TẮT

Môn học Cấu trúc dữ liệu và Giải thuật đóng vai trò nền tảng trong chương trình đào tạo Công nghệ thông tin, giúp sinh viên hiểu và áp dụng các thuật toán vào bài toán thực tế. Tuy nhiên, việc giảng dạy và học tập môn này gặp nhiều thách thức do sinh viên 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. Trước thực tế đó, bài báo đề xuất thiết kế và triển khai một hệ thống hỗ trợ kiểm tra, đánh giá kiến thức nhằm nâng cao hiệu quả học tập và giảng dạy. Hệ thống này không chỉ giúp sinh viên củng cố kiến thức, mà còn hỗ trợ giảng viên theo dõi tiến độ học tập và cung cấp phản hồi kịp thời, góp phần nâng cao chất lượng đào tạo ngành Công nghệ thông tin.

Từ khoá: biểu diễn tri thức, cấu trúc dữ liệu và giải thuật, giáo dục điện tử, hệ thống kiểm tra và đánh giá, hệ thống hỗ trợ học tập thông minh

1. TỔNG QUAN VỀ HỆ HỖ TRỢ KIỂM TRA ĐÁNH GIÁ KIẾN THỨC

Hệ thống hỗ trợ học tập thông minh (*Intelligent Tutoring Systems - ITS*) là một hệ thống phần mềm giáo dục dựa trên máy tính, ứng dụng trí tuệ nhân tạo nhằm nâng cao hiệu quả giảng dạy và học tập. Các hệ thống này hỗ trợ quyết định trong giáo dục, quản lý tri thức giảng dạy, theo dõi hoạt động học tập và đánh giá mức độ tiếp thu kiến thức của sinh viên [1]. Nhờ đó, ITS trở thành một công cụ phổ biến trong giáo dục, góp phần cải thiện đáng kể quá trình học tập của sinh viên [2 - 4].

Kiểm tra, đánh giá kiến thức là một thành phần không thể thiếu trong ITS. Mục tiêu của quá trình này là giúp sinh viên hiểu rõ mức độ tiếp thu kiến thức và xác định những nội dung còn hạn chế. Các nghiên cứu [1 - 4] đã chỉ ra rằng ITS hỗ trợ hiệu quả việc tiếp thu kiến thức trong nhiều lĩnh vực. Tuy nhiên, vẫn chưa có một giải pháp hoàn chỉnh cho việc kiểm tra, đánh giá môn học thuộc nhóm ngành Công nghệ thông tin. Nghiên cứu [5] đề xuất một hệ thống giáo dục thông minh với hai chức năng chính: truy vấn kiến thức và đánh giá trình độ người học thông qua bài kiểm tra trắc nghiệm. Hệ thống này sử dụng mô hình tích hợp ontology và đồ thị tri thức để thiết kế cơ sở dữ liệu, bộ suy luận và hệ thống theo dõi. Tuy nhiên, nó chưa áp dụng lý thuyết khảo thí để xây dựng câu hỏi thi hoặc ngân

hàng câu hỏi, đồng thời không đề cập cụ thể đến lĩnh vực ứng dụng. Nghiên cứu [6] trình bày hệ thống minh họa thuật toán tự động thông minh cho các môn học như Cấu trúc dữ liệu và Giải thuật, Lý thuyết đồ thị, v.v. Hệ thống hỗ trợ hai nhóm người dùng: Người học và nhà quản trị tri thức. Nhà quản trị có thể nhập thuật toán vào hệ thống, trong khi người học có thể theo dõi quá trình minh họa thuật toán từng bước. Tuy nhiên, hệ thống này chưa tích hợp chức năng kiểm tra, đánh giá kiến thức. Công trình [7] đề xuất hệ thống hỗ trợ đánh giá kiến thức tự động cho toán trung học phổ thông, ứng dụng hệ chuyên gia để giám sát và đánh giá năng lực học tập của sinh viên theo nội dung kiến thức. Nghiên cứu [8] đã đề xuất hệ thống kiểm tra, đánh giá trắc nghiệm kiến thức môn học Cấu trúc dữ liệu và Giải thuật, tập trung vào việc tạo đề kiểm tra theo yêu cầu và đánh giá kiến thức của sinh viên qua từng bài kiểm tra. Tuy nhiên, công trình này chưa đưa ra giải pháp đánh giá quá trình học của sinh viên dựa trên toàn bộ các bài kiểm tra. Ngoài ra, một số ứng dụng thương mại như *International English Test* [9] hoặc các nền tảng giáo dục như *VioEdu* và *HocMai* [10, 11] đã được triển khai để hỗ trợ đánh giá kiến thức trong các môn học phổ thông. Tuy nhiên, vẫn chưa có ứng

Tác giả liên hệ: Hoàng Ngọc Long

Email: longhn@hiu.vn

dụng cụ thể nào hỗ trợ đánh giá kiến thức trong lĩnh vực Công nghệ thông tin, đặc biệt là môn Cấu trúc dữ liệu và Giải thuật. Các hệ thống quản lý học tập (*Learning Management Systems* - LMS), như *E-learning* [12] dựa trên nền tảng nguồn mở Moodle, có hỗ trợ kiểm tra đánh giá dựa trên dữ liệu được nhập vào hệ thống. Tuy nhiên, các hệ thống này chỉ cung cấp bài kiểm tra mà không có cơ chế đánh giá chi tiết kết quả của người học. Do đó, vẫn chưa có giải pháp hoàn chỉnh cho hệ thống kiểm tra, đánh giá kiến thức môn Cấu trúc dữ liệu và Giải thuật.

Trong bài báo này, chúng tôi đề xuất mô hình tổ chức cơ sở tri thức kiểm tra đánh giá cho môn học Cấu trúc dữ liệu và Giải thuật. Hệ thống được xây dựng dựa trên mô hình này sẽ hỗ trợ tạo đề kiểm tra trắc nghiệm theo chủ đề hoặc chuẩn đầu ra, đảm bảo phù hợp với trình độ sinh viên. Dựa trên các kỹ thuật phân tích và thống kê, hệ thống có thể tự động đánh giá năng lực người học, cung cấp thông tin định tính và định lượng, đồng thời đưa ra nhận xét, khuyến nghị giúp sinh viên củng cố kiến thức và nâng cao kỹ năng lập trình. Bài báo này là phiên bản mở rộng của công trình [8] mà chúng tôi đã công bố. Các cải tiến chính bao gồm: (i) tổng quát hóa thuật toán đánh giá kiến thức của sinh viên qua từng bài kiểm tra; (ii) đề xuất thuật toán đánh giá tiến trình học của sinh viên qua toàn bộ các bài kiểm tra; (iii) bổ sung số lượng câu hỏi cho môn học để cải thiện độ chính xác trong đánh giá.

2. MÔ HÌNH BIỂU DIỄN TRI THỨC KIỂM TRA ĐÁNH GIÁ MÔN HỌC

Trong việc thiết kế hệ thống hỗ trợ kiểm tra và đánh giá kiến thức cho môn học Cấu trúc dữ liệu và Giải thuật theo hình thức trắc nghiệm khách quan, việc xây dựng một mô hình biểu diễn tri thức là một phần quan trọng nhằm đảm bảo tính linh hoạt, hiệu quả và tính đa dạng trong quá trình kiểm tra và đánh giá. Mô hình biểu diễn tri thức này cần phải: i) đảm bảo biểu diễn được khái niệm, kiến thức liên quan đến môn học; ii) có cấu trúc tổ chức, xây dựng ngân hàng câu hỏi trắc nghiệm; iii) có mô hình, liên kết được tri thức môn học và ngân hàng câu hỏi trắc nghiệm.

2.1. Mô hình cơ sở tri thức môn học Cấu trúc dữ liệu và Giải thuật

2.1.1. Tổ chức cơ sở tri thức cho nội dung kiến thức môn học Cấu trúc dữ liệu và Giải thuật

Định nghĩa 2.1: Mô hình tổ chức cơ sở tri thức để

biểu diễn nội dung kiến thức môn học Cấu trúc dữ liệu và Giải thuật có dạng:

$$(T, Q, R, CLO)$$

Trong đó, T là tập hợp các tiêu đề trong môn học và có cấu trúc cây; Q là tập hợp các câu hỏi trắc nghiệm của môn học; R là tập hợp mối quan hệ giữa các tiêu đề trong môn học và các câu hỏi trắc nghiệm của môn học; CLO là tập hợp các chuẩn đầu ra của môn học. Với mỗi câu hỏi $q \in Q$ sẽ có cấu trúc:

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

Trong mô hình này, *question_content* là nội dung của câu hỏi; *answers* là tập hợp các đáp án lựa chọn; *correct_answers* là tập các đáp án đúng ($correct_answers \in answers$); *difficult_level* là độ khó của câu hỏi, được tính theo lý thuyết trắc nghiệm cổ điển [13]; *discrimination_level* là độ phân biệt của câu hỏi, phản ánh khả năng phân loại sinh viên theo mức độ năng lực [13]; *bloom_level* là mức độ nhận thức của câu hỏi theo thang Bloom, tương ứng với chuẩn đầu ra kiến thức môn học [13]. Bài báo sử dụng 4 mức của thang Bloom như *Nhớ (Remembering - R)*, *Hiểu (Understanding - U)*, *Áp dụng (Applying - AP)*, *Phân tích (Analyzing - AN)*. Mô hình cơ sở tri thức này đã được áp dụng để tổ chức ngân hàng câu hỏi trắc nghiệm khác nhau. Ngân hàng câu hỏi này được sử dụng để tạo ra một bài kiểm tra phù hợp với yêu cầu, năng lực của sinh viên và kết quả của bài kiểm tra này là cơ sở để chẩn đoán trình độ kiến thức của sinh viên.

2.1.2. Mô hình của người dùng

Hệ thống đề xuất có thể theo dõi quá trình học của sinh viên thông qua các bài kiểm tra. Hệ thống thu thập nội dung và kết quả từ các bài kiểm tra của người dùng. Dựa trên những thông tin này, hệ thống có thể đánh giá sự phát triển kiến thức của sinh viên.

Định nghĩa 2.2: Mô hình của người dùng trong hệ thống có cấu trúc như sau:

$$(PROFILE, TEST_LIST)$$

Trong đó, *PROFILE* là thông tin của người dùng hệ thống; *TEST_LIST* là tập hợp danh sách các bài kiểm tra mà người dùng đã thực hiện. Với mỗi bài kiểm tra $test \in TEST_LIST$ sẽ có cấu trúc:

$$(t, cq, icq, m)$$

Trong mô hình này, t là bài kiểm tra mà người dùng

đã thực hiện; cq là tập hợp các câu hỏi trong bài kiểm tra t mà người dùng có câu trả lời đúng; icq là tập hợp các câu hỏi trong bài kiểm tra t mà người dùng có câu trả lời sai; m là điểm của bài kiểm tra t ; các câu hỏi cq và icq có cấu trúc tương tự như các câu hỏi $q \in Q$.

2.2. Mô hình ngân hàng câu hỏi trắc nghiệm và liên kết tri thức môn học

Một đề kiểm tra thường được thiết kế bởi các câu hỏi chọn từ một ngân hàng câu hỏi trắc nghiệm. Ngân hàng câu hỏi là tập hợp một số lượng tương đối lớn các câu hỏi, trong đó mỗi câu hỏi được mô tả gắn với các phần nội dung xác định và các tham số của nó, trong trường hợp câu hỏi trắc nghiệm đó là độ khó (*difficult_level*), độ phân biệt (*discrimination_level*) theo lý thuyết trắc nghiệm cổ điển [13]. Ngân hàng câu hỏi phải được thiết kế sao cho trên đó có thể thực hiện các thao tác loại trừ hoặc thay đổi các câu xấu, bổ sung các câu tốt để số lượng và chất lượng các câu hỏi ngày càng tăng. Có thể mô tả tóm tắt các bước của quy trình thiết kế một đề kiểm tra tiêu chuẩn hóa và một ngân hàng câu hỏi cho môn học Cấu trúc dữ liệu và Giải thuật như sau [13]:

- (1) Xây dựng ma trận kiến thức của môn Cấu trúc dữ liệu và Giải thuật;
- (2) Phân công cho các giảng viên, mỗi người chế tác một số câu hỏi theo các yêu cầu gắn với các ô ma trận kiến thức như ví dụ ở Bảng 1;
- (3) Trao đổi các câu hỏi trong nhóm đồng nghiệp,

tổ chức đọc duyệt, biên tập và đưa các câu hỏi lưu vào các kho dữ liệu trong máy tính. Bước này sẽ thu được một tập hợp các câu hỏi trắc nghiệm đã được chỉnh sửa công phu lưu trong máy tính. Tuy nhiên, đây chưa phải là ngân hàng câu hỏi vì các câu hỏi chưa được thử nghiệm để xác định tham số độ khó và độ phân biệt;

(4) Lập các đề kiểm tra thử và tổ chức trắc nghiệm thử trên các nhóm thí sinh đại diện cho tổng thể đối tượng sẽ được kiểm tra;

(5) Chấm và phân tích thống kê các kết quả trắc nghiệm thử. Quá trình phân tích thống kê và định cỡ câu hỏi trắc nghiệm sẽ cho hai loại kết quả: Một là cung cấp các tham số của câu hỏi trắc nghiệm, hai là phát hiện các câu hỏi có chất lượng kém.

(6) Xử lý các câu hỏi chất lượng kém: Hoặc là sửa đổi tu chỉnh, hoặc là loại bỏ nếu chất lượng quá kém không thể sửa đổi được. Các câu hỏi được tu chỉnh xong lại được đưa vào kho lưu trữ. Qua bước này một ngân hàng câu hỏi bắt đầu hình thành. Việc tổ chức trắc nghiệm thử và chỉnh sửa các câu hỏi trắc nghiệm có thể tổ chức rất nhiều lần, qua mỗi lần một số câu hỏi trong ngân hàng câu hỏi được chỉnh sửa, hoàn thiện và ngân hàng câu hỏi được bổ sung.

(7) Khi đã yên tâm về số lượng và chất lượng các câu hỏi trong ngân hàng câu hỏi có thể thiết kế các đề kiểm tra cho các kỳ thi chính thức. Cấu trúc của một đề kiểm tra chính thức phải được thể hiện bằng một ma trận kiến thức tương ứng.

Bảng 1. Ma trận kiến thức của một đề kiểm tra cuối môn Cấu trúc dữ liệu và Giải thuật ở Mức 1

Chuẩn đầu ra môn học	Nội dung môn học	Mức độ nhận thức theo thang Bloom				Tổng
		R	U	AP	AN	
CLO1	Chương 1. Tổng quan thuật toán và cấu trúc dữ liệu	5	3	1	1	10
CLO2, CLO3	Chương 2. Tìm kiếm và sắp xếp	4	3	1	1	9
CLO4, CLO5	Chương 3. Danh sách liên kết	4	3	1	1	9
CLO4, CLO5	Chương 4. Ngăn xếp, hàng đợi	4	3	1	1	9
CLO4, CLO5	Chương 5. Cây nhị phân tìm kiếm	4	2	1	1	8
CLO4, CLO5	Chương 6. Bảng băm	4	1	0	0	5
Tổng số câu		25	15	5	5	50
Tỷ lệ		50%	30%	10%	10%	100%

Định nghĩa 2.3: Mô hình cho cấu trúc bài kiểm tra có dạng:

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

Trong đó, L là mức độ khó của bài kiểm tra; $Q_R, Q_U,$

Q_{AP}, Q_{AN} là tập hợp các câu hỏi theo thang đo Bloom lần lượt với các mức độ *Nhớ*, *Hiểu*, *Áp dụng* và *Phân tích*.

Mức độ khó của một bài kiểm tra có thể được thiết

kế dựa trên tỷ lệ các câu hỏi theo thang đo Bloom, mức độ khó (p) của các câu hỏi và mức độ phân biệt (d) của các câu hỏi như Bảng 2. Độ khó của câu hỏi được tính toán theo lý thuyết khảo thí cổ điển và có thể được phân loại lần lượt theo các mức: Câu hỏi dễ ($difficult_level \geq 0.7$), câu hỏi có độ khó vừa phải ($0.6 \leq difficult_level < 0.7$), câu hỏi có độ

khó trung bình ($0.4 \leq difficult_level < 0.6$), câu hỏi tương đối khó ($0.3 \leq difficult_level < 0.4$), câu hỏi khá khó ($difficult_level < 0.3$) [13, 14]. Độ phân biệt của câu hỏi tốt nằm trong khoảng $[0.2 - 1]$ và cũng được tính dựa trên lý thuyết khảo thí cổ điển [13, 14]. Nếu $discrimination_level < 0.2$, thì câu hỏi kém, cần phải loại bỏ hoặc sửa lại cho tốt hơn.

Bảng 2. Mức độ khó của bài kiểm tra môn Cấu trúc dữ liệu & Giải thuật

Mức độ khó của bài kiểm tra	Tỷ lệ các câu hỏi theo mức độ nhận thức theo thang Bloom				Độ khó câu hỏi (p)	Độ phân biệt (d)
	Nhớ	Hiểu	Áp dụng	Phân tích		
Mức 1 (Dễ)	50%	30%	10%	10%	$p \geq 0.7$	$0.2 \leq d \leq 1$
Mức 2 (Trung Bình)	40%	30%	20%	10%	$0.6 \leq p < 0.7$	$0.2 \leq d \leq 1$
Mức 3 (Khá khó)	30%	40%	20%	10%	$0.4 \leq p < 0.6$	$0.2 \leq d \leq 1$
Mức 4 (Khó)	20%	30%	30%	20%	$0.3 \leq p < 0.4$	$0.2 \leq d \leq 1$
Mức 5 (Rất khó)	20%	20%	30%	30%	$p < 0.3$	$0.2 \leq d \leq 1$

3. BÀI TOÁN KIỂM TRA ĐÁNH GIÁ VÀ THIẾT KẾ CÁC THUẬT GIẢI

Dựa vào các định nghĩa 2.1, 2.2, 2.3, hệ thống hỗ trợ kiểm tra đánh giá kiến thức môn học Cấu trúc dữ liệu và Giải thuật cần phải giải quyết các vấn đề sau:

- (1) Tạo đề kiểm tra theo yêu cầu: Hệ thống có thể tạo đề theo số lượng câu hỏi mong muốn, mức độ khó của bài kiểm tra và chủ đề quan tâm trong môn học;
- (2) Đánh giá kiến thức của sinh viên theo từng bài kiểm tra: Dựa vào kết quả bài kiểm tra mà sinh viên đã thực hiện, hệ thống có thể đưa ra phản hồi kết quả và khả năng của sinh viên ở mỗi chủ đề hoặc theo chuẩn đầu ra môn học.
- (3) Đánh giá tiến trình học của sinh viên: Dựa vào tất cả các bài kiểm tra mà sinh viên, người học đã thực hiện, hệ thống có thể đưa ra nhận xét về kiến thức đạt được của sinh viên, người học ở mỗi chủ đề hoặc theo mỗi chuẩn đầu ra môn học qua các lần kiểm tra ở mức độ khó cụ thể.

3.1. Bài toán tạo đề theo yêu cầu và thuật giải

Định nghĩa 3.1: Bài toán tạo đề kiểm tra theo yêu cầu được định nghĩa như sau:

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

Trong đó, $\#question$ là số lượng câu hỏi trong bài kiểm tra; $test_level$ là mức độ khó của bài kiểm tra, được quy định trong Bảng 2; $topic$ là tập hợp các chủ đề trong môn học cần đánh giá ($topic \subseteq T$); $(L, Q_R, Q_U, Q_{AP}, Q_{AN})$ là các thành phần được định nghĩa ở 2.3, và được xác định dựa trên bộ tham số $(\#question, test_level, topic)$; $time$ là thời gian làm bài kiểm tra, tính theo phút.

Với tổ chức tri thức (T, Q, R, CLO) được định nghĩa ở 2.1 và điều kiện các tham số để tạo đề kiểm tra $(\#question, test_level, topic, time)$ như định nghĩa 3.1, thuật giải tạo đề kiểm tra môn học Cấu trúc dữ liệu và Giải thuật được mô tả như sau:

Giải thuật 3.1: Tạo đề kiểm tra môn học Cấu trúc dữ liệu và Giải thuật	
Input:	$(\#question, test_level, topic, time)$
Output:	$(L, Q_R, Q_U, Q_{AP}, Q_{AN})$
Bước 1:	Từ $\#question$ và $test_level$ và dựa vào Bảng 2, xác định tỷ lệ và số lượng các câu hỏi (Q_R, Q_U, Q_{AP}, Q_{AN}) cần có trong đề kiểm tra.
Bước 2:	Chọn ma trận kiến thức phù hợp với hình thức, nội dung kiểm tra ở mức $test_level$ ví dụ như ở Bảng 1
Bước 3:	Lần lượt chọn các câu hỏi theo trình tự ngẫu nhiên Q_R, Q_U, Q_{AP}, Q_{AN} có trong ngân hàng câu hỏi Q , lọc ra các câu hỏi theo chủ đề $topic$ và thoả mãn ma trận kiến thức cần kiểm tra ở mức $test_level$ tương ứng.
Bước 4:	Trộn các câu hỏi Q_R, Q_U, Q_{AP}, Q_{AN}
Bước 5:	Tạo bài kiểm tra cho người dùng với thời gian $time$ cho phép.

3.2. Bài toán đánh giá kiến thức sinh viên, người học qua từng bài kiểm tra và thuật giải

Trong bài toán đánh giá này, hệ thống phải cung cấp một số kết quả thống kê về các chủ đề hoặc chuẩn đầu ra môn học mà sinh viên, người học đã thực hiện ở một kiểm tra bất kỳ. Bài kiểm tra có thể chứa một hoặc nhiều chủ đề, chuẩn đầu ra và có mức độ khó *test_level*. Dựa trên mỗi chủ đề *t*, chuẩn đầu ra *clo* trong bài kiểm tra, cùng với các câu hỏi về *t* hoặc *clo*, hệ thống sẽ đánh giá khả năng của sinh viên, người học về chủ đề *t* hoặc chuẩn đầu ra *clo* khi thực hiện bài kiểm tra có mức độ khó *test_level*.

Định nghĩa 3.2: Kiến thức đạt được của sinh viên, người học về chủ đề *t* hoặc chuẩn đầu ra *clo* trong môn học được gọi là P_x , với *x* là *t* hoặc *clo*, dựa theo một bài kiểm tra bất kỳ được đánh giá theo công thức sau:

$$P_x = \frac{r'_x}{r_x} \times m + \frac{u'_x}{u_x} \times n + \frac{ap'_x}{ap_x} \times p + \frac{an'_x}{an_x} \times q; \text{ với } 0 < m, n, p, q < 1 \quad (1)$$

Trong đó, P_x là kiến thức đạt được của sinh viên về chủ đề *t* gọi là P_t hoặc theo chuẩn đầu ra *clo* gọi là P_{clo} trong môn Cấu trúc dữ liệu và Giải thuật dựa theo bài kiểm tra mà sinh viên đã thực hiện; (*m*, *n*, *p*, *q*) là tỷ lệ các câu hỏi lần lượt theo các mức độ nhận thức của thang Bloom mà bài báo sử dụng như *Nhớ*, *Hiểu*, *Áp dụng*, *Phân tích* về chủ đề *t* hoặc chuẩn đầu ra *clo* trong bài kiểm tra. Tỷ lệ này được tính dựa số câu hỏi theo nội dung kiểm tra được phân bố ở ma trận kiến thức, như ở ví dụ Bảng 1; r'_x , u'_x , ap'_x , an'_x lần lượt là số câu hỏi về chủ đề *t* hoặc chuẩn đầu ra *clo* mà sinh viên trả lời đúng trong bài kiểm tra tương ứng với các mức nhận thức *Nhớ*, *Hiểu*, *Áp dụng*, *Phân tích*; r_x , u_x , ap_x , an_x lần lượt là số câu hỏi về chủ đề *t* hoặc chuẩn đầu ra *clo* trong bài kiểm tra tương ứng với các mức nhận thức *Nhớ*, *Hiểu*, *Áp dụng*, *Phân tích*.

Kiến thức đạt được của sinh viên về chủ đề *t* trong bài kiểm tra được hệ thống đề xuất đánh giá như sau:

- (1) $P_t < 0.5$: sinh viên còn yếu về chủ đề *t* khi làm bài kiểm tra môn học Cấu trúc dữ liệu và Giải thuật ở mức *test_level*.
- (2) $0.5 \leq P_t \leq 0.75$: sinh viên thể hiện có kiến thức tốt về chủ đề *t* khi làm bài kiểm tra môn học Cấu trúc dữ liệu và Giải thuật ở mức *test_level*.
- (3) $P_t > 0.75$: sinh viên thể hiện có kiến thức rất tốt về chủ đề *t* khi làm bài kiểm tra môn học Cấu trúc dữ liệu và Giải thuật ở mức *test_level*.

Kiến thức đạt được của sinh viên theo chuẩn đầu ra *clo* được hệ thống đề xuất đánh giá như sau:

- (1) $P_{clo} < 0.5$: sinh viên chưa đạt chuẩn đầu ra *clo* khi làm bài kiểm tra môn học Cấu trúc dữ liệu và Giải thuật ở mức độ khó *test_level*.
- (2) $P_{clo} \geq 0.5$: sinh viên thể hiện có kiến thức tốt và đạt chuẩn đầu ra *clo* khi làm bài kiểm tra môn học Cấu trúc dữ liệu và Giải thuật ở mức độ khó *test_level*.

Định nghĩa 3.3: Bài toán đánh giá kiến thức môn học của sinh viên, người học qua từng bài kiểm tra theo chủ đề *t* hoặc chuẩn đầu ra *clo* được định nghĩa như sau:

(*PROFILE, QUIZ*)

Trong đó, *PROFILE* là thông tin của sinh viên, người học làm bài kiểm tra trên hệ thống; *QUIZ* là tập hợp các câu hỏi trong bài kiểm tra; *FEEDBACK* là kết quả và phản hồi mà hệ thống khuyến nghị cho sinh viên, người học dựa vào công thức (1).

Với tổ chức tri thức (*T, Q, R, CLO*) được định nghĩa ở 2.1, mô hình cấu trúc bài kiểm tra như ở định nghĩa 2.3, và công thức (1), bài toán đánh giá kiến thức sinh viên, người học qua từng bài kiểm tra theo chủ đề môn học hoặc chuẩn đầu ra môn học được mô tả như ở thuật giải sau:

Giải thuật 3.2: Đánh giá kiến thức sinh viên qua từng bài kiểm tra theo chủ đề hoặc chuẩn đầu ra môn học	
Input:	(<i>PROFILE, QUIZ</i>)
Output:	P_x theo công thức (1) và phản hồi về kiến thức đạt được của sinh viên, người học về chủ đề <i>t</i> hoặc chuẩn đầu ra <i>clo</i> trong bài kiểm tra <i>QUIZ</i> .
Bước 1:	Xác định tập các chủ đề <i>topic</i> , tập các chuẩn đầu ra <i>clo</i> , nội dung kiểm tra trong <i>QUIZ</i> .
Bước 2:	Với mỗi chủ đề <i>t</i> hoặc chuẩn đầu ra <i>clo</i> trong bài kiểm tra <i>QUIZ</i> :
Bước 2.1:	Xác định số câu hỏi về chủ đề <i>t</i> hoặc chuẩn đầu ra <i>clo</i> trong bài kiểm tra (r_x, u_x, ap_x, an_x) tương ứng với các mức nhận thức <i>Nhớ</i> , <i>Hiểu</i> , <i>Áp dụng</i> , <i>Phân tích</i> được phân bố ở ma trận kiến thức tạo ra bài kiểm tra <i>QUIZ</i> .

Bước 2.2:	Xác định các tỷ lệ các câu hỏi (m, n, p, q) theo các mức độ nhận thức <i>Nhớ, Hiểu, Áp dụng, Phân tích</i> về chủ đề t hoặc chuẩn đầu ra clo trong bài kiểm tra được phân bố ở ma trận kiến thức tạo ra bài kiểm tra <i>QUIZ</i> .
Bước 2.3:	Xác định số câu hỏi về chủ đề t hoặc chuẩn đầu ra clo mà sinh viên trả lời đúng trong bài kiểm tra (r'_x, u'_x, ap'_x, an'_x) tương ứng với các mức nhận thức <i>Nhớ, Hiểu, Áp dụng, Phân tích</i> .
Bước 2.4:	Tính P_x theo công thức (1) với x lần lượt là t và clo .
Bước 2.5:	Phản hồi về kiến thức đạt được của sinh viên, người học về chủ đề t dựa vào P_t . Phản hồi về kiến thức đạt được của sinh viên, người học về chuẩn đầu ra clo dựa vào P_{clo} .
Bước 3:	Tổng hợp các phản hồi về kiến thức đạt được của sinh viên, người học theo chủ đề hoặc chuẩn đầu ra môn học.

3.3. Bài toán đánh giá tiến trình học của sinh viên, người học qua các bài kiểm tra

Giả sử rằng với chủ đề t hoặc chuẩn đầu ra clo trong môn học mà sinh viên, người học đã thực hiện kiểm tra nhiều lần ở mức độ khó *level*. Trong bài toán đánh giá này, hệ thống cần phải đánh giá được kiến thức của sinh viên, người học về chủ đề t hoặc chuẩn đầu ra clo qua tất cả các bài kiểm tra mà họ đã thực hiện ở mức độ khó *level*.

Định nghĩa 3.4: Giả sử rằng kiến thức đạt được của sinh viên, người học về chủ đề t hoặc chuẩn đầu ra clo trong môn học ở lần kiểm tra i là $P_x(q_i)$, với x là t hoặc clo , trong đó $P_x(q_i)$ được tính theo công thức (1) và $Pn_{x_level} = \{P_{x_level}(q_1), P_{x_level}(q_2), \dots, P_{x_level}(q_n)\}$, với $1 \leq i \leq n$, là tập hợp các lần kiểm tra mà họ đã thực hiện về chủ đề t hoặc chuẩn đầu ra clo ở mức độ khó *level*. Khi đó, kiến thức của sinh viên, người học đạt được về chủ đề t hoặc chuẩn đầu ra clo qua n lần kiểm tra ở mức độ khó *level* được đánh giá theo công thức sau:

$$\mu(Pn_{x_level}) = \frac{1}{n} \sum_{i=1}^n P_{x_level}(q_i) \quad (2)$$

Dựa vào công thức (2), hệ thống được xây dựng đề xuất đưa ra các phản hồi và nhận xét như sau:

i) Khi x là chủ đề t , nhận xét về kiến thức đạt được của sinh viên, người học về chủ đề t qua n lần kiểm tra ở mức độ khó *level*.

- Nếu $\mu(Pn_{t_level}) \leq 0.3$, sinh viên kém về chủ đề t ở môn học;
- Nếu $0.3 < \mu(Pn_{t_level}) \leq 0.5$, sinh viên còn yếu về chủ đề t ở môn học;
- Nếu $0.5 < \mu(Pn_{t_level}) \leq 0.7$, sinh viên có kiến thức trung bình về chủ đề t ở môn học;
- Nếu $0.7 < \mu(Pn_{t_level}) \leq 0.9$, sinh viên có kiến thức

tốt về chủ đề t ở môn học;

- Nếu $0.9 < \mu(Pn_{t_level})$, sinh viên có kiến thức rất tốt về chủ đề t ở môn học;

ii) Khi x là chuẩn đầu ra clo , nhận xét về kiến thức đạt được của sinh viên, người học theo chuẩn đầu ra clo qua n lần kiểm tra ở mức độ khó *level*.

- Nếu $\mu(Pn_{clo_level}) < 0.5$, sinh viên chưa đạt được chuẩn đầu ra clo ở môn học với mức độ khó *level*;

- Nếu $\mu(Pn_{clo_level}) \geq 0.5$, sinh viên thể hiện có kiến thức tốt và đạt chuẩn đầu ra clo của môn học Cấu trúc dữ liệu và Giải thuật với mức độ khó *level*;

iii) Nhận xét về tiến trình học của của sinh viên, người học về chủ đề t hoặc chuẩn đầu ra clo qua n lần kiểm tra ở mức độ khó *level*.

- Nếu $P_{x_level}(q_1) = P_{x_level}(q_2) = \dots = P_{x_level}(q_n)$, tiến trình học của sinh viên không có sự thay đổi;
- Nếu $P_{x_level}(q_1) \leq P_{x_level}(q_2) \leq \dots \leq P_{x_level}(q_n)$, tiến trình học của sinh viên có sự tiến bộ;
- Nếu $P_{x_level}(q_1) \geq P_{x_level}(q_2) \geq \dots \geq P_{x_level}(q_n)$, tiến trình học của sinh viên đang có sự sa sút, sinh viên cần xem lại kiến thức và thực hành nhiều hơn;
- Các trường hợp còn lại, nếu Pn_{x_level} có vài điểm cực trị khác với sự đồng điệu của $P_{x_level}(q_1)$ và $P_{x_level}(q_n)$, tiến trình học của sinh viên chưa có sự ổn định:
 - Nếu $P_{x_level}(q_1) < P_{x_level}(q_n)$, mặc dù sinh viên có sự tiến bộ ở bài kiểm tra gần nhất nhưng tiến trình học của sinh viên chưa ổn định;
 - Nếu $P_{x_level}(q_1) > P_{x_level}(q_n)$, tiến trình học của sinh viên đang có sự sa sút ở lần kiểm tra gần

nhất, sinh viên cần xem lại kiến thức và thực hành nhiều hơn.

Định nghĩa 3.5: Bài toán đánh giá kiến thức của sinh viên, người học về chủ đề *TOPIC* hoặc chuẩn đầu ra *LO* qua các bài kiểm tra ở mức *LEVEL* và tiến trình học của sinh viên, người học được định nghĩa như sau:

$(PROFILE, QUIZN, TOPIC, LO, LEVEL) \rightarrow FEEDBACK$

Trong đó, *PROFILE* là thông tin của sinh viên, người học trên hệ thống; *QUIZN* là tập hợp các bài kiểm tra mà sinh viên, người học đã thực hiện; *TOPIC* là tập hợp các chủ đề mà sinh viên, người học đã

kiểm tra; *LO* là tập hợp các chuẩn đầu ra môn học mà sinh viên, người học đã kiểm tra; *LEVEL* là tập hợp các mức độ khó của bài kiểm tra mà sinh viên, người học đã thực hiện; *FEEDBACK* là kết quả và nhận xét mà hệ thống khuyến nghị cho sinh viên, người học dựa vào công thức (2).

Với tổ chức tri thức (*T, Q, R, CLO*) được định nghĩa ở 2.1, mô hình cấu trúc bài kiểm tra như ở định nghĩa 2.3, và các công thức (1) (2), thuật giải được thiết kế để hệ thống đánh giá kiến thức môn học của sinh viên, người học theo chủ đề hoặc chuẩn đầu ra qua các bài kiểm tra như sau:

Giải thuật 3.3: Đánh giá tiến trình học của sinh viên qua các bài kiểm tra theo chủ đề hoặc chuẩn đầu ra	
Input:	$(PROFILE, QUIZN, TOPIC, LO, LEVEL)$
Output:	- $\mu(Pn_{x_level})$ theo công thức (2) và nhận xét về kiến thức đạt được của sinh viên, người học về chủ đề $t \in TOPIC$ và chuẩn đầu ra $clo \in LO$ ở mức độ bài kiểm tra $level \in LEVEL$; - Nhận xét về quá trình học của sinh viên, người học về chủ đề $t \in TOPIC$ và theo chuẩn đầu ra $clo \in LO$ ở mức độ bài kiểm tra $level \in LEVEL$.
Bước 1:	- Với mỗi chủ đề $t \in TOPIC$ ở mức độ bài kiểm tra $level \in LEVEL$, lọc các bài kiểm tra <i>QUIZT</i> từ tập <i>QUIZN</i> mà sinh viên, người học đã thực hiện và tính số lần thực hiện bài kiểm tra $n_t = QUIZT$; - Với mỗi chuẩn đầu ra $clo \in LO$ ở mức độ bài kiểm tra $level \in LEVEL$, lọc các bài kiểm tra <i>QUIZLO</i> từ tập <i>QUIZN</i> mà sinh viên, người học đã thực hiện và tính số lần thực hiện bài kiểm tra $n_{clo} = QUIZLO$.
Bước 2:	- Với mỗi $q \in QUIZT$, tính $P_{t_level}(q_i)$ theo công thức (1); - Với mỗi $q \in QUIZLO$, tính $P_{clo_level}(q_i)$ theo công thức (1).
Bước 3:	- Tính $\mu(Pn_{t_level})$ theo công thức (2); - Tính $\mu(Pn_{clo_level})$ theo công thức (2).
Bước 4:	- Nhận xét về kiến thức đạt được của sinh viên, người học về chủ đề t qua n_t lần kiểm tra ở mức độ khó $level$ dựa vào $\mu(Pn_{t_level})$; - Nhận xét về kiến thức đạt được của sinh viên, người học theo chuẩn đầu ra clo qua n_{clo} lần kiểm tra ở mức độ khó $level$ dựa vào $\mu(Pn_{clo_level})$.
Bước 5:	- Nhận xét về quá trình học của của sinh viên, người học về chủ đề t qua n_t lần kiểm tra ở mức độ khó $level$ dựa vào $\{P_{t_level}(q_1), P_{t_level}(q_2), \dots, P_{t_level}(q_n)\}$; - Nhận xét về quá trình học của của sinh viên, người học theo chuẩn đầu ra clo qua n_{clo} lần kiểm tra ở mức độ khó $level$ dựa vào $\{P_{clo_level}(q_1), P_{clo_level}(q_2), \dots, P_{clo_level}(q_n)\}$; - Vẽ đồ thị mô tả tiến trình học của sinh viên, người học.

4. CÀI ĐẶT VÀ THỬ NGHIỆM

Dựa trên mô hình kiến thức cho môn học Cấu trúc dữ liệu và Giải thuật trong phần 2 và mô hình các bài toán trong phần 3, nhóm nghiên cứu đã triển khai và cài đặt một ứng dụng kiểm tra đánh giá kiến thức môn học Cấu trúc dữ liệu và Giải thuật cho sinh viên ngành Công nghệ thông tin tại Trường Đại học Quốc tế Hồng Bàng. Trong phần này, nhóm

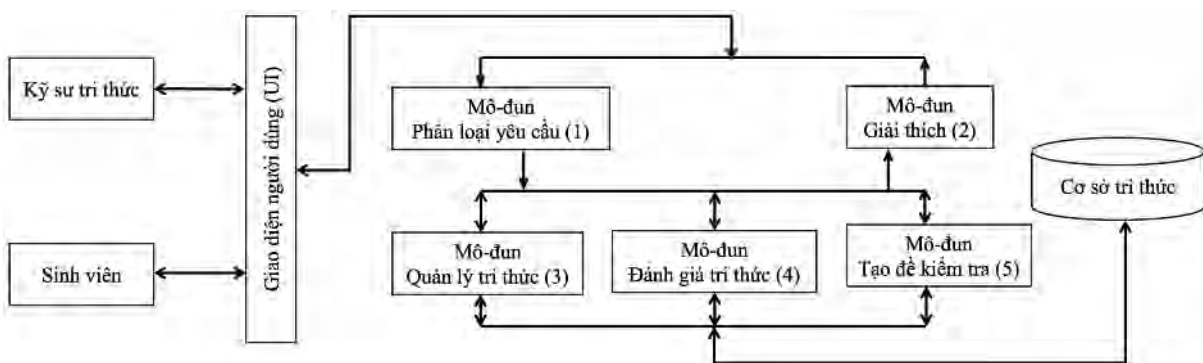
trình bày kiến trúc và cơ sở tri thức của hệ thống, cũng như đánh giá hệ thống đã được xây dựng.

4.1. Kiến trúc và cơ sở tri thức của hệ thống

Kiến trúc của hệ thống đánh giá kiến thức cho Cấu trúc dữ liệu và Giải thuật được hiển thị như trong Hình 1. Hệ thống có hai loại người dùng: sinh viên và kỹ sư tri thức. Sinh viên sẽ sử dụng chương trình

để hỗ trợ việc học của họ bằng cách làm bài kiểm tra và theo dõi tiến trình học tập của mình. Kỹ sư tri thức có thể quản lý cơ sở kiến thức của hệ thống thông qua giao diện người dùng. Mô-đun (1) sẽ phân loại yêu cầu từ người dùng và phân phối yêu cầu đó vào các mô-đun (3), (4), (5) một cách thích hợp. Mô-đun (3) có các chức năng để quản lý kiến thức như chèn, cập nhật, xóa. Mô-đun (4) sẽ đánh giá kiến thức về cấu trúc dữ liệu và thuật toán thông qua quá trình kiểm tra. Mô-đun (5) chịu trách nhiệm tạo ra một bài kiểm tra theo các yêu cầu cụ thể. Cơ sở kiến thức lưu trữ miền kiến thức về cấu trúc dữ liệu và thuật toán trong cơ sở dữ liệu SQL.

Các chủ đề của môn học Cấu trúc dữ liệu & Giải thuật được thu thập từ sách giáo khoa và bài giảng sử dụng trong môn học Cấu trúc dữ liệu & Giải thuật ngành Công nghệ thông tin tại Trường Đại học Quốc tế Hồng Bàng [15, 16]. Một số chủ đề đã được trình bày trong Bảng 3. Nhiều câu hỏi cũng được thu thập và xây dựng từ [15, 16]. Những câu hỏi này được chia thành các chủ đề và xác định độ khó bởi một số giảng viên Công nghệ thông tin tại Đại học Quốc tế Hồng Bàng. Hiện tại, kho câu hỏi của hệ thống mà nhóm xây dựng có khoảng 345 câu. Số lượng câu hỏi trong 6 chương của môn học Cấu trúc dữ liệu và Giải thuật mà nhóm thu thập và xây dựng được trình bày trong Bảng 3.



Hình 1. Kiến trúc của hệ thống kiểm tra đánh giá

Bảng 3. Số lượng các câu hỏi trong 6 chương của môn học Cấu trúc dữ liệu và Giải thuật

TT	Nội dung môn học Cấu trúc dữ liệu & Giải thuật	Số lượng các câu hỏi theo thang đo Bloom				Tổng
		R	U	AP	AN	
1	Chương 1: Tổng quan Cấu trúc dữ liệu và Giải thuật	47	17	12	12	86
2	Chương 2: Tìm kiếm và Sắp Xếp	20	24	16	11	71
3	Chương 3: Danh sách liên kết	21	16	6	6	49
4	Chương 4: Ngăn xếp, Hàng đợi	15	8	10	11	44
5	Chương 5: Cây	12	9	15	15	51
6	Chương 6: Bảng băm	10	12	12	8	52
Tổng		125	86	71	63	345

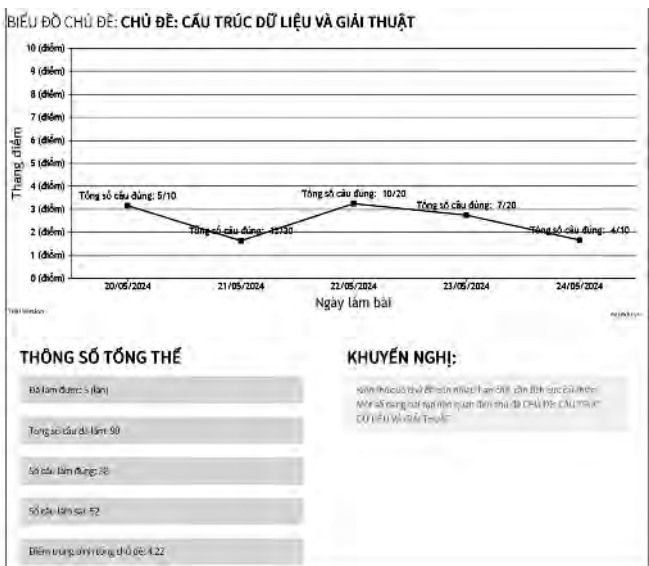
4.2. Cài đặt và thử nghiệm chương trình

Hệ thống đánh giá kiến thức được đề xuất có thể tạo ra một bài kiểm tra theo các yêu cầu cụ thể. Bài kiểm tra này đáp ứng yêu cầu của sinh viên về nội dung, chủ đề và mức độ khó. Bên cạnh đó, nó có thể đánh giá mức độ kiến thức của sinh viên trên từng chủ đề. Từ những kết quả này, sinh viên sẽ có kế hoạch tự học để cải thiện các chủ đề còn yếu. Chương trình được cài đặt bằng mã nguồn C# và framework ASP.NET MVC, cơ sở tri thức được mô hình hóa và lưu trữ dưới dạng một cơ sở dữ liệu quan hệ bằng công cụ Microsoft SQL Server. Ứng

dụng được viết thành một ứng dụng web (web application), cho phép người sử dụng là sinh viên, học sinh và giảng viên có thể thao tác với hệ thống qua môi trường Internet. Sau khi hoàn thành một bài kiểm tra, hệ thống sẽ phân tích các câu hỏi của bài kiểm tra này theo các chủ đề và hiển thị kết quả chẩn đoán khả năng của sinh viên về từng chủ đề. Như được hiển thị trong Hình 2, hệ thống hiển thị đánh giá của sinh viên về 6 chủ đề của khóa học Cấu trúc Dữ liệu và Giải thuật. Sau khi hoàn thành nhiều bài kiểm tra, hệ thống sẽ đưa ra biểu đồ, thông số tổng thể và khuyến nghị cho sinh viên như ở Hình 3.



Hình 2. Giao diện kết quả sau khi thực hiện bài kiểm tra



Hình 3. Giao diện xem tiến trình học của sinh viên

4.3. So sánh và đánh giá hệ thống đề xuất

Bảng 4 dưới đây cho thấy sự so sánh giữa hệ thống kiểm tra, đánh giá mà nhóm thực hiện và các hệ thống hỗ trợ kiểm tra trắc nghiệm khác. Các tiêu

chí so sánh được dựa trên chức năng tạo bài kiểm tra, đánh giá sinh viên dựa trên kết quả kiểm tra và miền tri thức mà các hệ thống hỗ trợ kiểm tra, đánh giá.

Bảng 4. So sánh và đánh giá hệ thống đề xuất

STT	Ứng dụng, Chương trình	Khả năng tạo đề kiểm tra	Khả năng đưa ra đánh giá	Miền tri thức
1	Hệ thống hỗ trợ thông minh cho việc đánh giá kiến thức trong toán học trung học phổ thông [7]	- Hệ thống có thể tạo ngẫu nhiên một bài kiểm tra với các tham số chẳng hạn như: thời gian, số lượng câu hỏi, mức độ khó. - Hệ thống không thể tạo một bài kiểm tra với các nội dung và chủ đề được chọn.	- Hệ thống có thể đánh giá mức độ kiến thức của người học trên từng chủ đề thông qua một bài kiểm tra. - Hệ thống có thể theo dõi quá trình học của người dùng thông qua các bài kiểm tra của họ và chẩn đoán nội dung mà họ cần cải thiện thông qua kết quả.	Toán THPT
2	International English Test [9]	- Các chức năng không được công khai cho người dùng của hệ thống.	- Hệ thống chỉ đánh giá dựa trên điểm số - Hệ thống không đưa ra phân tích cách để tính điểm	Tiếng Anh
3	VioEdu [10]	- Các chức năng không được công khai cho người dùng của hệ thống.	- Hệ thống có thể đánh giá mức độ kiến thức của người học trên từng chủ đề thông qua một bài kiểm tra. - Hệ thống có thể theo dõi quá trình học của người dùng thông qua các bài kiểm tra của họ và chẩn đoán nội dung mà họ cần cải thiện thông qua kết quả	Toán Tiểu học, THCS, và THPT

STT	Ứng dụng, Chương trình	Khả năng tạo đề kiểm tra	Khả năng đưa ra đánh giá	Miền tri thức
4	HocMai [11]	- Hệ thống có các bài kiểm tra sẵn có do các giảng viên có kinh nghiệm tạo ra. - Hệ thống không cho phép người dùng tạo đề theo yêu cầu	- Hệ thống chỉ đánh giá dựa trên điểm số. - Hệ thống không đưa ra phân tích cách để tính điểm.	Toán THPT
5	Elearning [12]	- Các giảng viên có thể tạo một bài kiểm tra nhưng họ phải nhập hoặc tạo câu hỏi thủ công cho từng bài kiểm tra cụ thể. - Hệ thống không thể tạo một bài kiểm tra với các nội dung và chủ đề được chọn.	- Hệ thống chỉ đánh giá dựa trên điểm số. - Hệ thống không đưa ra phân tích cách để tính điểm	Cho phép thực hiện kiểm tra trên nhiều miền tri thức
6	Hệ thống đề xuất	- Hệ thống có thể tạo ra một bài kiểm tra ngẫu nhiên với các tham số như: số lượng câu hỏi, độ khó của bài kiểm tra, chủ đề, và thời gian. - Hệ thống cho phép tạo một bài kiểm tra với các nội dung và chủ đề được chọn.	- Hệ thống có thể đánh giá kiến thức của người học về mỗi chủ đề hoặc chuẩn đầu ra của môn học thông qua một bài kiểm tra. - Hệ thống có thể đánh giá tiến trình học của sinh viên, người học theo chủ đề, hoặc chuẩn đầu ra qua toàn bộ các bài kiểm tra.	Cấu trúc dữ liệu và Giải thuật

4.4. Bàn luận kết quả

Kết quả thử nghiệm bước đầu cho thấy hệ thống đề xuất có khả năng tạo đề kiểm tra phù hợp với chuẩn đầu ra của môn học, đồng thời sử dụng hiệu quả ngân hàng câu hỏi theo từng chủ đề và mức độ nhận thức. Hệ thống có thể đánh giá năng lực người học thông qua việc thống kê số câu trả lời đúng theo chuẩn đầu ra môn học và chủ đề kiến thức. Dựa trên kiến trúc hệ thống trình bày tại Mục 4.1, mô hình được thiết kế gồm các thành phần rõ ràng: giao diện người dùng, mô-đun sinh đề, mô-đun đánh giá và cơ sở tri thức. Mỗi yêu cầu từ người dùng được phân loại và xử lý thông qua hệ thống suy luận, nhằm đảm bảo sự phù hợp giữa nội dung kiểm tra và chuẩn đầu ra của môn học. Ứng dụng đã được triển khai thử nghiệm (Mục 4.2), với giao diện thân thiện, dễ thao tác và hỗ trợ sinh đề kiểm tra theo nhiều tiêu chí như: Chủ đề, mức độ khó, thời gian làm bài và số lượng câu hỏi. Điều này cho thấy hệ thống có tính thực

tiễn và tiềm năng triển khai trong môi trường đào tạo đại học. Kết quả so sánh với các nền tảng hiện có như trong Mục 4.3 cho thấy hệ thống đề xuất nổi bật ở khả năng đánh giá tiến trình học tập tự động theo từng bài kiểm tra, đồng thời tổng hợp đánh giá được năng lực của sinh viên qua các bài kiểm tra, đặc biệt là đối với học phần Cấu trúc dữ liệu và Giải thuật. Ngoài ra, chức năng phản hồi cá nhân hóa dựa trên thống kê định tính và định lượng giúp sinh viên nhận biết các nội dung cần cải thiện, đồng thời hỗ trợ giảng viên xây dựng chiến lược điều chỉnh phù hợp. Sự liên kết giữa mô hình lý thuyết và phần cài đặt thực nghiệm được thể hiện rõ thông qua việc ánh xạ trực tiếp các trường dữ liệu (*question_content*, *bloom_level*, *discrimination_level*, ...) vào quy trình sinh đề và đánh giá. Các thuộc tính này được sử dụng làm điều kiện lọc khi lựa chọn câu hỏi, góp phần đảm bảo tính khách quan và tương thích với năng lực của người học. Hệ thống cũng lưu lại kết quả làm

bài của sinh viên theo từng bài kiểm tra, từ đó phân tích tiến trình học tập và đưa ra các khuyến nghị cải thiện phù hợp.

5. KẾT LUẬN

Bài báo này trình bày thiết kế và kỹ thuật xây dựng một ứng dụng hỗ trợ kiểm tra, đánh giá kiến thức trong môn học Cấu trúc dữ liệu và Giải thuật. Hệ thống đề xuất cho phép tạo đề kiểm tra trắc nghiệm khách quan theo chuẩn đầu ra và đề cương môn học Cấu trúc dữ liệu và Giải thuật trong chương trình đào tạo ngành Công nghệ thông tin tại Trường Đại học Quốc tế Hồng Bàng. Hệ thống đánh giá kiến thức của sinh viên dựa trên lý thuyết khảo thí cổ điển. Tuy nhiên, cần có những nghiên cứu sâu hơn để đánh giá tác động của hệ thống đối với quá trình học tập của sinh viên.

Trong tương lai, ngân hàng câu hỏi của môn học cần được rà soát và mở rộng. Nhóm nghiên cứu dự

kiến tích hợp lý thuyết khảo thí hiện đại (Item Response Theory - IRT) [13] nhằm đánh giá khách quan hơn về độ khó của câu hỏi và năng lực của sinh viên. Ngoài ra, hệ thống cần được mở rộng và cải tiến với các chức năng sau: (i) quản trị, thống kê và báo cáo; (ii) loại bỏ, cải tiến câu hỏi trắc nghiệm và đáp án; (iii) tổng quát hóa mã nguồn để hỗ trợ kiểm tra, đánh giá nhiều môn học trong chương trình đào tạo Công nghệ thông tin bậc đại học; (iv) nghiên cứu và phát triển các hình thức kiểm tra đánh giá khác ngoài trắc nghiệm khách quan, bao gồm câu hỏi điền số, điền cụm từ, câu trả lời ngắn dạng số, câu trả lời ngắn dạng cụm từ, câu ghép đôi và bài tập lập trình.

LỜI CẢM ƠN

Nghiên cứu này được Trường Đại học Quốc tế Hồng Bàng cấp kinh phí thực hiện dưới mã số đề tài GVTC18.02.

TÀI LIỆU THAM KHẢO

- [1] Lin, C. C., Huang, A. Y., & Lu, O. H. (2023). "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] H. N. Long, M. T. Thành, và Đ. V. Nhơn, "Designing a knowledge assessment system for the Data structures and Algorithms course", *HIJUS*, vol 6, tr 65–74, tháng 6 2024.
- [9] International English Test, "International English Test". [Online]. Available: <https://internationalenglishtest.com/>. [Accessed 10 03 2025].
- [10] FPT, "VioEdu". [Online]. Available: <https://vio.edu.vn/>. [Accessed 10 03 2025].
- [11] Hệ thống Giáo dục HocMai, "HocMai". [Online]. Available: <https://hocmai.vn/>. [Accessed 10 03 2025].

- [12] Hong Bang International University, "Elearning". [Online]. Available: <https://elearning.hiu.vn/>. [Accessed 10 03 2025].
- [13] L. Q. Thiệp, "Đo lường trong giáo dục - Lí thuyết và Ứng dụng", NXB Đại học Quốc gia Hà Nội, 2010
- [14] T. T. Đào, "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
- [15] Đ. V. Nhơn, T. Q. Sơn, "Giáo trình Cấu trúc dữ liệu & Giải thuật", NXB ĐHQG TP. HCM, 2015
- [16] H. N. Long, "Lectures of Data Structures and Algorithms". [Online]. Available: https://www.youtube.com/playlist?list=PLrcfrbKhqmAWrloc47jVx1eEzS9gJUdg_. [Accessed 10 03 2025].

Developing an Application for Knowledge Assessment in the Data Structures and Algorithms Course

Hoang Ngoc Long, Do Van Nhon

ABSTRACT

The Data Structures and Algorithms course serves as a fundamental component of the Information Technology curriculum, enabling students to understand and apply algorithms to real-world problems. However, teaching and learning this subject present several challenges, as students often struggle to grasp and implement concepts, algorithms, and data structures effectively. Additionally, assessing students' knowledge and academic performance requires accuracy and objectivity. In response to these challenges, this paper proposes the design and implementation of a system that supports knowledge assessment to enhance both learning and teaching effectiveness. This system not only helps students reinforce their understanding but also assists instructors in monitoring learning progress and providing timely feedback, ultimately improving the quality of Information Technology education.

Keywords: *assessment system, data structures and algorithms, E-learning, intelligent tutoring system, knowledge representation*

Received: 20/3/2025

Revised: 06/5/2025

Accepted for publication: 10/5/2025