

**Mã đề: W11**

- LƯU Ý:**
- Sinh viên được *tham khảo tài liệu trong 2 tờ giấy A4*.
  - Đề kiểm tra gồm **24 câu hỏi trắc nghiệm và 2 câu tự luận**.

**HƯỚNG DẪN LÀM BÀI:**

Sinh viên chọn 1 câu trả lời đúng nhất cho các câu hỏi trắc nghiệm và trình bày lời giải cho các câu tự luận. Nếu chọn câu trả lời (E) cho câu hỏi trắc nghiệm thì sinh viên cần trình bày đáp án khác so với đáp án ở các câu (A), (B), (C), và (D) và giải thích lựa chọn (E) của mình.

---

**I. Phần trắc nghiệm (0.25 điểm/câu):**

**Câu 1.** Khả năng hỗ trợ lưu trữ bền vững của hệ quản trị cơ sở dữ liệu (*database management system, DBMS*) được thể hiện qua .... **sustainable storage**

A. Thực hiện phục hồi dữ liệu khi chương trình ứng dụng của người dùng đang thực thi trên cơ sở dữ liệu được hỗ trợ bởi DBMS thì gặp sự cố.

B. Dữ liệu trong cơ sở dữ liệu được hỗ trợ bởi DBMS luôn nhất quán.

C. Các thay đổi dữ liệu của các giao tác đã commit luôn được ghi nhận trong cơ sở dữ liệu được hỗ trợ bởi DBMS.

**D. Câu A, B, và C đều đúng.**

E. Ý kiến khác.

**Câu 2.** Trong DBMS, thành phần ***Storage Manager*** chịu trách nhiệm chính cho công việc gì ?

A. Xác định dữ liệu và chỉ mục cần được lưu trữ và truy xuất cho việc xử lý câu truy vấn.

B. Xác định địa chỉ khối và giao tiếp với thiết bị lưu trữ (*storage device*) để đọc/ghi dữ liệu tương ứng từ vùng bộ đệm (*buffer*).

**C. Giao tiếp với trình điều khiển đĩa (*disk controller*) để đọc/ghi dữ liệu tương ứng từ vùng bộ đệm (*buffer*) đối với thiết bị lưu trữ (*storage device*).**

D. Câu A, B, và C đều sai.

E. Ý kiến khác.

**Câu 3.** Cho kích thước của một khối (*block*) là  $B = 512$  byte. Một tập tin Student gồm các bản ghi có chiều dài cố định và được lưu trữ trên đĩa theo cách phân khối không phủ (*unspanned blocking*). Mỗi bản ghi gồm các vùng tin (*field*) sau: ID (10 byte), FirstName (30 byte), LastName (30 byte), DOB (8 byte), Sex (1 byte), Hometown (15 byte), Contact (25 byte), DeletionMarker (1 byte). Xác định hệ số phân khối dành cho tập tin này.

A. 3.

**B. 4.**

C. 5.

D. 6.

E. Ý kiến khác.

**Câu 4.** Nếu các khối dữ liệu của tập tin Student này được đặt liền kề nhau thì khi đọc toàn bộ dữ liệu của tập tin, kỹ thuật bộ đệm đôi (*double buffering*) ... khi so với trường hợp các khối không liền kề nhau.

**A. Tiết kiệm thời gian tìm kiếm (*Seek time*).**

C. Giảm thời gian truyền khối (*Block transfer time*).

B. Giữ nguyên độ trễ quay (*Rotational delay*).

D. Tăng các thời gian này.

E. Ý kiến khác.

**Câu 5.** Việc lưu trữ các khối dữ liệu của tập tin Student liên tục nhau trên đĩa phù hợp khi ...

A. Tập tin động.

**B. Tập tin tĩnh.**

C. Cập nhật liên tục.

D. Câu A, B, và C đều sai.

E. Ý kiến khác.

**Câu 6.** Giả sử các bản ghi trong tập tin Student được sắp thứ tự vật lý theo vùng tin khóa ID. Chỉ mục B+-tree được định nghĩa trên vùng tin ID. Chỉ mục này là dạng chỉ mục ...

- A. Thưa, không có dùng neo khối.
- B. Dày, có dùng neo khối.
- C. Thưa, có dùng neo khối.
- D. Dày, không có dùng neo khối.
- E. Ý kiến khác.

**Câu 7.** Giả sử các bản ghi trong tập tin Student không được băm hay sắp thứ tự vật lý theo bất kỳ vùng tin nào. Chỉ mục B+-tree được định nghĩa trên vùng tin ID. Thao tác nào trên tập tin này không hiệu quả?

- A. Thêm mới 1 bản ghi với ID = 1234567890.
- B. Tìm kiếm tuần tự với ID = 1234567890.
- C. Tìm kiếm trên chỉ mục với ID = 1234567890.
- D. Cả ba thao tác trên.
- E. Ý kiến khác.

**Câu 8.** Sự khác biệt giữa chỉ mục cụm và chỉ mục thứ cấp trên vùng tin khóa ID của 1 tập tin là ...

- A. Chỉ mục cụm là chỉ mục thưa và chỉ mục thứ cấp là chỉ mục dày.
- B. Chỉ mục cụm có thể dùng neo khối, nhưng chỉ mục thứ cấp không thể dùng neo khối.
- C. Không gian lưu trữ của chỉ mục cụm nhỏ hơn so với không gian lưu trữ của chỉ mục thứ cấp.
- D. Câu A, B, và C đều đúng.
- E. Ý kiến khác.

**Câu 9.** Cho chỉ mục thứ cấp B-tree có số lượng con trở cây  $p$  ở mỗi nút là 38. Số mức (bao gồm mức gốc, root) của B-tree này là bao nhiêu khi chỉ mục cho vùng tin khóa ID không có thứ tự của tập tin dữ liệu gồm 2,000,000 bản ghi?

- A. 3
- B. 4
- C. 5
- D. 6
- E. Ý kiến khác.

**Câu 10.** Sắp thứ tự ngoại không được dùng trong việc xử lý cho phép toán nào sau đây?

- A. Kết 2 tập tin dữ liệu với phương pháp sắp thứ tự-trộn (*sort-merge join*).
- B. Trả về kết quả khi câu lệnh có mệnh đề "ORDER BY".
- C. Hợp 2 tập tin dữ liệu.
- D. Chọn các bản ghi của tập tin dữ liệu với điều kiện ">".
- E. Ý kiến khác.

**Câu 11.** Dạng biểu diễn bên trong nào mà các DBMS thường dùng khi xử lý và tối ưu hóa truy vấn SQL?

- A. Cây truy vấn.
- B. Đồ thị truy vấn.
- C. Biểu thức đại số quan hệ.
- D. Câu lệnh SELECT của ngôn ngữ SQL.
- E. Ý kiến khác.

**Câu 12.** Chi phí tính toán cần được xem xét khi tối ưu hóa dựa trên chi phí cho loại cơ sở dữ liệu nào?

- A. Cơ sở dữ liệu nhỏ.
- B. Cơ sở dữ liệu lớn.
- C. Cơ sở dữ liệu phân tán.
- D. Cả ba trường hợp trên.
- E. Ý kiến khác.

**Câu 13.** Cho tập tin Customer có vùng tin khóa ID là vùng tin được dùng sắp thứ tự cho các bản ghi, có chỉ mục sơ cấp B+-tree trên vùng tin này. Phép chọn  $\sigma_{ID>1000}(\text{Customer})$  có thể được xử lý theo các phương pháp nào sau đây?

- A. Tìm kiếm tuần tự.
- B. Tìm kiếm nhị phân.
- C. Tìm kiếm qua chỉ mục B+-tree.
- D. Cả ba phương pháp trên.
- E. Ý kiến khác.

**Câu 14.** Cho lịch biểu sau, hệ thống cần làm gì vì ngưng thực thi khi có sự cố trong chương trình của giao tác T3 tại thời điểm giao tác T1 đã *commit*?

S:  $r_2(X); r_1(X); r_3(Y); w_2(X); w_1(X); r_2(Y); c_2; r_1(Y); r_3(X); w_3(Y); c_1$ ; (hệ thống ngưng thực thi)

- A. Đảm bảo bền vững cho các giao tác T1 và T2.
- B. Đảm bảo nhất quán cho giao tác T3.
- C. Đảm bảo các giao tác T1 và T2 đều không quay lui theo giao tác T3.
- D. Đảm bảo giao tác T3 không quay lui.
- E. Ý kiến khác.

**Câu 15.** Cho biết vấn đề gì xảy ra với trình tự thực thi dưới đây khi điều khiển tương tranh không được thực hiện?

S:  $r_1(X); w_1(X); r_2(X); r_1(Y); w_1(Y); w_2(X); a_1$

- A. Không có vấn đề xảy ra.
- B. Mất cập nhật (*lost update*).
- C. Cập nhật tạm thời (*temporary update*).
- D. Không thể đọc lặp lại dữ liệu (*unrepeatable reads*).
- E. Ý kiến khác.

**Câu 16.** Khi giao tác T đạt đến điểm *commit*, các nội dung nào sau đây cần được hoàn thành?

- A. Các tác vụ xử lý dữ liệu của chương trình ứng dụng thực thi giao tác T.
- B. Ghi nhật ký cho các thao tác trên dữ liệu của giao tác T.
- C. Ghi nhật ký cho việc commit của giao tác T với mục [commit, T].
- D. Ghi các cập nhật trên dữ liệu của giao tác T vào cơ sở dữ liệu trên đĩa.
- E. Ý kiến khác.

**Câu 17.** Trong lịch biểu S, cặp tác vụ nào là xung đột? S:  $r_1(X); w_1(X); r_2(X); r_1(Y); w_1(Y); w_2(X)$ .

- A.  $r_1(X); w_1(X)$ .
- B.  $w_1(X); r_2(X)$ .
- C.  $r_2(X); r_1(Y)$ .
- D.  $w_1(Y); w_2(X)$ .
- E. Ý kiến khác.

**Câu 18.** Cho nội dung điều khiển tương tranh của lịch biểu sau dựa trên kỹ thuật điều khiển tương tranh dùng khóa hai pha với `read_lock()`, `write_lock()`, và `unlock()`. Nội dung này **KHÔNG ĐÚNG** vì ...

T1	T2
<code>read_lock(X);</code> <code>r<sub>1</sub>(X);</code> <code>unlock(X);</code>	
	<code>write_lock(X);</code> <code>r<sub>2</sub>(X);</code> <code>w<sub>2</sub>(X);</code> <code>unlock(X);</code>
<code>write_lock(X);</code> <code>r<sub>1</sub>(X);</code> <code>unlock(X);</code>	

- A. Thực hiện `unlock()` trên X khi không có khóa trên X.
- B. Trước khi đọc X thì không có khóa `read_lock()` hoặc `write_lock()` trên X.
- C. Trước khi ghi X thì không có khóa `write_lock()` trên X.
- D. Khi thực hiện xong thao tác trên X thì mở khóa `unlock()` trên X.
- E. Ý kiến khác.

**Câu 19.** Đặc điểm chung của các kỹ thuật điều khiển tương tranh đa phiên bản là ...

- A. Khóa chết không bao giờ xảy ra.
- B. Thực hiện điều khiển tương tranh trên từng phiên bản dữ liệu trong quá trình thực thi của các giao tác.
- C. Tác vụ đọc không bao giờ bị từ chối, nhưng tác vụ ghi có thể bị từ chối.
- D. Câu A, B, và C đều đúng.

E. Ý kiến khác.

**Câu 20.** Cho nội dung điều khiển tương tranh của lịch biểu sau dựa trên thứ tự nhãn thời gian (*timestamp ordering*). Điền giá trị nhãn thời gian đọc (*read\_TS*) và nhãn thời gian ghi (*write\_TS*) cho Y?

T1	T2	T3	Y
TS (T1) = 48	TS(T2) = 55	TS(T3) = 22	read_TS = 0 write_TS = 0
$r_1(Y)$			read_TS = 48 write_TS = 0
	$r_2(Y)$		read_TS = 55 write_TS = 0
		$r_3(Y)$	read_TS = 55 write_TS = 0
	<u><math>w_2(Y)</math></u>		read_TS = ... write_TS = ...
		<u><math>w_3(Y)</math></u>	read_TS = ... write_TS = ...

A. Y có *read\_TS* = 55 và *write\_TS* = 55 sau khi T2 thực hiện  $w_2(Y)$ . Các giá trị này không đổi sau khi T3 thực hiện  $w_3(Y)$  vì  $w_3(Y)$  bị từ chối và T3 bị quay lui với TS mới.

B. Y có *read\_TS* = 55 và *write\_TS* = 55 sau khi T2 thực hiện  $w_2(Y)$ . Y có *read\_TS* = 22 và *write\_TS* = 22 sau khi T3 thực hiện  $w_3(Y)$ .

C. Y có *read\_TS* = 55 và *write\_TS* = 0 sau khi T2 thực hiện  $w_2(Y)$  vì  $w_2(Y)$  bị từ chối và T2 bị quay lui với TS mới. Y có *read\_TS* = 55 và *write\_TS* = 22 sau khi T3 thực hiện  $w_3(Y)$ .

D. Không đủ thông tin để xác định *read\_TS* và *write\_TS* cho Y.

E. Ý kiến khác.

**Câu 21.** Cho nội dung điều khiển tương tranh của lịch biểu sau với kỹ thuật dựa trên thứ tự nhãn thời gian đa phiên bản (*multiversion timestamp ordering*). Tác vụ ghi của giao tác T1 sẽ được xử lý như thế nào?

T1	T2	T3	X
TS(T1) = 50	TS(T2) = 65	TS(T3) = 45	read_TS=0 write_TS=0
$r_1(X)$			
	$r_2(X)$		
	$w_2(X)$		
		$r_3(X)$	
<u><math>w_1(X)</math></u>			

A. Tạo phiên bản mới cho X với *read\_TS* = 50 và *write\_TS* = 50.

B. Từ chối vì không đảm bảo thứ tự cho các tác vụ xung đột giữa T1 và T3.

C. Từ chối vì không đảm bảo thứ tự cho các tác vụ xung đột giữa T1 và T2.

D. Chưa đủ thông tin mô tả để xác định cách xử lý cho tác vụ ghi của giao tác T1.

E. Ý kiến khác.

**Câu 22.** Giao thức ghi nhật ký trước (*Write-Ahead Logging*) đảm bảo phục hồi cho ...

A. các giao tác đã commit với các nội dung AFIM và các giao tác chưa commit với các nội dung BFIM trong các kỹ thuật phục hồi tại chỗ.

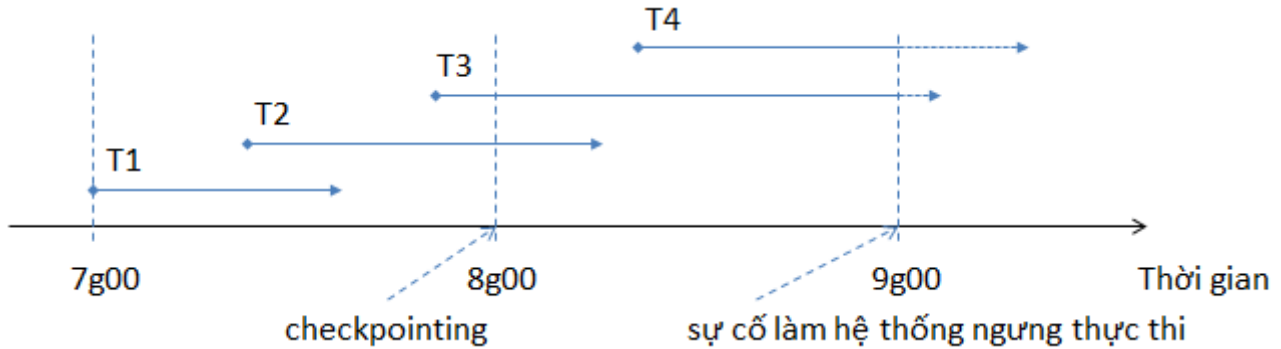
B. các giao tác đã commit với các nội dung BFIM và các giao tác chưa commit với các nội dung AFIM trong các kỹ thuật phục hồi tại chỗ.

C. các giao tác đã commit với các nội dung AFIM và các giao tác chưa commit với các nội dung BFIM trong tất cả các kỹ thuật phục hồi.

D. các giao tác đã commit với các nội dung BFIM và các giao tác chưa commit với các nội dung AFIM trong tất cả các kỹ thuật phục hồi.

E. Ý kiến khác.

**Câu 23.** Cho hiện trạng của hệ thống ngay khi hệ thống gặp sự cố lúc 9g00. Giả sử kỹ thuật cập nhật tức thời với dạng UNDO/NO-REDO được dùng để phục hồi hệ thống. Xác định phục hồi cho mỗi giao tác.



- A. Bỏ qua các tác vụ của giao tác T1, tái thực hiện cho các tác vụ của giao tác T2, và tháo gỡ cho các tác vụ của giao tác T3 và T4.
- B. Bỏ các tác vụ của giao tác T1 và T2, tháo gỡ cho các tác vụ của giao tác T3 và T4.**
- C. Bỏ các tác vụ của giao tác T3 và T4, tái thực hiện cho các tác vụ của giao tác T1 và T2.
- D. Bỏ qua các tác vụ của giao tác T1, T2, và T3, tháo gỡ cho các tác vụ của giao tác T4.
- E. Ý kiến khác.

**Câu 24.** ARIES là kỹ thuật phục hồi khác với phân trang bóng âm (*shadow paging*) ở điểm nào sau đây?

- A. ARIES dạng phục hồi tại chỗ và phân trang bóng âm không phải.
- B. ARIES tái hiện lịch sử cập nhật dữ liệu và phân trang bóng âm không phải.
- C. ARIES thuộc dạng UNDO/REDO và phân trang bóng âm không phải.
- D. Cả ba đặc điểm trên.**
- E. Ý kiến khác.

## II. Phần tự luận:

**Câu 25.** Cho lược đồ của hai tập tin dữ liệu *Student* và *Major*:

*Student*(ID, FName, LName, DOB, Hometown, EntranceYear, MCode)  
*Major*(Code, Name, Description, EstablishedYear, Faculty)

Tập tin dữ liệu *Student* có thông tin như sau:

- Số bản ghi  $r_S = 15,000$ , được chứa trong  $b_S = 5,000$  khối, với cách phân khối không phủ.
- Khóa chính: *ID*, có chỉ mục thứ cấp B+-tree gồm 4 mức.
- Khóa ngoại: *MCode*, tham chiếu đến khóa chính *Code* của tập tin dữ liệu *Major*, có 40 giá trị phân biệt, có chỉ mục cụm B+-tree gồm 2 mức.
- Vùng tin không khóa: *EntranceYear*, có 8 giá trị phân biệt và được phân bố theo tỉ lệ % như trong bảng sau, có chỉ mục thứ cấp B+-tree gồm 1 mức và 1 mức trung gian chứa các con trỏ bản ghi.

<i>EntranceYear</i>	2012	2013	2014	2015	2016	2017	2018	2019
Tỉ lệ %	1	3	8	12	18	19	19	20

- Vùng tin không khóa: *Hometown*, có 30 giá trị phân biệt và được phân bố đều cho các bản ghi, có chỉ mục thứ cấp B+-tree gồm 2 mức và 1 mức trung gian chứa các con trỏ bản ghi.

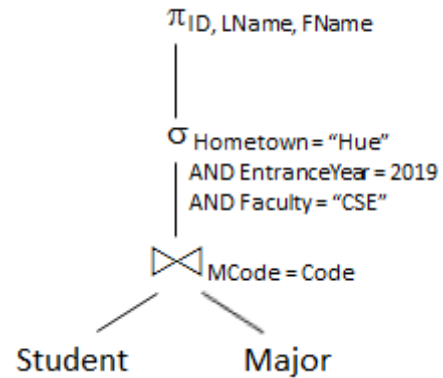
Tập tin dữ liệu *Major* có thông tin như sau:

- Số bản ghi  $r_M = 40$ , được chứa trong  $b_M = 6$  khối, với cách phân khối không phủ.
- Khóa chính: *Code*.
- Khóa ngoại: *Faculty*, tham chiếu đến khóa chính *Code* của tập tin dữ liệu *Faculty*.
- Vùng tin không khóa: *EstablishedYear*, có chỉ mục thứ cấp B+-tree gồm 1 mức và 1 mức trung gian chứa các con trỏ bản ghi.

Hệ số phân khối của kết quả kết đầy đủ giữa *Student* và *Major* là 2 với cách phân khối không phủ.

**25.1.** Mô tả đặc điểm lưu trữ vật lý của mỗi tập tin dữ liệu: *Student*, *Major*. (0.5 điểm)

**25.2.** Cho cây truy vấn sau dùng để lấy ra mã số sinh viên, họ, và tên của những sinh viên có quê quán ở Huế, mới nhập học vào năm 2019 ở khoa Khoa Học và Kỹ Thuật Máy Tính. Vẽ cây truy vấn tương đương cây truy vấn đã cho. (0.5 điểm)



**25.3.** Cho các phương pháp xử lý cho phép toán chọn sau đây:

Phương pháp 1: Tìm kiếm tuần tự

Phương pháp 2: Tìm kiếm nhị phân

Phương pháp 3: Tìm kiếm dựa trên chỉ mục

Sử dụng phương pháp tối ưu hóa truy vấn **dựa trên chi phí**, chi phí của mỗi phương pháp là bao nhiêu cho phép chọn đã chọn? Phương pháp nào nên được chọn? (1.25 điểm)

Phép chọn:  $\sigma_{Hometown="Hue" \text{ AND } EntranceYear=2019} (Student)$

**25.4.** Giả sử thực hiện phép kết *Student*  $\bowtie_{MCode = Code}$  *Major* bằng tối ưu hóa **dựa trên chi phí**.

Xác định chi phí của mỗi phương pháp sau cho phép kết. Phương pháp nào nên được chọn để xử lý phép kết cho câu truy vấn? (0.75 điểm)

Phương pháp 4: Kết với 1 vòng lặp (*single loop*)

Phương pháp 5: Kết với băm (*hash join*). Giả sử tập tin dữ liệu *Major* là tập tin nhỏ, có thể được băm và lưu trong vùng đệm.

**Câu 26.** Cho lịch biểu S trong bảng sau.

**26.1.** Giả sử phần này bỏ qua các thông tin về *commit*, *checkpoint*, sự cố làm hệ thống ngưng thực thi. Lịch biểu S được viết lại là:

S:  $r_1(X); r_3(X); r_2(Y); r_1(Y); w_2(Y); r_3(Y); w_3(X); w_1(Y)$

Lịch biểu này có khả năng tuần tự hóa (*serializable*) không tính đến thời điểm trước 8g30? Nếu có thì lịch biểu tuần tự tương đương là gì? Nếu không thì tại sao? (0.5 điểm)

Thời gian	T1	T2	T3
7g00	$r_1(X)$		
7g05			$r_3(X)$
7g10		$r_2(Y)$	
7g15	$r_1(Y)$		
7g20		$w_2(Y, 5, 10)$	
7g30			$r_3(Y)$
7g35		commit	
7g40	[checkpoint]		
8g00			$w_3(X, 3, 9)$
8g10	$w_1(Y, 5, 18)$		
8g30	- sự cố làm hệ thống ngưng thực thi -		

**26.2.** Giả sử lịch biểu S được điều khiển tương tranh bằng kỹ thuật khóa hai pha với khóa chia sẻ (*shared, read*) và khóa loại trừ (*exclusive, write*). Trình bày trình tự thực hiện các thao tác khóa và mở khóa tương ứng cho S. Khóa chết có xảy ra không? Vì sao? Nếu khóa chết xảy ra thì trình bày cách giải quyết để các giao tác có thể hoàn thành tất cả các tác vụ trên trước 8g30. (1.25 điểm)

**26.3.** Trình bày nội dung sổ ghi nhật ký trước 8g30. (0.5 điểm)

**26.4.** Giả sử hệ thống thực hiện *checkpoint* lúc 7g40 và tiếp theo gặp sự cố lúc 8g30. Mô tả quá trình phục hồi cho các giao tác nếu kỹ thuật cập nhật trì hoãn được sử dụng. (0.75 điểm)

(Cán bộ coi thi không giải thích gì thêm.)

Ngày: ...../...../2019

Bộ môn/Khoa	Giảng viên ra đề
	TS. Võ Thị Ngọc Châu

**Bảng đối sánh giữa câu hỏi và chuẩn đầu ra môn học:**

Câu hỏi	Chuẩn đầu ra môn học
1	LO.1.1
2	LO.1.2
3	LO.2.1
4	LO.2.1
5	LO.2.1
6	LO.2.2
7	LO.2.2
8	LO.2.2
9	LO.2.2
10	LO.3.1
11	LO.3.1
12	LO.3.3

Câu hỏi	Chuẩn đầu ra môn học
13	LO.3.1
14	LO.4.1
15	LO.4.1
16	LO.4.1
17	LO.4.1
18	LO.4.2
19	LO.4.2
20	LO.4.2
21	LO.4.2
22	LO.4.3
23	LO.4.3
24	LO.4.3

Câu hỏi	Chuẩn đầu ra môn học
25.1	LO.2.1
25.2	LO.3.2
25.3	LO.3.3
25.4	LO.3.3
26.1	LO.4.1
26.2	LO.4.2
26.3	LO.4.1
26.4	LO.4.3



## **PHẦN TRẢ LỜI**

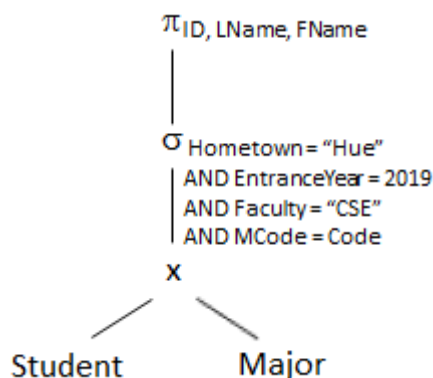
### **I. Phần trắc nghiệm:**

Câu	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15	16	17	18	19	20	21	22	23	24
<b>A</b>				√							√	√		√						√		√		
<b>B</b>			√		√		√		√							√	√						√	
<b>C</b>		√				√									√				√		√			
<b>D</b>	√							√		√			√					√						√
<b>E</b>																								

### **II. Phần tự luận:**

**Câu 25.1.** Student là tập tin có thứ tự với các bản ghi được sắp thứ tự theo giá trị của vùng tin MCode. Major là tập tin không có thứ tự.

**Câu 25.2.** Cây truy vấn tương đương.



**Câu 25.3.** Xử lý và tối ưu hóa dựa trên chi phí cho phép chọn.

Phép toán	Tuần tự	Nhi phân	Chỉ mục
Hometown	5,000	-	$2+1+\lceil 15,000/30 \rceil = 503$
EntranceYear	5,000	-	$1+1+\lceil 15,000*20/100 \rceil = 3,002$
All	5,000	-	503



Họ - Tên: ..... Mã Số Sinh Viên: ..... Mã đề: **W11**

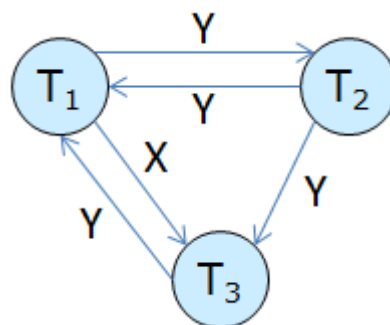
Cho phép chọn đã cho, dùng xử lý thông qua chỉ mục cho phép chọn với điều kiện trên vùng tin Hometown rồi sau đó, kiểm tra cho điều kiện còn lại trên vùng tin EntranceYear trong bộ nhớ chính. Chi phí là 503 truy đạt khối.

**Câu 25.4.** Xử lý và tối ưu hóa dựa trên chi phí cho phép kết.

Single loop (Student – outer)	Single loop (Major – outer)	Hash join
-	$6 + 40 * (2 + \lceil 15,000 / 40 \rceil \lceil 15,000 / 5,000 \rceil)$ $+ \lceil (1/40) * 40 * 15,000 / 2 \rceil = 12,586$	$6 + 5,000 + \lceil (1/40) * 40 * 15,000 / 2 \rceil = 12,506$

Cho phép kết đã cho, dùng hash join để xử lý cho phép kết với chi phí là 12,506 truy đạt khối.

**Câu 26.1.** Lịch biểu đã cho không khả tuần tự hóa vì đồ thị trước sau có chu trình: T1-T2-T3-T1.

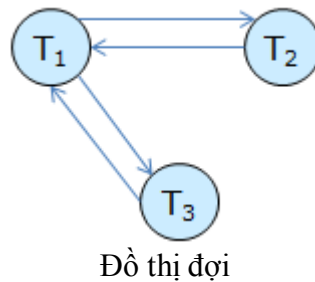


Đồ thị trước sau của lịch biểu S

**Câu 26.2.** Điều khiển tương tranh bằng kỹ thuật khóa hai pha: read\_lock(), write\_lock(), và unlock()

T1	T2	T3
Read_lock(X) $r_1(X)$		
		Read_lock(X) $r_3(X)$
	Read_lock(Y) $r_2(Y)$	
Read_lock(Y) $r_1(Y)$		
	Write_lock(Y) – đợi T1 $w_2(Y, 5, 10)$	
		Read_lock(Y) $r_3(Y)$
		Write_lock(X) – đợi T1 $w_3(X, 3, 9)$
Write_lock(Y) – đợi T2 và T3 => rollback $w_1(Y, 5, 18)$		
		Unlock(X)
		Unlock(Y)
	Write_lock(Y) $w_2(Y, 5, 10)$	
	Unlock(Y)	

Deadlock xảy ra khi T1 thực hiện  $w_1(Y, 5, 18)$ . Khi này, T1 đợi T2 và T3 để có được khóa write\_lock(Y). Trước đó, T2 đợi T1 để có được khóa write\_lock(Y) và T3 đợi T1 để có được khóa write\_lock(X). Giải quyết bằng cách ngưng thực thi và quay lui T1, tiếp tục thực hiện T2 và T3, T1 được khởi động lại.



**Câu 26.3.** Nội dung sổ ghi nhật ký

```

[start_transaction, T1]
[read_item, T1, X]
[start_transaction, T3]
[read_item, T3, X]
[start_transaction, T2]
[read_item, T2, Y]
[read_item, T1, Y]
[write_item, T2, Y, 5, 10]
[read_item, T3, Y]
[commit, T2]
[checkpoint]
[write_item, T3, X, 3, 9]
[write_item, T1, Y, 5, 18]
  
```

Họ - Tên: ..... Mã Số Sinh Viên: ..... Mã đề: **W11**

**Câu 26.4.** Quá trình phục hồi với kỹ thuật cập nhật trì hoãn => lược đồ phục hồi dạng NO-UNDO/REDO

T1: giao tác chưa commit nên không cần xử lý gì cho T1.

T2: giao tác commit trước điểm checkpoint nên các cập nhật đã trong CSDL. Do đó, không cần xử lý gì cho T2.

T3: giao tác chưa commit nên không cần xử lý gì cho T3.

**Thang điểm:**

Câu	Điểm tối đa	Ghi chú
1-24	0.25	Nếu đáp án (E) được chọn và giải thích đúng thì được tính trọn điểm: 0.25 điểm/câu.
25.1.1	0.25	Đúng lưu trữ vật lý của tập tin Student
25.1.2	0.25	Đúng lưu trữ vật lý của tập tin Major
25.2	0.5	Cây truy vấn tương đương đúng (theo các quy tắc biến đổi)
25.3	1.25	0.25 điểm/phương pháp cho mỗi điều kiện, 0.25 điểm/tổng hợp cho AND, 0.25 điểm/phương pháp được chọn đúng
25.4	0.75	0.25 điểm/phương pháp, 0.25 điểm/phương pháp được chọn đúng
26.1.1	0.25	Đúng đặc điểm serializable của lịch biểu
26.1.2	0.25	Giải thích đúng
26.2.1	0.75	Áp khóa đúng cho các giao tác: 0.25 điểm/giao tác
26.2.2	0.25	Mô tả khóa chết
26.2.3	0.25	Giải quyết khóa chết (nếu áp khóa mà không xảy ra khóa chết thì phần này được trọn điểm)
26.3	0.5	Đúng và đầy đủ nội dung số ghi nhật ký, nếu sai 2-6 entry thì bị trừ 0.25 điểm, nếu sai > 6 entry thì không được tính điểm
26.4	0.75	Mô tả đúng phục hồi dạng NO-UNDO/REDO cho mỗi giao tác: 0.25 điểm/giao tác