BÀI TẬP — FIRST-ORDER LOGIC

kèm đáp án giải thích đầy đủ

Intro to AI — Faculty of Data Science & AI, National Economics University
Autumn 2026 · Trong-Nghia Nguyen

Bài tập bám theo deck *First-Order Logic*, sắp xếp đúng trình tự kiến thức trong slide: cú pháp → ngữ nghĩa → dịch câu → phủ định lượng từ → chuyển clause → unification & resolution. Mọi đáp án đã được kiểm bằng code: các giá trị đúng/sai được tính bằng cách duyệt toàn bộ model trên miền nhỏ, các phép thế và bước resolution được chạy bằng một bộ unify/resolve.

Cách dùng tài liệu. Đây là phần **tự học ở nhà**, không giải trên lớp.
1. Làm **trọn một Phần** ra giấy trước, chưa xem lời giải.
2. Toàn bộ đề bài nằm ở **nửa đầu** tài liệu, toàn bộ lời giải ở **nửa sau**.
3. Sai câu nào thì quay lại **đúng slide ghi ở tiêu đề Phần** rồi làm lại câu đó.

Bảng *Tự chấm* ở cuối cho biết sai ở phần nào thì hổng kiến thức gì.

Quy ước ký hiệu

- Toán tử: $\neg$, $\wedge$, $\vee$, $\Rightarrow$, $\Leftrightarrow$. Thứ tự ưu tiên (mạnh → yếu): $\neg > \wedge > \vee > \Rightarrow > \Leftrightarrow$.
- Lượng từ: $\forall$ (với mọi), $\exists$ (tồn tại). **Luôn đặt ngoặc sau lượng từ**: $\forall x\,(\ldots)$. Ngoặc cho biết lượng từ phủ tới đâu.
- **Hằng và vị từ viết hoa chữ đầu** (*Nam*, *Student*); **biến và hàm viết thường** (x, *father*).
- $\{x/Nam\}$ = phép thế: thay x bằng *Nam*. $\bot$ = clause rỗng (mâu thuẫn).
- Hai mẫu phải thuộc lòng: $\forall$ **đi với** $\Rightarrow$, $\exists$ **đi với** $\wedge$.

PHẦN I — ĐỀ BÀI

Làm hết phần này trước khi xem lời giải ở trang 5

1. Cú pháp: term, câu, biểu thức không hợp lệ (slide 6–8)

1A. Phân loại

Với mỗi biểu thức, cho biết nó là **term** (chỉ một đối tượng), **câu nguyên tử** (đúng/sai được), **câu phức** (ghép bằng $\neg\ \wedge\ \vee\ \Rightarrow\ \Leftrightarrow$ hoặc có lượng từ), hay **không hợp lệ**.

Từ vựng: hằng *Nam*, *Mai*, *Hung*; hàm *father*(x); vị từ *Student*(x), *Smart*(x), *Hard*(x), *Friend*(x, y).

1. *Mai*
2. *father*(*father*(*Nam*))
3. *Student*(*Mai*)
4. *Friend*(*Nam*,*father*(*Mai*))
5. *father*(*Nam*) = *Hung*
6. *father*(*Student*(*Nam*))

7. $Student(Nam) \wedge \neg Hard(Nam)$
8. $\forall x\,(Student(x) \Rightarrow Hard(x))$
9. $Hard(Student(Nam))$
10. $Smart(father(x))$

1B. Đọc ngược công thức thành lời

Diễn đạt mỗi công thức bằng một câu tiếng Việt tự nhiên.

11. $\forall x\,(Student(x) \Rightarrow Smart(father(x)))$
12. $\exists x\,(Student(x) \wedge \forall y\,(AISubject(y) \Rightarrow Studies(x, y)))$
13. $\forall x\,(Hard(x) \Rightarrow Student(x))$
14. So sánh hai câu sau, chúng khác nhau ở chỗ nào?
    $\forall x\,\exists y\,Knows(x, y)$ và $\exists y\,\forall x\,Knows(x, y)$

2. Ngữ nghĩa: tính đúng/sai trong một thế giới nhỏ (slide 10–13)

Cho thế giới nhỏ sau. (Quan hệ *Friend* là phần bổ sung so với slide, để luyện lượng từ lồng nhau.)

Người	Student	Hard
An	✓	✓
Bình	✓	×
Chi	×	×

Friend (quan hệ **hai chiều**): An–Bình và Bình–Chi. Viết đầy đủ:

$$Friend = \{(An, Bình),\ (Bình, An),\ (Bình, Chi),\ (Chi, Bình)\}$$

Không ai là bạn của chính mình. An **không** là bạn của Chi.

Với mỗi câu, cho biết **đúng hay sai**, và nêu **nhân chứng** (với ∃) hoặc **phản ví dụ** (với ∀).

1. $\forall x\,(Student(x) \Rightarrow Hard(x))$
2. $\exists x\,(Student(x) \wedge Hard(x))$
3. $\forall x\,(Student(x) \wedge Hard(x))$
4. $\exists x\,(Student(x) \Rightarrow Hard(x))$
5. $\forall x\,(Hard(x) \Rightarrow Student(x))$
6. $\neg\exists x\,(Hard(x) \wedge \neg Student(x))$
7. $\forall x\,\exists y\,Friend(x, y)$
8. $\exists y\,\forall x\,Friend(x, y)$
9. $\exists x\,(Student(x) \wedge \forall y\,(Friend(x, y) \Rightarrow Hard(y)))$
10. $\forall x\,(Student(x) \Rightarrow \exists y\,(Friend(x, y) \wedge Hard(y)))$
11. $\exists x\,(Student(x) \wedge \exists y\,(Friend(x, y) \wedge Hard(y)))$

3. Dịch câu sang FOL (slide 15–16)

Từ vựng dùng chung: $Student(x)$, $Lecturer(x)$, $Hard(x)$ (chăm chỉ), $Course(x)$ (là môn học), $AICourse(x)$, $Takes(x, y)$ (x học môn y), $Teaches(x, y)$ (x dạy môn y), $Likes(x, y)$ (x thích y).

1. Mọi sinh viên đều chăm chỉ.
2. Có một giảng viên chăm chỉ.
3. Không giảng viên nào học môn nào cả.
4. Không phải mọi sinh viên đều học một môn AI nào đó.
5. Chỉ giảng viên mới dạy môn học.
6. Mọi sinh viên đều học ít nhất một môn.
7. Có một sinh viên học tất cả các môn AI.
8. Có một môn mà mọi sinh viên đều học.
9. Không ai thích một giảng viên lười.
10. Mọi giảng viên đều dạy một môn mà có sinh viên theo học.

4. Tìm lỗi trong bản dịch (slide 17)

Mỗi câu dưới đây có một bản dịch **sai**. Hãy (a) chỉ ra công thức sai đó **thật sự nói gì**, (b) nêu một tình huống cụ thể cho thấy nó sai, (c) viết lại cho đúng.

1. "Có một sinh viên FDA chăm chỉ." → $\exists x\,(FDA(x) \Rightarrow Hard(x))$
2. "Mọi sinh viên FDA đều chăm chỉ." → $\forall x\,(FDA(x) \wedge Hard(x))$
3. "Có một sinh viên học tất cả các môn AI." → $\exists x\,\forall y\,(Student(x) \wedge Takes(x, y) \wedge AICourse(y))$
4. "Không sinh viên nào lười." → $\forall x\,(\neg Student(x) \Rightarrow \neg Lazy(x))$
5. "Chỉ sinh viên mới học các môn AI." → $\forall x\,(Student(x) \Rightarrow \exists y\,(AICourse(y) \wedge Takes(x, y)))$

5. Phủ định câu có lượng từ (slide 18)

Đẩy dấu $\neg$ vào **sâu nhất có thể** (chỉ còn $\neg$ đứng trước các vị từ).

> Nhắc lại: khi $\neg$ đi qua lượng từ thì **lượng từ lật**: $\neg\forall x\,P \equiv \exists x\,\neg P$ $\neg\exists x\,P \equiv \forall x\,\neg P$

1. $\neg\forall x\,(Student(x) \Rightarrow Hard(x))$
2. $\neg\exists x\,(Student(x) \wedge Hard(x))$
3. $\neg\forall x\,\exists y\,Friend(x, y)$
4. $\neg\exists x\,(Student(x) \wedge \forall y\,(Course(y) \Rightarrow Takes(x, y)))$
5. $\neg\forall x\,(Lecturer(x) \Rightarrow \exists y\,(Course(y) \wedge Teaches(x, y)))$
6. $\neg\neg\forall x\,(Dog(x) \Rightarrow Barks(x))$

6. Chuyển sang clause (slide 21–22)

Nhắc lại 5 bước: (1) khử $\Leftrightarrow$ · (2) khử $\Rightarrow$ · (3) đẩy $\neg$ vào trong (lượng từ lật) · (4) bỏ lượng từ: $\exists \to$ **đặt tên mới**, rồi xoá $\forall$ · (5) phân phối $\vee$ qua $\wedge$ và tách clause.

Chuyển mỗi câu sau thành tập clause.

1. $\forall x\,(Dog(x) \Rightarrow Barks(x))$
2. $\forall x\,\forall y\,((Owns(x,y) \wedge Dog(y)) \Rightarrow AnimalLover(x))$
3. $\exists x\,(Student(x) \wedge Hard(x))$
4. $\neg\forall x\,(Dog(x) \Rightarrow Barks(x))$
5. $\exists x\,(Student(x) \wedge \forall y\,(AICourse(y) \Rightarrow Takes(x,y)))$
6. $\forall x\,(Lecturer(x) \Rightarrow \neg\exists y\,(Course(y) \wedge Takes(x,y)))$
7. $\forall x\,(Hard(x) \Leftrightarrow Student(x))$

7. Unification và một bước resolution (slide 23)

7A. Unification

Tìm phép thế θ làm hai biểu thức **giống hệt nhau**, hoặc giải thích vì sao **thất bại**.

1. $Knows(Nam, x)$ và $Knows(Nam, Mai)$
2. $Knows(Nam, x)$ và $Knows(y, Lan)$
3. $Knows(Nam, Mai)$ và $Knows(Lan, x)$
4. $P(x, x)$ và $P(A, B)$
5. $P(x)$ và $Q(x)$
6. $Loves(x, father(x))$ và $Loves(Nam, y)$
7. $Friend(x, y)$ và $Friend(y, Nam)$
8. $P(A)$ và $P(A)$

7B. Một bước resolution

Với mỗi cặp clause, cho biết **cặp literal bù**, **phép thế** θ, và **resolvent** (nhớ áp θ cho **toàn bộ** phần còn lại).

9. $\neg Student(x) \vee Smart(father(x))$ và $Student(Nam)$
10. $\neg Dog(x) \vee Barks(x)$ và $Dog(D)$
11. $\neg Owns(x, y) \vee \neg Dog(y) \vee AnimalLover(x)$ và $Owns(Minh, D)$
12. $\neg Cheater(x) \vee \neg Timid(x)$ và $Cheater(A)$
13. $P(x)$ và $\neg P(A)$
14. $Student(x)$ và $Hard(Nam)$

PHẦN II — LỜI GIẢI

Chỉ xem sau khi đã làm hết phần đề bài

L1. Lời giải — Cú pháp (slide 6–8)

1. *Mai* — **term** (hằng). Nó gọi tên một người, không có chuyện đúng hay sai.
2. $father(father(Nam))$ — **term**. Hàm lồng hàm: "bố của bố của Nam", tức ông nội. Hàm nhận term và trả về **một đối tượng**, nên kết quả vẫn là term.
3. $Student(Mai)$ — **câu nguyên tử**. Vị từ áp lên một term → đúng hoặc sai.
4. $Friend(Nam, father(Mai))$ — **câu nguyên tử**. Muốn biết đúng/sai, làm **hai bước**: tính $father(Mai)$ ra một người cụ thể, rồi hỏi Nam có phải bạn của người đó không.
5. $father(Nam) = Hung$ — **câu nguyên tử**. Dấu = ở đây **không phải phép gán** như trong lập trình. Nó là một khẳng định "hai cách gọi này chỉ cùng một người", nên hoàn toàn có thể **sai** (nếu bố Nam là người khác).
6. $father(Student(Nam))$ — **không hợp lệ**. Hàm nhận **term**, nhưng $Student(Nam)$ là một **câu** (đúng/sai). Giống như viết "bố của (Nam là sinh viên)".
7. $Student(Nam) \land \neg Hard(Nam)$ — **câu phức**. Hai câu nguyên tử ghép bằng $\land$ và $\neg$.
8. $\forall x\,(Student(x) \Rightarrow Hard(x))$ — **câu phức**. Có lượng từ.
9. $Hard(Student(Nam))$ — **không hợp lệ**. Lỗi giống câu 6: vị từ *Hard* nhận **đối tượng**, không nhận câu. Muốn nói "sinh viên chăm chỉ" phải viết $Student(x) \land Hard(x)$.
10. $Smart(father(x))$ — **câu nguyên tử**. Có biến x nên chưa xác định đúng/sai cho đến khi x được lượng từ ràng buộc, nhưng về cấu trúc nó vẫn là một vị từ áp lên một term.

> **Mẹo phân biệt hàm với vị từ:** hàm trả về **một đối tượng** (danh từ), vị từ trả về **đúng/sai** (câu). $father(x)$ là "bố của x" — một người. $Student(x)$ là "x là sinh viên" — một khẳng định.

11. "Với mọi x, nếu x là sinh viên thì **bố của** x thông minh." Chú ý: câu nói về **người bố**, không nói về sinh viên.
12. "Có một sinh viên học **tất cả** các môn AI." Lượng từ ngoài là $\exists$ (một sinh viên) nên đi với $\land$; lượng từ trong là $\forall$ (mọi môn AI) nên đi với $\Rightarrow$.
13. "**Chỉ** sinh viên mới chăm chỉ", hay nói cách khác "ai chăm chỉ thì người đó là sinh viên". Đây là mẫu *Only A are B*, mũi tên **đảo chiều** so với cảm giác ban đầu.
14. Đọc lượng từ **từ trái sang phải**, cái nào đứng trước thì "chọn" trước.
    - $\forall x\,\exists y\,Knows(x, y)$: đưa một người x bất kỳ, ta chỉ ra **một** lĩnh vực mà *người đó* biết. Mỗi người có thể biết một lĩnh vực khác nhau.
    - $\exists y\,\forall x\,Knows(x, y)$: ta chỉ ra **một** lĩnh vực trước, rồi **mọi** người đều biết lĩnh vực đó.

    Câu thứ hai **mạnh hơn**: nó suy ra câu thứ nhất, nhưng chiều ngược lại thì không.

L2. Lời giải — Ngữ nghĩa trong thế giới nhỏ (slide 10–13)

#	Câu	Giá trị	Giải thích
1	$\forall x\,(S \Rightarrow H)$	**Sai**	Phản ví dụ **Bình**: là sinh viên nhưng không chăm. Chỉ cần **một** phản ví dụ là câu $\forall$ sai.
2	$\exists x\,(S \wedge H)$	**Đúng**	Nhân chứng **An**: vừa là sinh viên vừa chăm.
3	$\forall x\,(S \wedge H)$	**Sai**	Phản ví dụ **Chi**: không phải sinh viên. Câu này đòi **mọi người** đều là sinh viên chăm — quá mạnh.
4	$\exists x\,(S \Rightarrow H)$	**Đúng**	Nhân chứng **Chi**: $Student(Chi)$ sai, mà F $\Rightarrow$ bất kỳ = T. **Đây là bẫy** — xem ghi chú bên dưới.
5	$\forall x\,(H \Rightarrow S)$	**Đúng**	Người chăm duy nhất là An, mà An là sinh viên. Bình và Chi không chăm nên vế trái sai, cả dòng tự động đúng.
6	$\neg\exists x\,(H \wedge \neg S)$	**Đúng**	"Không có ai vừa chăm vừa không phải sinh viên." Tương đương hoàn toàn với câu 5 — hai cách viết của mẫu *No A is B*.
7	$\forall x\,\exists y\,F(x,y)$	**Đúng**	An có bạn Bình; Bình có bạn An; Chi có bạn Bình. Ai cũng có **ít nhất một** bạn.
8	$\exists y\,\forall x\,F(x,y)$	**Sai**	Cần một người y mà **mọi** người đều kết bạn, kể cả chính y. Nhưng không ai là bạn của chính mình.
9	$\exists x\,(S \wedge \forall y\,(F \Rightarrow H))$	**Sai**	Thử An: bạn duy nhất là Bình, mà Bình không chăm. Thử Bình: bạn là An (chăm) và Chi (không chăm). Không còn sinh viên nào.
10	$\forall x\,(S \Rightarrow \exists y\,(F \wedge H))$	**Sai**	Phản ví dụ **An**: bạn duy nhất của An là Bình, mà Bình không chăm.
11	$\exists x\,(S \wedge \exists y\,(F \wedge H))$	**Đúng**	Nhân chứng **Bình**: Bình là sinh viên và có bạn An là người chăm.

Về câu 4 — bẫy quan trọng nhất của phần lượng từ. Người ta định nói "có một sinh viên chăm chỉ" nhưng lại viết $\exists x\,(Student(x) \Rightarrow Hard(x))$. Câu này đúng **ngay khi tồn tại một người không phải sinh viên**, hoàn toàn không liên quan đến chuyện có ai chăm hay không. Trong thế giới trên, kể cả khi **không ai** chăm chỉ, câu vẫn đúng nhờ Chi. Vì vậy: $\exists$ **phải đi với** $\wedge$.

So sánh câu 7 và 8. Cùng những ký hiệu, chỉ đổi thứ tự hai lượng từ, mà một câu đúng một câu sai.

So sánh câu 10 và 11. Câu 10 dùng $\forall$ nên đòi hỏi **mọi** sinh viên đều có bạn chăm — An làm hỏng. Câu 11 dùng $\exists$ nên chỉ cần **một** sinh viên như vậy — Bình là đủ.

L3. Lời giải — Dịch câu sang FOL (slide 15–16)

1. Mẫu *All A are B*: $\forall x\,(Student(x) \Rightarrow Hard(x))$
2. Mẫu *Some A are B*: $\exists x\,(Lecturer(x) \wedge Hard(x))$
3. Mẫu *No A is B*. Hai cách viết, đã kiểm là **tương đương**:

$$\forall x\,(Lecturer(x) \Rightarrow \neg\exists y\,(Course(y) \wedge Takes(x,y)))$$
$$\equiv \forall x\,\forall y\,((Lecturer(x) \wedge Course(y)) \Rightarrow \neg Takes(x,y))$$

   Dạng thứ hai gọn hơn và tiện hơn khi chuyển sang clause ở Phần 6.

4. Mẫu *Not all A are B*. Đẩy $\neg$ vào trong (xem Phần 5):

$$\neg\forall x\,(Student(x) \Rightarrow \exists y\,(AICourse(y) \wedge Takes(x,y)))$$
$$\equiv \exists x\,(Student(x) \wedge \forall y\,(AICourse(y) \Rightarrow \neg Takes(x,y)))$$

Đọc dạng thứ hai: "có một sinh viên mà **mọi** môn AI người đó đều không học."

5. Mẫu *Only A are B* — **mũi tên đảo chiều**:

$$\forall x\, \forall y\, ((Course(y) \wedge Teaches(x, y)) \Rightarrow Lecturer(x))$$

"Ai mà dạy một môn học thì người đó là giảng viên." Viết $\forall x\, (Lecturer(x) \Rightarrow \dots)$ là **sai chiều** (xem Phần 4, câu 5).

6. Ngoài $\forall$ đi với $\Rightarrow$, trong $\exists$ đi với $\wedge$:

$$\forall x\, (Student(x) \Rightarrow \exists y\, (Course(y) \wedge Takes(x, y)))$$

7. Ngoài $\exists$ đi với $\wedge$, trong $\forall$ đi với $\Rightarrow$:

$$\exists x\, (Student(x) \wedge \forall y\, (AICourse(y) \Rightarrow Takes(x, y)))$$

8. Chú ý: **không phải** đổi chỗ lượng từ của câu 6.

$$\exists y\, (Course(y) \wedge \forall x\, (Student(x) \Rightarrow Takes(x, y)))$$

Câu 6 và câu 8 khác nhau thế nào? Đã kiểm bằng model cụ thể: An học môn C_1, Bình học môn C_2, không ai học môn của người kia.
- Câu 6 ($\forall x\, \exists y$) **đúng**: mỗi sinh viên đều có môn của mình.
- Câu 8 ($\exists y\, \forall x$) **sai**: không có môn nào mà cả An lẫn Bình cùng học.

Đây là cặp $\forall\exists$ / $\exists\forall$ của slide 13, lần này trong một ngữ cảnh thật.

9. Mẫu *No A is B* với hai biến:

$$\forall x\, \forall y\, ((Lecturer(y) \wedge \neg Hard(y)) \Rightarrow \neg Likes(x, y))$$

Đọc: "với mọi x và y, nếu y là giảng viên lười thì x không thích y."

10. Ba lượng từ lồng nhau:

$$\forall x\, (Lecturer(x) \Rightarrow \exists y\, (Course(y) \wedge Teaches(x, y) \wedge \exists z\, (Student(z) \wedge Takes(z, y))))$$

Từng lớp vẫn theo đúng mẫu của nó: $\forall$ ngoài cùng đi với $\Rightarrow$, hai $\exists$ bên trong đều đi với $\wedge$.

L4. Lời giải — Tìm lỗi trong bản dịch (slide 17)

Mọi phản ví dụ dưới đây đã được tìm và kiểm bằng code: trong mỗi tình huống, công thức sai và công thức đúng cho **hai giá trị khác nhau**.

1. Công thức sai nói: "**tồn tại một người** không phải sinh viên FDA, **hoặc** chăm chỉ."
   - *Phản ví dụ*: $FDA = \{An\}$, $Hard = \{\}$ (An là sinh viên FDA, không ai chăm chỉ). Công thức sai cho **đúng** (nhờ Bình không phải sinh viên FDA), công thức đúng cho **sai**.
   - *Sửa*: $\exists x\, (FDA(x) \wedge Hard(x))$.
2. Công thức sai nói: "**mọi người trên đời** đều là sinh viên FDA và đều chăm chỉ."
   - *Phản ví dụ*: $FDA = \{An\}$, $Hard = \{An\}$. Câu đúng cho **đúng**, nhưng công thức sai cho **sai** vì Bình không phải sinh viên FDA.
   - *Sửa*: $\forall x\, (FDA(x) \Rightarrow Hard(x))$.
3. $\forall y$ bị đặt sai chỗ: nó phủ cả $Student(x)$ lẫn $AICourse(y)$, nên công thức nói "**mọi đối tượng trên đời** đều là môn AI và x học tất cả".

- *Phản ví dụ*: miền $\{An, AI_1, AI_2\}$, $Student = \{An\}$, $AICourse = \{AI_1, AI_2\}$, An học cả hai môn. Câu đúng cho **đúng**, công thức sai cho **sai** (vì chính An không phải môn AI).
- *Sửa*: $\exists x\,(Student(x) \land \forall y\,(AICourse(y) \Rightarrow Takes(x, y)))$ — đẩy $\forall y$ vào trong và dùng $\Rightarrow$.

4. Công thức sai nói: "ai **không phải** sinh viên thì không lười" — nói về người ngoài, không nói gì về sinh viên.
   - *Phản ví dụ*: $Student = \{An\}$, $Lazy = \{Bình\}$. Thực tế không sinh viên nào lười nên câu đúng cho **đúng**; công thức sai cho **sai** vì Bình không phải sinh viên mà lại lười.
   - *Sửa*: $\forall x\,(Student(x) \Rightarrow \neg Lazy(x))$.
5. Mũi tên ngược chiều. Công thức sai nói "mọi sinh viên **đều** học một môn AI" — một ý hoàn toàn khác.
   - *Phản ví dụ*: $Student = \{An, Bình\}$, $AICourse = \{AI_1\}$, chỉ An học AI_1. Câu đúng cho **đúng** (người duy nhất học môn AI là An, mà An là sinh viên); công thức sai cho **sai** vì Bình là sinh viên mà không học môn AI nào.
   - *Sửa*: $\forall x\,\forall y\,((AICourse(y) \land Takes(x, y)) \Rightarrow Student(x))$.

Cách tự kiểm tra bản dịch của mình: đọc ngược công thức thành tiếng Việt **từng chữ một**, đừng đọc theo ý mình muốn nói. Nghe lạ tai là sai.

L5. Lời giải — Phủ định câu có lượng từ (slide 18)

Cả 5 tương đương đầu đã được kiểm bằng cách duyệt **toàn bộ** model trên miền 3 phần tử.

1. $\exists x\,(Student(x) \land \neg Hard(x))$
   Các bước: $\neg\forall x\,(\neg S \lor H) \to \exists x\,\neg(\neg S \lor H) \to \exists x\,(S \land \neg H)$.
   Đọc: "không phải mọi sinh viên đều chăm" = "**có** một sinh viên không chăm". Chú ý $\Rightarrow$ biến thành $\land$ khi phủ định.
2. $\forall x\,(Student(x) \Rightarrow \neg Hard(x))$
   Các bước: $\forall x\,\neg(S \land H) \to \forall x\,(\neg S \lor \neg H) \to$ viết lại thành $\Rightarrow$.
   Đọc: "không có sinh viên nào chăm" = "mọi sinh viên đều không chăm".
3. $\exists x\,\forall y\,\neg Friend(x, y)$
   Cả hai lượng từ đều lật. Đọc: "không phải ai cũng có bạn" = "có một người mà **mọi** người đều không là bạn của họ".
4. $\forall x\,(Student(x) \Rightarrow \exists y\,(Course(y) \land \neg Takes(x, y)))$
   Các bước: $\neg$ qua $\exists x \to \forall x$; $\neg(S \land \dots) \to S \Rightarrow \neg(\dots)$; rồi $\neg$ qua $\forall y \to \exists y$; $\neg(C \Rightarrow T) \to C \land \neg T$.
   Đọc: "không có sinh viên nào học hết mọi môn" = "mọi sinh viên đều có **ít nhất một** môn không học".
5. $\exists x\,(Lecturer(x) \land \forall y\,(Course(y) \Rightarrow \neg Teaches(x, y)))$
   Đọc: "không phải giảng viên nào cũng dạy một môn" = "có một giảng viên **không dạy môn nào cả**".
6. $\forall x\,(Dog(x) \Rightarrow Barks(x))$
   Hai dấu $\neg$ triệt tiêu nhau. Trường hợp này xuất hiện thật khi phủ định một kết luận vốn đã có dạng phủ định — chính là bước $\neg\alpha$ của Example 1 trên slide 26.

L6. Lời giải — Chuyển sang clause (slide 21–22)

Mọi kết quả đã được kiểm bằng code: các câu chỉ có $\forall$ được kiểm **tương đương logic** với tập clause; các câu có đặt tên mới được kiểm **đồng thoả mãn** (câu gốc thoả được khi và chỉ khi tập clause thoả được).

1. Khử $\Rightarrow$ rồi bỏ $\forall$:
   $$\neg Dog(x) \lor Barks(x) \qquad \textbf{— 1 clause}$$

2. $\neg(Owns \wedge Dog) \vee AnimalLover$ rồi De Morgan:

$$\neg Owns(x,y) \vee \neg Dog(y) \vee AnimalLover(x) \qquad \textbf{— 1 clause}$$

Mẫu chung: luật có **AND ở vế IF** luôn cho một clause gồm các literal mang dấu ¬, cộng với kết luận.

3. Có ∃ và **không có** ∀ **nào bao ngoài** → đặt tên mới A. Phép ∧ tách thành hai clause:

$$Student(A) \qquad \text{và} \qquad Hard(A) \qquad \textbf{— 2 clause}$$

A phải là một tên **chưa xuất hiện ở đâu khác**. Nếu viết *Nam* thay cho A, ta đã vô tình khẳng định thêm rằng *Nam* chính là người sinh viên chăm đó.

4. Đây là câu phải làm **đúng thứ tự bước**:
   - Bước 2 (khử ⇒): $\neg\forall x\,(\neg Dog(x) \vee Barks(x))$
   - Bước 3 (đẩy ¬ vào): $\exists x\,(Dog(x) \wedge \neg Barks(x))$ — ∀ đã lật thành ∃
   - Bước 4 (đặt tên mới D): $Dog(D) \wedge \neg Barks(D)$

$$Dog(D) \qquad \text{và} \qquad \neg Barks(D) \qquad \textbf{— 2 clause}$$

Nếu làm bước 4 trước bước 3, ta sẽ nhìn thấy $\forall x$ và xoá nó đi — **sai**, vì dấu ¬ phía trước biến nó thành ∃.

5. $\exists x$ nằm ngoài cùng → đặt tên mới C. $\forall y$ nằm trong, cứ để rồi xoá ở bước 4:
   - Bước 2: $\exists x\,(Student(x) \wedge \forall y\,(\neg AICourse(y) \vee Takes(x,y)))$
   - Bước 4: $Student(C) \wedge (\neg AICourse(y) \vee Takes(C,y))$

$$Student(C) \qquad \text{và} \qquad \neg AICourse(y) \vee Takes(C,y) \qquad \textbf{— 2 clause}$$

Chú ý y vẫn còn trong clause thứ hai và được hiểu ngầm là "với mọi y".

6. Dấu ¬ đứng trước ∃ nên ∃ lật thành ∀:
   - Bước 2: $\neg Lecturer(x) \vee \neg\exists y\,(Course(y) \wedge Takes(x,y))$
   - Bước 3: $\neg Lecturer(x) \vee \forall y\,(\neg Course(y) \vee \neg Takes(x,y))$

$$\neg Lecturer(x) \vee \neg Course(y) \vee \neg Takes(x,y) \qquad \textbf{— 1 clause}$$

Câu này **không cần đặt tên mới**, dù nhìn thấy ∃ trong đề. Đây là lý do bước 3 phải làm trước bước 4.

7. Bước 1 — khử ⇔ thành hai chiều: $\forall x\,((Hard(x) \Rightarrow Student(x)) \wedge (Student(x) \Rightarrow Hard(x)))$

$$\neg Hard(x) \vee Student(x) \qquad \text{và} \qquad \neg Student(x) \vee Hard(x) \qquad \textbf{— 2 clause}$$

L7. Lời giải — Unification và resolution (slide 23)

Toàn bộ kết quả dưới đây do một bộ unify/resolve chạy ra.

7A. Unification

#	θ	Giải thích
1	$\{x/Mai\}$	Chỉ cần thay một biến.
2	$\{y/Nam,\ x/Lan\}$	Thay ở **cả hai phía**: vị trí 1 cho y = *Nam*, vị trí 2 cho x = *Lan*.
3	**Thất bại**	Vị trí 1 đòi *Nam* khớp *Lan*. Cả hai đều là **hằng**, mà hằng thì không thay được. Chỉ biến mới thay được.
4	**Thất bại**	Vị trí 1 cho $x = A$, vị trí 2 cho $x = B$. **Cùng một biến** không thể vừa là A vừa là B.
5	**Thất bại**	Hai **vị từ khác nhau** (P và Q). Unify chỉ làm việc khi cùng tên vị từ và cùng số đối số.

#	θ	Giải thích
6	$\{x/Nam,\ y/father(Nam)\}$	Một biến có thể được thay bằng cả một **term có hàm**. Sau khi $x = Nam$, vế trái thành $Loves(Nam, father(Nam))$, nên y phải là $father(Nam)$.
7	$\{x/Nam,\ y/Nam\}$	Vị trí 1 cho $x = y$; vị trí 2 cho $y = Nam$. Dây chuyền lại thành $x = Nam$ luôn. Hai biến **gộp làm một**.
8	$\{\,\}$	Hai biểu thức đã giống hệt nhau, không cần thay gì. Phép thế rỗng vẫn là một unifier hợp lệ.

7B. Một bước resolution

#	Cặp bù	θ	Resolvent
9	$\neg Student(x)\ /\ Student(Nam)$	$\{x/Nam\}$	$Smart(father(Nam))$
10	$\neg Dog(x)\ /\ Dog(D)$	$\{x/D\}$	$Barks(D)$
11	$\neg Owns(x, y)\ /\ Owns(Minh, D)$	$\{x/Minh,\ y/D\}$	$\neg Dog(D) \vee AnimalLover(Minh)$
12	$\neg Cheater(x)\ /\ Cheater(A)$	$\{x/A\}$	$\neg Timid(A)$
13	$P(x)\ /\ \neg P(A)$	$\{x/A\}$	$\bot$ (clause rỗng)
14	—	—	**Không resolve được**

Ghi chú câu 9 — lỗi hay gặp nhất khi làm bài. Sau khi bỏ cặp bù, phần còn lại là $Smart(father(x))$. Phải **áp θ cho phần còn lại** để ra $Smart(father(Nam))$. Nhiều bạn giữ nguyên $Smart(father(x))$ và bước sau sẽ sai theo.

- **Câu 11**: θ có **hai** phép thay, và cả hai đều phải áp vào literal còn lại: $\neg Dog(y)$ thành $\neg Dog(D)$, còn $AnimalLover(x)$ thành $AnimalLover(Minh)$.
- **Câu 13**: hai clause đơn bù nhau cho ra **clause rỗng**. Đây chính là đích đến của mọi chứng minh bằng resolution.
- **Câu 14**: hai clause không có **cặp literal bù** nào: một bên là *Student*, bên kia là *Hard*, hai vị từ khác nhau. Không phải cặp clause nào cũng resolve được với nhau; chọn đúng cặp là việc của chiến lược *unit preference* và *set of support*.

Đáp án tóm tắt

Phần	Đáp án
1A	1 term · 2 term · 3 nguyên tử · 4 nguyên tử · 5 nguyên tử · **6 không hợp lệ** · 7 phức · 8 phức · **9 không hợp lệ** · 10 nguyên tử
2	1 S · 2 Đ · 3 S · **4 Đ** · 5 Đ · 6 Đ · 7 Đ · 8 S · 9 S · 10 S · 11 Đ
3	xem lời giải — chú ý câu 5 (đảo mũi tên) và cặp câu 6 / 8
4	cả 5 bản dịch đều sai; mỗi câu có một phản ví dụ cụ thể
5	1 $\exists(S \wedge \neg H)$ · 2 $\forall(S \Rightarrow \neg H)$ · 3 $\exists\forall\neg F$ · 4 $\forall(S \Rightarrow \exists(C \wedge \neg T))$ · 5 $\exists(L \wedge \forall(C \Rightarrow \neg Te))$ · 6 không đổi
6	số clause: 1→1 · 2→1 · 3→2 · 4→2 · 5→2 · **6→1** · 7→2
7A	1, 2, 6, 7, 8 thành công · **3, 4, 5 thất bại**
7B	14 không resolve được; 13 ra $\perp$

Tự chấm

Sai ở	Nghĩa là hổng chỗ nào	Xem lại
Phần 1	Chưa phân biệt **hàm** (trả về đối tượng) với **vị từ** (trả về đúng/sai)	slide 6–8
Phần 2 câu 4	Chưa thấy vì sao $\exists$ **không đi với** $\Rightarrow$	slide 12
Phần 2 câu 7–8	Chưa nắm **thứ tự lượng từ**	slide 13
Phần 3 câu 5	Chưa nhớ mẫu *Only A are B* đảo mũi tên	slide 15
Phần 3 câu 6/8	Nhầm $\forall\exists$ với $\exists\forall$	slide 13, 16
Phần 4	Dịch theo ý muốn nói, chưa đọc ngược lại để kiểm	slide 17
Phần 5	Quên **lật lượng từ** khi $\neg$ đi qua	slide 18
Phần 6 câu 4, 6	Làm **bước 4 trước bước 3**	slide 21–22
Phần 6 câu 3, 5	Chưa nắm quy tắc **đặt tên mới** cho $\exists$	slide 22
Phần 7A câu 3, 4	Tưởng hằng cũng thay được, hoặc cho một biến hai giá trị	slide 23
Phần 7B câu 9, 11	**Quên áp θ cho phần còn lại** của resolvent	slide 23

Làm xong tài liệu này thì chuyển sang **ba bài chứng minh đầy đủ** (file `BaiTap_FOL_DapAn.md`) — ghép tất cả các kỹ năng trên lại: dịch câu → lập clause → chạy resolution tới $\perp$.