Theoretical, Question 2:

∃x ( IsEasy(course) →  IsHappy(x)), ∀x ( HasFinal(course) →  ¬IsHappy(x)) ⊨? HasFinal(course) →  ¬IsEasy(course)

∃x ( IsEasy(course) →  IsHappy(x)) ≡ ∃x (¬IsEasy(course) v IsHappy(x))

∀x ( HasFinal(course) →  ¬IsHappy(x)) ≡ ∀x(¬HasFinal(course) v ¬IsHappy(x))

S = HasFinal(course) →  ¬IsEasy(course) ≡ ¬HasFinal(course) v ¬IsEasy(course)

Using contradiction, assume ¬S

¬S = HasFinal(course) ^ IsEasy(course)

KB

1. ∃x (¬IsEasy(course) v IsHappy(x))

2. ≡ (¬HasFinal(course) v ¬IsHappy(x))

3. HasFinal(course)

4. IsEasy(course)

Skolemize

1 => ¬IsEasy(course) v IsHappy(A)

Drop Universal Quantifiers

2 => ¬HasFinal(course) v ¬IsHappy(x))

Resolve 1,3

5. IsHappy(A)

Resolve 2,5 (x=A Unification binding)

6. ¬HasFinal(course)

Resolve 3,6

7. {}

Empty clause, therefore contradiction.

Therefore, S is true, that is “If a course has a final, the course is not easy”