COSC 290 Discrete Structures

Lecture 8: Nested quantifiers

Prof. Michael Hay

Friday, Feb. 9, 2018

Colgate University

Plan for today

- 1. Practice: nested and negated quantifiers
- 2. Logistics

Practice: nested and negated

True love, expressed mathematically

Predicate $loves(p_1, p_2)$ means " p_1 loves p_2 ." We can express the loves predicate visually using a graph.

Nodes are individuals. Edge from p_1 to p_2 indicates loves(u, v).

Example	
The proposition	
$loves(Alice, Darmesh) \land loves(Bob, Chloe) \land loves(Chloe, Darmesh)$	
can be shown as.	
	figs/loveso.pdf

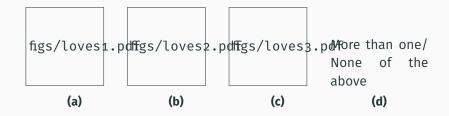
2

Poll: nested quantifiers, part 1

Predicate $loves(p_1, p_2)$ means " p_1 loves p_2 ," shown by an arrow from p_1 to p_2 .

For which figure(s) is the following proposition true:

$$\forall p_1 \in P : \exists p_2 \in P : loves(p_1, p_2)$$

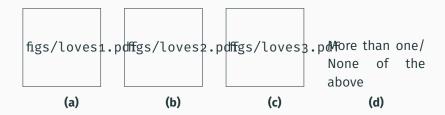


Poll: nested quantifiers, part 2

Predicate $loves(p_1, p_2)$ means " p_1 loves p_2 ," shown by an arrow from p_1 to p_2 .

For which figure(s) is the following proposition true:

$$\exists p_2 \in P : \forall p_1 \in P : loves(p_1, p_2)$$



Poll: showing a proposition is false

Consider the following proposition

$$\forall p_1 \in P : \exists p_2 \in P : loves(p_1, p_2)$$

How could we show that this proposition is false?

- A) Show there is a person who loves everyone
- B) Show there is a person who loves no one
- C) Show there is a person who nobody loves
- D) Show there is a person who everyone loves
- E) Other/more/none

Poll: negating nested quantifiers

Consider the following proposition

$$\varphi \coloneqq \exists p_2 \in P : \forall p_1 \in P : loves(p_1, p_2)$$

What is the correct negation of φ ? In other words, which of the following is logically equivalent to $\neg \varphi$?

- A) $\forall p_2 \forall p_1 \neg loves(p_1, p_2)$
- B) $\forall p_2 \exists p_1 \neg loves(p_1, p_2)$
- C) $\exists p_2 \forall p_1 \neg loves(p_1, p_2)$
- D) $\forall p_2 \exists p_1 \neg loves(p_2, p_1)$
- E) Other/more/none



Logistics

Next topic on schedule: proofs!

Reading does not align neatly with lecture plan.

We will spend some time (roughly 1 lecture each) on several proof techniques:

- · direct proof
- proof by contrapositive
- · proof by contradiction
- · proof by cases

We will *return* to Ch. 4.2 to apply these techniques to Hamming codes.