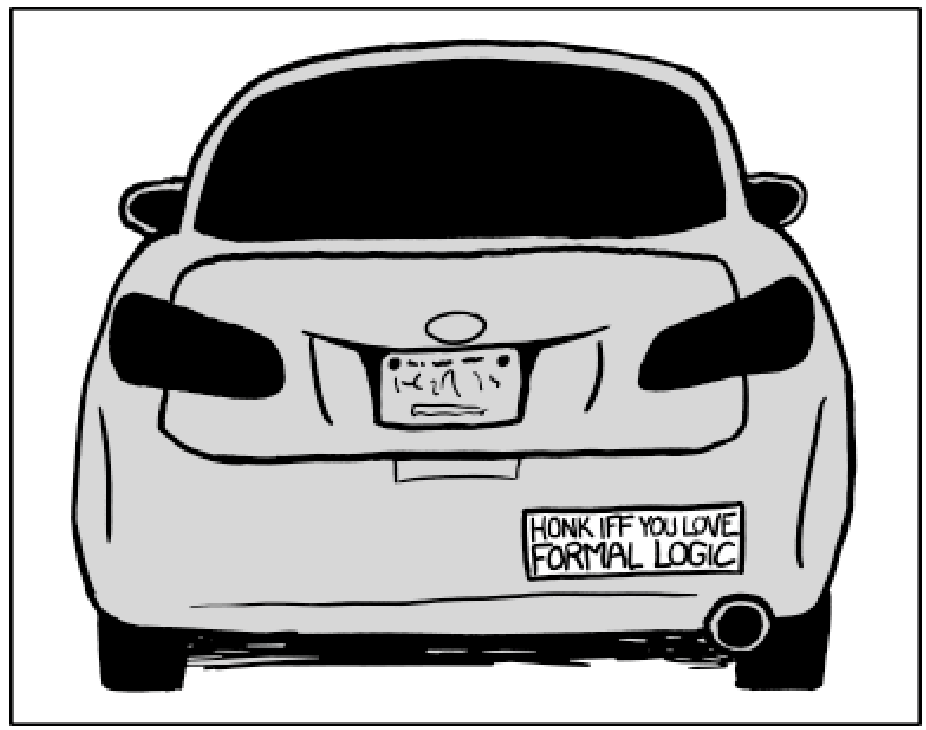
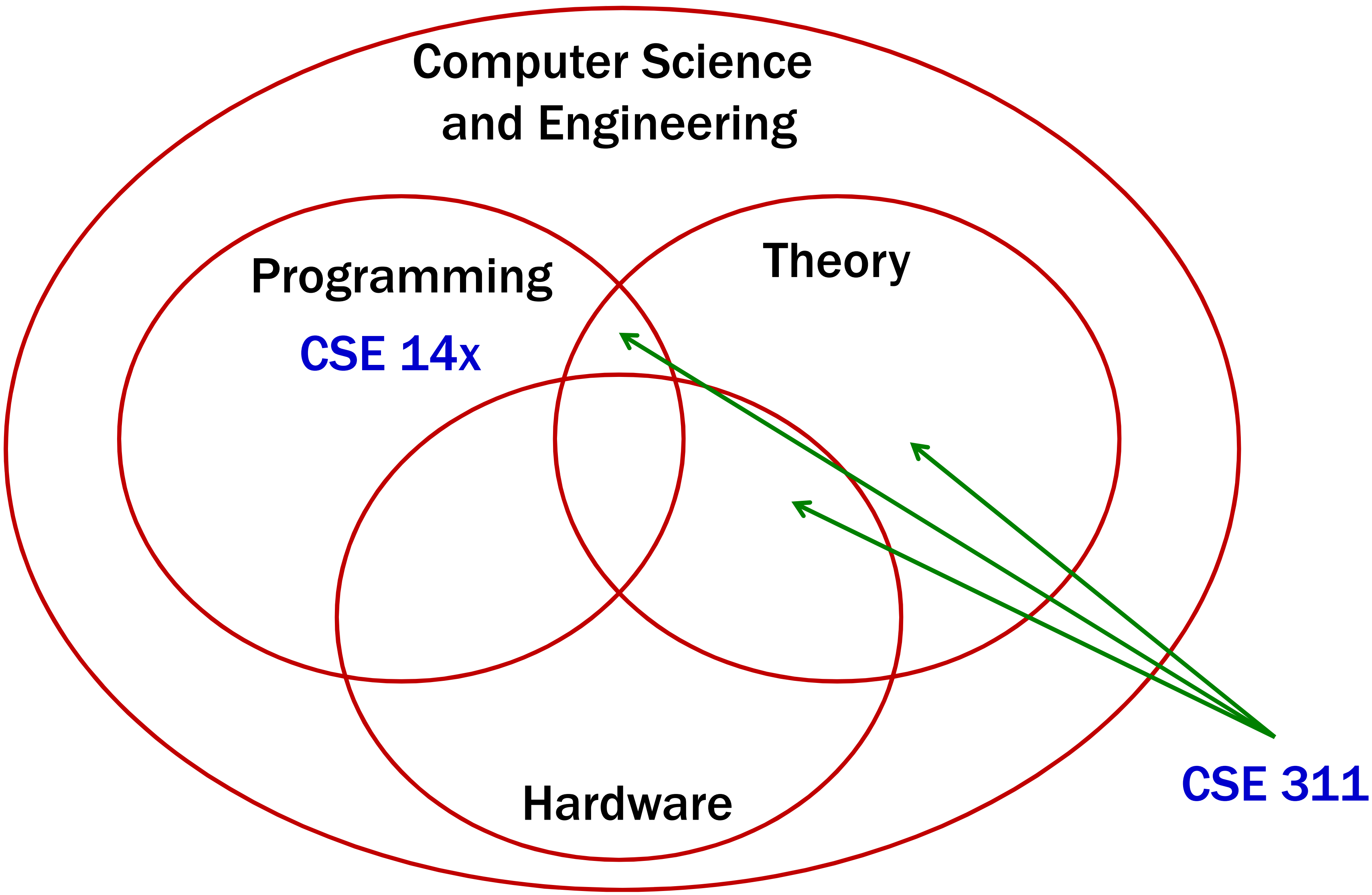
CSE 311: Foundations of Computing I

Lecture 1: Propositional Logic



About CSE 311

Some Perspective



About the Course

We will study the *theory* needed for CSE:

Logic:

How can we describe ideas *precisely*?

Formal Proofs:

How can we be *positive* we're correct?

Number Theory:

How do we keep data *secure?*

Relations/Relational Algebra:

How do we store information?

Finite State Machines:

How do we design hardware and software?

Turing Machines:

Are there problems computers *can't* solve?

About the Course

Will help you become a better programmer

By the end of the course, you will have the tools for….

- reasoning about difficult problems
- automating difficult problems
- communicating ideas, methods, objectives

and will understand fundamental structures of CS

Course Logistics

Instructor

Paul Beame

MWF 1:30-2:20 in CSE2 G01

Office Hours (tentative):

M 2:30-4:00 and WF 2:30-3:00 in CSE 668

TAs

Teaching Assistants:

Siddharth Iyer
Suraj Jagadeesh
Karishma Mandyam
Josh Shin
Xiaoyue Sun
Jason Waataja

Section:

Thursdays

– starting this week

Office Hours: TBD

(Optional) Book:

Rosen: Readings for 6^{th} or 7^{th} editions.

Many used copies available

Good for practice with solved problems

Course Webpage

CSE 311: Foundations of Computing I

Winter, 2020

Paul Beame

> **MWF 1:30-2:20,** CSE2 G01
> Office hours: TBA
> CSE 668

Email and discussion:

email list: cse311a_wi20 [archives]
Please send any e-mail about the course to **cse311-staff@cs**.

Textbook:

There is no required text for the course. Especially over the first 6-7 weeks of the course, the following textbook can be a useful companion: Rosen, *Discrete Mathematics and Its Applications*, McGraw-Hill. There are many editions of this book and lots of used copies available; new copies are extremely expensive. A copy should be available on short-term loan from the Engineering Library.

Lectures

#	date	topic	slides	inked	reading (Rosen)
1	Mon, Jan 6	Propositional Logic			1.1, 1.2 (7th) 1.1 (6th)
2	Wed, Jan 8	Logical Equivalence/Gates			1.1-1.3 (7th) 1.1-1.2 (6th)
3	Fri, Jan 10	More Logic/Circuits			12.1-12.3 (7th) 11.1-11.3 (6th)
4	Mon, Jan 13	Boolean Algebra/Circuits			12.1-12.3 (7th) 11.1-11.3 (6th)
5	Wed, Jan 15	Canonical Forms, Predicate Logic			1.4-1.5 (7th) 1.3-1.4 (6th)
6	Fri, Jan 17	Predicate Logic			1.6-1.7 (7th) 1.5-1.7 (6th)
	Mon, Jan 20	**Martin Luther King Day NO CLASS**			
7	Wed, Jan 22	Logical Inference and Proofs			1.6-1.7 (7th) 1.5-1.7 (6th)
8	Fri, Jan 24	Predicate Logic Proofs			1.6-1.7 (7th) 1.5-1.7 (6th)
9	Mon, Jan 27	Set Theory			2.1-2.3 (6th, 7th)
10	Wed, Jan 29	Modular Arithmetic			4.1-4.2 (7th) 3.4-3.5 (6th)
11	Fri, Jan 31	Applications of Mod, Number Theory, Factoring			4.1-4.3 (7th) 3.4-3.6 (6th)
12	Mon, Feb 3	GCD, Euclid's Algorithm, Modular Equations			4.3-4.4 (7th), 3.5-3.7 (6th)
13	Wed, Feb 5	Induction			5.1 (7th), 4.1 (6th)
14	Fri, Feb 7	More Induction			5.1 (7th), 4.1 (6th)

TA	Office hours	Room
Siddharth Iyer		
Suraj Jagadeesh		
Karishma Mandyam		
Josh Shin		
Xiaoyue Sun		
Jason Waataja		

Section	Day/Time	Room
AA	Th 11:30-12:20	DEN 213
AB	Th 12:30-1:20	LOW 220
AC	Th 1:30-2:20	LOW 101
AD	Th 2:30-3:20	LOW 101

Section Materials	Date	Problems	Solns
01	Jan 9		

Homework

Exams:

- **Midterm exam**:
  In class, Friday 14-Feb-2020,
  Closed book. Closed notes.
- **Final exam**:
  The final exam will at the officially scheduled time
  Monday 16-Mar-2020, 2:30-4:20 pm.

All Course Information @cs.uw.edu/311

Work

Homework:

- Due WED at 11:00 pm online (Gradescope)
- Write up individually
- Extra Credit

Grading (roughly):

- 50% Homework
- 15-20% Midterm
- 30-35% Final Exam

Exams:

- Midterm in class on Friday, Feb 14
- Final exam:
  - Monday, March 16 2:30-4:20 pm

Communication

- You are already on the class e-mail list
  - Major announcements here, archive reachable from the course webpage
- If you want to email to us (me & TAs):

  cse311-staff@cs.washington.edu
- Discussion board
  - accept invitation to Ed class discussion board

About grades…

- Grades were very important up until now…

About grades…

- Grades were very important up until now

- Grades are much less important going forward
  - companies care much more about your interviews
  - grad schools care much more about recommendations

About grades…

- Grades were very important up until now
- Grades are much less important going forward
  - companies care much more about your interviews
  - grad schools care much more about recommendations
- Understanding the material is much more important
  - interviews test your knowledge from these classes
  - good recommendations involve knowledge beyond the classes

About grades…

- Grades were very important up until now
- Grades are much less important going forward
  - companies care much more about your interviews
  - grad schools care much more about recommendations
- Understanding the material is much more important
  - interviews test your knowledge from these classes
  - good recommendations involve knowledge beyond the classes
- Please <u>relax</u> and focus on learning

Please calm down about grades

- Most time spent on questions about grading issues is not worthwhile to either the student or teacher
- Try to avoid asking "will I lose points if…"
- If the thought of losing points worries you, show more work
  - no sense having a 30 minute discussion to save 10 minutes

Collaboration Policy

- Collaboration with others is encouraged
- BUT you must:
  - list anyone you work with
  - turn in only your own work
- Recommended approach for group work
  - do not leave with any solution written down or photographed
  - wait 30 minutes before writing up your solution
- See Allen School Academic Misconduct policy also

No Late Days

- To be accepted, late submission (with good reason) must be arranged in advance 48 hours before the deadline

If you are worried about Mathy aspects of 311

- Associated 1-credit CR/NC workshop
  - CSE 390ZA (not yet available for enrollment)
  - Extra collaborative practice on 311 concepts, study skills, a small amount of assigned work
  - 1.5 hours Thursdays 3:30 pm
  - Full attendance is required, else NC
  - NOT for help with 311 homework
- Anyone in 311 can sign up but enrollment is limited

Getting used to being formal

As problems we deal with get harder we need stronger tools…

Formalism is a tool we apply when problems get difficult

- helps us get through without making mistakes
- sometimes even gives “turn the crank” solutions

Propositional Logic

What is logic and why do we need it?

Logic is a language, like English or Java, with its own

- words and rules for combining words into sentences (syntax)
- ways to assign meaning to words and sentences (semantics)

Why learn another language when we know English and Java already?

Why not use English?

- Turn right here…

- Buffalo buffalo Buffalo buffalo buffalo buffalo Buffalo buffalo

- We saw her duck

Why not use English?

– Turn right here…

Does “right” mean the direction or now?

– Buffalo buffalo Buffalo buffalo buffalo buffalo Buffalo buffalo

This means “Bison from Buffalo, that bison from Buffalo bully, themselves bully bison from Buffalo.

– We saw her duck

Does “duck” mean the animal or crouch down?

Why not use English?

- Turn right here…

  Does "right" mean the direction or now?

- Buffalo buffalo Buffalo buffalo buffalo buffalo Buffalo buffalo

  This means "Bison from Buffalo, that bison from Buffalo bully, themselves bully bison from Buffalo.

- We saw her duck

  Does "duck" mean the animal or crouch down?

Natural languages can be imprecise

Why not use Java?

What does this code do:

```
public static boolean mystery(int x) {
  for (int r = 2; r < x; r++) {
    for (int q = 2; q < x; q++) {
      if (r*q == x)
        return false;
      }
    }
  }
  return x > 1;
}
```

Why not use Java?

What does this code do:

```
public static boolean mystery(int x) {
  for (int r = 2; r < x; r++) {
    for (int q = 2; q < x; q++) {
      if (r*q == x)
        return false;
      }
    }
  }
  return x > 1;
}
```

Determines if x is a prime number

Why not use Java?

What does this code do:

```
public static boolean mystery(int x) {
  for (int r = 2; r < x; r++) {
    for (int q = 2; q < x; q++) {
      if (r*q == x)
        return false;
      }
    }
  }
  return x > 1;
}
```

Determines if x is a prime number

Programming languages can be verbose

Why learn a new language?

We need a language of reasoning to

- state sentences more precisely
- state sentences more concisely
- understand sentences more quickly

Propositions: building blocks of logic

A ***proposition*** is a statement that

- is either true or false
- is "well-formed"

Propositions: building blocks of logic

A ***proposition*** is a statement that

- is either true or false
- is "well-formed"

All cats are mammals

true

All mammals are cats

false

Are These Propositions?

$2 + 2 = 5$

$x + 2 = 5$

Akjsdf!

Who are you?

Every positive even integer can be written as the sum of two primes.

Are These Propositions?

$2 + 2 = 5$

This is a proposition. It's okay for propositions to be false.

$x + 2 = 5$

Not a proposition. Doesn't have a fixed truth value

Akjsdf!

Not a proposition because it's gibberish.

Who are you?

This is a question which means it doesn't have a truth value.

Every positive even integer can be written as the sum of two primes.

This is a proposition. We don't know if it's true or false, but we know it's one of them!

A first application of logic

"If I were to ask you out, would your answer to that question be the same as your answer to this one?"

Propositions

We need a way of talking about *arbitrary* ideas…

Propositional Variables: $p, q, r, s, \ldots$

Truth Values:

- **T** for **true**
- **F** for **false**

A Compound Proposition

"Garfield has black stripes if he is an orange cat and likes lasagna, and he is an orange cat or does not like lasagna"

We'd like to *understand* what this proposition means.

A Compound Proposition

"Garfield has black stripes if he is an orange cat and likes lasagna, and he is an orange cat or does not like lasagna"

We'd like to *understand* what this proposition means.

First find the simplest (atomic) propositions:

p "Garfield has black stripes"

q "Garfield is an orange cat"

r "Garfield likes lasagna"

A Compound Proposition

"Garfield has black stripes if he is an orange cat and likes lasagna, and he is an orange cat or does not like lasagna"

We'd like to *understand* what this proposition means.

First find the simplest (atomic) propositions:

p "Garfield has black stripes"

q "Garfield is an orange cat"

r "Garfield likes lasagna"

(p if (q and r)) and (q or (not r))

Logical Connectives

Negation (not)	$\neg p$
Conjunction (and)	$p \wedge q$
Disjunction (or)	$p \vee q$
Exclusive Or	$p \oplus q$
Implication	$p \longrightarrow q$
Biconditional	$p \longleftrightarrow q$

Logical Connectives

Negation (not)	$\neg p$
Conjunction (and)	$p \wedge q$
Disjunction (or)	$p \vee q$
Exclusive Or	$p \oplus q$
Implication	$p \longrightarrow q$
Biconditional	$p \longleftrightarrow q$

p	"Garfield has black stripes"
q	"Garfield is an orange cat"
r	"Garfield likes lasagna"

"Garfield has black stripes if he is an orange cat and likes lasagna, and he is an orange cat or does not like lasagna"

$\Downarrow$

(p if (q and r)) and (q or (not r))

Logical Connectives

Negation (not)	$\neg p$
Conjunction (and)	$p \land q$
Disjunction (or)	$p \lor q$
Exclusive Or	$p \oplus q$
Implication	$p \longrightarrow q$
Biconditional	$p \longleftrightarrow q$

p	"Garfield has black stripes"
q	"Garfield is an orange cat"
r	"Garfield likes lasagna"

"Garfield has black stripes if he is an orange cat and likes lasagna, and he is an orange cat or does not like lasagna"

(p if (q and r)) and (q or (not r))

(p if (q ∧ r)) ∧ **(q** ∨ ¬**r)**

Some Truth Tables

p	$\neg p$

p	q	$p \wedge q$

p	q	$p \vee q$

p	q	$p \oplus q$

Some Truth Tables

p	$\neg p$
T	F
F	T

p	q	$p \wedge q$
T	T	T
T	F	F
F	T	F
F	F	F

p	q	$p \vee q$
T	T	T
T	F	T
F	T	T
F	F	F

p	q	$p \oplus q$
T	T	F
T	F	T
F	T	T
F	F	F

Implication

"If it's raining, then I have my umbrella"

p	q	$p \rightarrow q$
T	**T**	**T**
T	**F**	**F**
F	**T**	**T**
F	**F**	**T**

It's useful to think of implications as promises. That is "Did I lie?"

	It's raining	It's not raining
I have my umbrella		
I do not have my umbrella		

Implication

p	q	$p \to q$
T	T	T
T	F	F
F	T	T
F	F	T

"If it's raining, then I have my umbrella"

It's useful to think of implications as promises. That is "Did I lie?"

	It's raining	It's not raining
I have my umbrella	No	No
I do not have my umbrella	**Yes**	No

The only lie is when:

(a) It's raining AND

(b) I don't have my umbrella

Implication

"If it's raining, then I have my umbrella"

p	q	$p \rightarrow q$
T	T	T
T	F	F
F	T	T
F	F	T

Are these true?

$2 + 2 = 4 \rightarrow$ *earth is a planet*

$2 + 2 = 5 \rightarrow$ *26 is prime*

Implication

"If it's raining, then I have my umbrella"

p	q	$p \rightarrow q$
T	T	T
T	F	F
F	T	T
F	F	T

Are these true?

$2 + 2 = 4 \rightarrow$ *earth is a planet*

The fact that these are unrelated doesn't make the statement false! "2 + 2 = 4" is true; "earth is a planet" is true. T$\rightarrow$T is true. So, the statement is true.

$2 + 2 = 5 \rightarrow$ *26 is prime*

Again, these statements may or may not be related. "2 + 2 = 5" is false; so, the implication is true. (Whether 26 is prime or not is irrelevant).

Implication is not a causal relationship!

$p \rightarrow q$

(1) "I have collected all 151 Pokémon if I am a Pokémon master"

(2) "I have collected all 151 Pokémon only if I am a Pokémon master"

These sentences are implications in opposite directions:

$p \rightarrow q$

(1) "I have collected all 151 Pokémon if I am a Pokémon master"

(2) "I have collected all 151 Pokémon only if I am a Pokémon master"

These sentences are implications in opposite directions:

(1) "Pokémon masters have all 151 Pokémon"

(2) "People who have 151 Pokémon are Pokémon masters"

So, the implications are:

(1) If I am a Pokémon master, *then* I have collected all 151 Pokémon.

(2) If I have collected all 151 Pokémon, *then* I am a Pokémon master.

$p \rightarrow q$

p	q	$p \rightarrow q$
T	T	T
T	F	F
F	T	T
F	F	T

Implication:

- p implies q
- whenever p is true q must be true
- if p then q
- q if p
- p is sufficient for q
- p only if q
- *q is necessary for p*

Biconditional: $p \leftrightarrow q$

- *p* iff *q*
- *p* is equivalent to *q*
- *p* implies *q* and *q* implies *p*
- *p* is necessary and sufficient for *q*

p	q	$p \leftrightarrow q$

Biconditional: $p \leftrightarrow q$

- *p* iff *q*
- *p* is equivalent to *q*
- *p* implies *q* and *q* implies *p*
- *p* is necessary and sufficient for *q*

p	q	$p \leftrightarrow q$
T	T	T
T	F	F
F	T	F
F	F	T

Back to Garfield...

p	"Garfield has black stripes"
q	"Garfield is an orange cat"
r	"Garfield likes lasagna"

"Garfield has black stripes if he is an orange cat and likes lasagna, and he is an orange cat or does not like lasagna"

↓

(p if (q and r)) and (q or (not r))

↓

(p "if" $(q \wedge r)) \wedge ($**q** $\vee \neg$**r)**

Back to Garfield...

p	**“Garfield has black stripes”**
q	**“Garfield is an orange cat”**
r	**“Garfield likes lasagna”**

“Garfield has black stripes if he is an orange cat and likes lasagna, and he is an orange cat or does not like lasagna”

↓

(p if (q and r)) and (q or (not r))

↓

(**p** **“if”** (q ∧ r)) ∧ (**q** ∨ ¬**r**)

↓

((q ∧ r) ⟶ **p**) ∧ (**q** ∨ ¬**r**)

Analyzing the Garfield Sentence with a Truth Table

p	q	r	$\neg r$	$q \vee \neg r$	$q \wedge r$	$(q \wedge r) \rightarrow p$	$((q \wedge r) \rightarrow p) \wedge (q \vee \neg r)$
F	F	F					
F	F	T					
F	T	F					
F	T	T					
T	F	F					
T	F	T					
T	T	F					
T	T	T					

Analyzing the Garfield Sentence with a Truth Table

p	q	r	$\neg r$	$q \vee \neg r$	$q \wedge r$	$(q \wedge r) \rightarrow p$	$((q \wedge r) \rightarrow p) \wedge (q \vee \neg r)$
F	F	F	T	T	F	T	T
F	F	T	F	F	F	T	F
F	T	F	T	T	F	T	T
F	T	T	F	T	T	F	F
T	F	F	T	T	F	T	T
T	F	T	F	F	F	T	F
T	T	F	T	T	F	T	T
T	T	T	F	T	T	T	T