Instructions

Instructions:

For the remainder of class, you will work through a series of questions to help you understand LTL better. In each part, you will answer a few questions about a single LTL formula. In some parts we provide an additional explanation of the formula. You may use any other resources you feel appropriate during this class session, just make a note of what resources you use as we will ask you to document them at the end of the worksheet. This is not a test. You will be evaluated only on your thoughtful participation. We will provide answers to the questions at the end of class (available on Moodle).

Cheat Sheet:

Logical Operator	Meaning	Example
AND (&)	Both conditions must be true.	a & b *means* 'Both "a" AND "b" must be true.'
OR (|)	At least one condition must be true.	a | b *means* 'Either "a" OR "b" (or both) must be true.'
NOT (!)	The condition must be false.	!a *means* '"a" must be false' or 'NOT "a"'
IMPLIES (⇒)	If the first condition is true, then the second must also be true.	a ⇒ b *means* 'IF "a" is true, THEN "b" must be true'

Temporal Operator	Meaning	Example
Next (X)	A statement must be true in the next time step.	X d *means* 'In the next moment, "d" must be true.'
Globally (G)	A statement must always be true in every future step.	G b *means* '"b" must always hold.'
Eventually (F)	A statement must be true at some future point.	F a *means* 'At some point, "a" must hold.'

Until (U)	One statement must hold until another becomes true.	b U a *means* '"b" must hold until "a" eventually holds.'

Please enter your full name to continue:

Formula-A

For each of the questions on this page, consider the formula below.

G(!a)

Given a sequence where "a" is true at some future point, what happens to the formula?

- The formula is satisfied.
- The formula is violated.
- The satisfaction of the formula depends on values after "a" becomes true.

Does the following sequence of values for "a" satisfy the formula? Why or Why not?

a:T T T T F F F (false from now on) ..

For a sequence to **satisfy** the formula, what is required of "a"?

- "a" must always be false.
- "a" must be false at least once.
- "a" can be true at some point, but it must always be followed by false.
- "a" must eventually become false.

Give an example of a sequence of values for "a" that satisfies this formula.

Formula-B

For each of the questions on this page, consider the formula and explanation below.

F(!e)

Formula:

```
F(!e)
```

Breakdown:

`F(...)` : "Finally" (F) — at some point in the future

`!e` : "e does not hold"

Rebuilding meaning from the bottom up:

`!e` means "e does not hold."

`F(!e)` means "At some point in the system's future, e should not hold."

Refined phrasing:

"At least once during execution, e must be false."

Do you think the explanation of the formula above is correct (Yes/No)? Why or Why not?

Which of these sequences satisfies the formula?

- `e: T T T T T T T T T (true from now on) ..`
- `e: T T T F T F T F T (true from now on) ..`
- `e: F F F T F F F F F (false from now on) ..`
- Both option 2 and 3

Given a sequence where "e" is always true, what happens to the formula?

- The formula is satisfied.
- The formula is violated.

True or False: For a sequence to **satisfy** the formula, "e" must be **permanently false** after some point.

- False
- True

Formula-C

For each of the questions on this page, consider the formula and explanation below.

G(!p⇒Xs)

Formula:

```
G(!p ⇒ Xs)
```

Breakdown:

`G` : "Globally" — this must hold at all times.

`!p` : "p does not hold."

`X` : "Next" — in the immediate next step.

`s` : "s holds."

`⇒` : "Implication" — if the left side holds, then the right must hold.

Rebuilding meaning from the bottom up:

`!p` means "p does not hold."

`Xs` means "in the next step, s must hold."

`!p ⇒ Xs` means "If p does not hold, then in the next moment, s must hold."

`G(!p ⇒ Xs)` means "At all times, if p is false, then in the next moment s must hold."

Refined phrasing:

"Whenever p does not hold, s must hold in the next step."

Do you think the explanation of the formula above is correct (Yes/No)? Why or Why not?

Which of these sequences **satisfies** the formula?

- **p:** T T T F F F T T T F T T ..
  s: F F F T T T F F F T T T ..
- **p:** T T T F F F T F F T ..
  s: F F F T T T T T T T ..
- **p:** T T T F F F T T T F F T ..
  s: F F F T F T T F F F T F ..

Given a sequence where "p" is always false and "s" is always false, what happens to the formula?

- The formula is satisfied.
- The formula is violated.

True or False: For a sequence to **satisfy** the formula, "p" must eventually be set to false.

- False
- True

Formula-D

For each of the questions on this page, consider the formula below.

G(c⇒X G r)

In a sequence, if "c" is false now, what can be said about "r"?

- "r" must be true from now on.
- "r" must be true in the next step and can then become false later.
- The formula says nothing about "r".

Which of these sequences **violate** the formula?

- c: F F T F F T F F ..
  r: T F F T T T T T ..
- c: T T T F F F F F ..
  r: F F F T T T F T ..
- c: T F F T F F F F ..
  r: T T T F T T T T ..

- ○ The first and third options.
- ○ The second and third options.
- ○ All of the options.

True or False: In a sequence where "r" is always false, the formula is **violated**.

- ○ False
- ○ True

Please explain the meaning of the formula in your own words (1 sentence).

Formula-E

For each of the questions on this page, consider the formula below.

F(!p & !u)

Which of these sequences **violates** the formula?

- ○

```
p: T T T F T T T T ..
u: F F F T T T F T ..
```

- p: T T T F T F T F ..
  u: F F F T T T F T ..
- p: T T T F T T T T ..
  u: F F F F T T F T ..
- Both the first and second options.
- Both the first and third options.

True or False: Given a sequence where "p" is always false, the formula is **violated**.

- False
- True

Please explain the meaning of the formula in your own words (1 sentence).

Give an example of a sequence that **satisfies** this formula.

Formula-F

For each of the questions on this page, consider the formula and explanation below.

G(!q) | G(!m)

Formula:

```
G(!q) | G(!m)
```

Breakdown:

`G(!q)` : **"Always, q does not hold."**

`G(!m)` : **"Always, m does not hold."**

`|` : **"Or" — at least one of these conditions must be true.**

Rebuilding meaning from the bottom up:

`G(!q)` **means "q never holds."**

`G(!m)` **means "m never holds."**

`G(!q) | G(!m)` **means "Either q never holds, or m never holds (or both)."**

Refined phrasing:

"At least one of the following must be true: q is always false, or m is always false."

Do you think the explanation of the formula above is correct (Yes/No)? Why or Why not?

Which of these sequences **satisfies** the formula?

- **q:** T T T F T T T T ..
  m: F F F T T T F T ..
- **q:** T T T T T T T T ..
  m: F F F F F F F F ..
- **q:** T T T F T T T T ..
  m: T F F T T T F T ..
- Both the first and second options.
- Both the second and third options.
- All of the options.
- None of the options.

True or False: The formula is **violated**, if "q" is sometimes true in a sequence.

- ○ False
- ○ True

Please explain the meaning of the formula in your own words (1 sentence).

Formula-G

For each of the questions on this page, consider the formula and explanation below.

$$G(!a \Rightarrow F\ z)$$

Formula:

```
G(!a ⇒ F z)
```

Breakdown:

- `G(...)` : "Globally" (G) — this must hold at all times.
- `!a` : "a does not hold."
- `F(...)` : "Finally" (F) — at some point in the future.
- `z` : "z holds."
- `⇒` : "Implication" — if the left side holds, then the right must hold.

Rebuilding meaning from the bottom up:

- `!a` means "a does not hold."
- `F z` means "At some point in the future, z must hold."
- `!a ⇒ F z` means "If a is false, then z must eventually hold."
- `G(!a ⇒ F z)` means "At all times, if a is false, then z must eventually hold in the future."

Refined phrasing:

"Whenever a is false, z must eventually become true."

Do you think the explanation of the formula above is correct (Yes/No)? Why or Why not?

For a sequence to **satisfy** the formula, what is required of "z"?

- "z" must always be true.
- Whenever "a" is false, "z" must eventually be true.
- None of the above. Please explain:

True or False: The following sequence **satisfies** the formula.

a: T T T F T F T T ..

z: T T F F F F F T ..

- False
- True

Please explain the meaning of the formula in your own words (1 sentence).

Give an example of a sequence that satisfies this formula.

Formula-H

For each of the questions on this page, consider the formula below.

$$G(a \Rightarrow F(vUw))$$

Given a sequence where "a" is false throughout, what happens to the formula?

- The formula is satisfied.
- The formula is violated.
- The satisfaction of the formula depends on the value of "F(vUw)".

Given a sequence where "a" is true and "w" never occurs, what happens to the formula?

- The formula is satisfied.
- The formula is violated.
- The satisfaction of the formula depends on the value of "v" over time.

Please explain the meaning of the formula in your own words (1 sentence).

Give an example of a sequence that satisfies this formula.

Wrap-Up

Final Step - Feedback

In the final step of today's class, we ask for your feedback on this activity.

Did you have **any prior knowledge of, or experience with, LTL** before today's class? If so, in what context?

- ○ No
- ○ Yes

Did you **use any additional resources** to answer these questions (e.g., the internet, the presented slides, an AI assistant, a peer, etc.)? If so, please give details.

- ○ No
- ○ Yes

How did this topic relate to other topics in this course?

- ○ LTL was **more challenging** than other topics in CSC250
- ○ LTL was **similarly challenging** than other topics in CSC250
- ○ LTL was **less challenging** than other topics in CSC250

When comparing the questions that provided explanations with those that did not, **to what extent were the explanations helpful in understanding the LTL formula**?

- ○ Not at all helpful
- ○ Slightly helpful
- ○ Moderately helpful
- ○ Very helpful
- ○ Extremely helpful

Which aspect(s) made the questions **challenging** to answer? (Select all that apply.)

- ☐ Understanding the syntax of LTL
- ☐ Understanding the meaning of temporal operators (e.g., G, F, X, U)
- ☐ Interpreting the implications of formulae
- ☐ Applying the formula to sequences
- ☐ Other (please specify):
- ☐ None, I did not find the questions challenging

Which **temporal operator was the most challenging** to understand (e.g., X (Next), G (Globally), F (Eventually), or U (Until))? If they were all equally challenging, respond "ALL".

How would **you improve the explanations** (e.g., shorter, longer, more visual, more step-by-step breakdowns)?

How could you **use LTL** (or this kind of reasoning) in a computer science **project**?

Comments / Corrections / Suggestions:

Excluded

Formula:

```
F(!e)
```

Breakdown:

`F(...)` : "Finally" (F) — at some point in the future

`!e` : "e does not hold"

Rebuilding meaning from the bottom up:

`!e` means "e does not hold."

`F(!e)` means "At some point in the system's future, e should not hold."

Refined phrasing:

"At least once during execution, e must be false."

Formula:

```
G(!p ⇒ Xs)
```

Breakdown:

`G` : "Globally" — this must hold at all times.

`!p` : "p does not hold."

`X` : "Next" — in the immediate next step.

`s` : "s holds."

`⇒` : "Implication" — if the left side holds, then the right must hold.

Rebuilding meaning from the bottom up:

`!p` means "p does not hold."

`Xs` means "in the next step, s must hold."

`!p ⇒ Xs` means "If p does not hold, then in the next moment, s must hold."

`G(!p ⇒ Xs)` means "At all times, if p is false, then in the next moment s must hold."

Refined phrasing:

"Whenever p does not hold, s must hold in the next step."

Formula:

```
G(!q) | G(!m)
```

Breakdown:

`G(!q)` : "Always, q does not hold."

`G(!m)` : "Always, m does not hold."

`|` : "Or" — at least one of these conditions must be true.

Rebuilding meaning from the bottom up:

`G(!q)` means "q never holds."

`G(!m)` means "m never holds."

`G(!q) | G(!m)` means "Either q never holds, or m never holds (or both)."

Refined phrasing:

"At least one of the following must be true: q is always false, or m is always false."

Formula:

```
G(!a ⇒ F z)
```

Breakdown:

`G(...)` : "Globally" (G) — this must hold at all times.

`!a` : "a does not hold."

`F(...)` : "Finally" (F) — at some point in the future.

`z` : "z holds."

⇒ : "Implication" — if the left side holds, then the right must hold.

Rebuilding meaning from the bottom up:

`!a` means "a does not hold."

`F z` means "At some point in the future, z must hold."

`!a ⇒ F z` means "If a is false, then z must eventually hold."

`G(!a ⇒ F z)` means "At all times, if a is false, then z must eventually hold in the future."

Refined phrasing:

"Whenever a is false, z must eventually become true."