Assignment 1

Andrew Ungureanu
40283344

Tuesday, March 4th 2025
SOEN 331

Problem 1: Propositional Logic (7 pts)

1.1 (3 pts)

P: Humans are nice to Sofia
Q: Sofia is nice to humans $P \to Q$

- $P \to Q$ If humans are nice to Sofia, then Sofia is nice to humans.
- $\neg P$ Humans are not nice to Sofia.
- Therefore, $\neg Q$ Sofia is not nice to humans.

P	Q	$P \to Q$
T	T	T
T	F	F
F	T	T
F	F	T

As seen from this truth table, it is not implied that when humans are not nice, then Sofia will not be nice to humans. Indeed, there are two possibilities for it. Therefore, there is an inverse error and you can not determine Sofia is a threat to humans.

1.2 (4 pts)

P: Each man has a definite set of rules of conduct
Q: Man is no better than a machine

- $P \to Q$ If men have strict rules, then they are like machines.
- $\neg P$ There are no such rules.
- Therefore, $\neg Q$ Men cannot be machines.

This argument follows once again falls in the inverse error category, which is an invalid form of reasoning. In general, from $P \to Q$ and $\neg P$, we cannot logically conclude $\neg Q$.

Problem 2: Predicate logic (8 pts)

2.1 (4 pts)

(a) $\forall x \in \mathbb{R} : \text{number}(x) \rightarrow \neg\text{rational}(x)$
For all x, if x is a real number, then x is not rational.
Type: E

(b) $\exists x \in \mathbb{R} : \text{number}(x) \wedge \text{rational}(x)$
There exists a number that is real and rational.
Type: I

2.1 (4 pts)

(a) $\forall x \in \mathbb{R}, \text{number}(x) \rightarrow \text{rational}(x)$
Type: E

(b) $\exists x \in \mathbb{R}, \text{number}(x) \wedge \neg\text{rational}(x)$
Type: I

Problem 3: Linear Temporal Logic 1 (15 pts)

3.1 (3 pts)

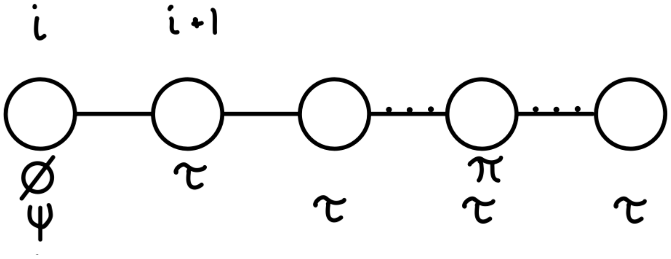
$$\bigcirc\Box(\phi \oplus \psi) \rightarrow \bigcirc^2\Diamond(\chi W \tau)$$

3.2 (3 pts)

$$(\phi \oplus \psi) \rightarrow \bigcirc\Box(\chi \rightarrow (\kappa R \omega))$$

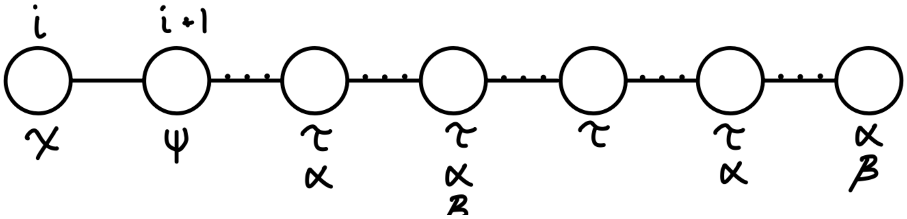
3.3 (3 pts)

$$\neg(\neg\varphi \vee \neg\psi) \rightarrow \Diamond\Box\tau \wedge \bigcirc^2\Diamond\pi$$



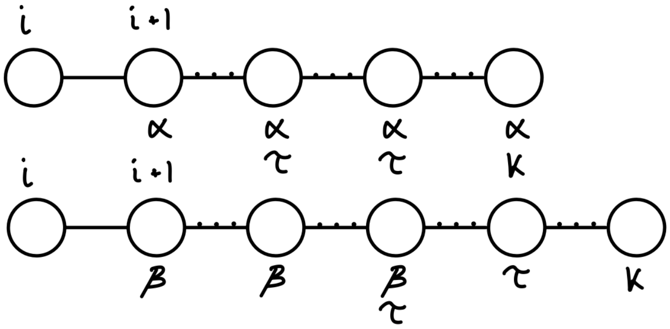
If φ and ψ are true, then τ will eventually become invariant and starting from time $i+2$, π will eventually happen.

3.4 (3 pts)



If starting at time $i+2$, τ eventually becomes invariant and at time i, x becomes true and $/psi$ becomes true at time $i+1$, then starting at time $i+2$, τ will be true at all states until and including when β becomes true, infinitely often.

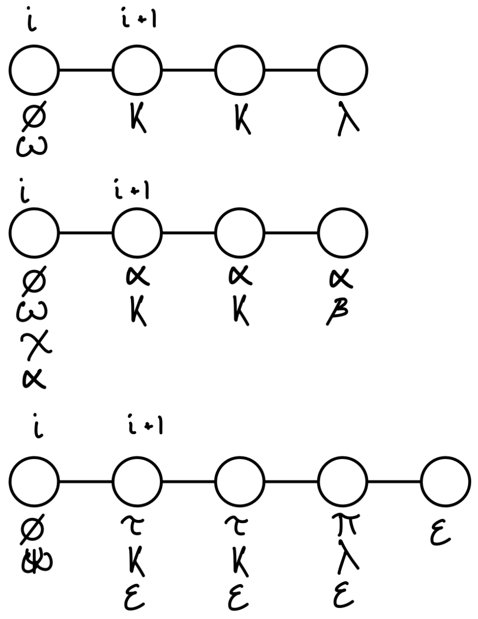
3.5 (3 pts)



If at time $i+1$, either one but not both of α and β are true, then starting at time $i+1$, τ will eventually be true and remain until κ becomes true. (κ will become true)

Problem 4: Linear Temporal Logic 2 (15 pts)

4.1 Visualizing all models of behavior (9 pts)



4.2 (4 pts)

$$\langle(\phi \wedge \omega), \kappa, \kappa, \lambda\rangle$$
$$\langle(\phi \wedge \chi \wedge \omega \wedge \alpha), (\alpha \wedge \kappa), (\alpha \wedge \kappa), (\alpha \wedge \beta \wedge \lambda)\rangle$$
$$\langle(\phi \wedge \psi \wedge \omega), (\tau \wedge \kappa \wedge \epsilon), (\tau \wedge \kappa \wedge \epsilon), (\pi \wedge \lambda \wedge \epsilon), \epsilon\rangle$$

4.3 (2 pts)

Model 1: The first model fully terminates. Additionally, it is consistent.
Model 2: The second model fully terminates. Additionally, it is consistent.
Model 3: The third model fully terminates. Additionally, it is consistent.

Problem 5: Unordered Structures (10 pts)

1. $Crew = \mathbb{P}\ Names$

2. *Spock* would not be a legitimate value, as all values of *Crew* must be a set of names. *Spock* is not (it does not have curly braces).

3. {Kirk} is a valid subset of *Names.* In $\mathbb{P}$ Names, it is the set of all subsets of S. {Kirk} is one of those subsets which makes this valid.

4. No, it is not. Writing *Crew:* $\mathbb{P}$ *Names* is interpreted as the variable *Crew* can assume any value supported by the Powerset of *Names.* However, $Crew = \mathbb{P}\ Names$ means that the variable Crew is the Powerset of *Names.*

5. No, the elements in *Names* aren't sets. Therefore, they can't be subsets of $\mathbb{P}$ *Names* if they are individual names.

6. $\{Names\} = \{\{Kirk,\ Spock,\ ...\}\}$ is not an element of Powerset of *Names.*

7. No, the elements of the product are actually ordered pairs.

8. Yes, because ever Powerset has an empty set.

9. The variable *Commander* can assume any value supported by the Powerset of *Crew.* For example, can be the empty set {}, one element of the said or the entire set.

10. They must be ordered pairs. Therefore, in this case it is false.

Problem 6: Ordered Structures (10 pts)

6.1 (4 pts)

To implement a Queue Abstract Data Type with element $\langle 1, 2, 3 \rangle$, I would utilize the Stack data type. However, these two different data types operate differently. Queues work with the priniciple of first-in and first-out. Stacks work by adding removing elements from the same end. Therefore, using two lists of stacks will help us modify this difference by utilizing the push and pop operations. In addition, this proposed model will allow us to have elements of generic type T.

6.2 (6 pts)

You define two lists L1 and L2 which are both stacks and have the proprieties of a stack
Enqueue: Add an element to the back of a line

- To add an element to the queue, simply push it onto L1:

$$L1.push(el)$$

Dequeue: Remove first element of a line

- To remove an element, we need to reverse the order:

  1. Move all elements from L1 to L2 using:

  $$L2.push(L1.pop()) \quad \text{(repeat until L1 is empty)}$$

  2. Remove the top element from L2:

  $$L2.pop()$$

  3. Move all elements back from L2 to L1:

  $$L1.push(L2.pop()) \quad \text{(repeat until L2 is empty)}$$

This ensures that elements are dequeued in the correct order while still using stack operations.

Problem 7: Binary Relations, Functions, and Orderings (15 pts)

7.1

*LACK OF TIME

7.2 (1 pts)

*LACK OF TIME

7.3 (12 pts)

- a) The variable *map* is a function. It is a partial function because not the element 2 does not have a corresping element in the codomain. The domain is $\{2, 4, 6, 24, 36, 72\}$, and the codomain is the power set of $\{1, 2, 3\}$.

- b)

  - **Injective:** No, because multiple domain elements map to the same set (36 and 6 $\{3\}$).
  - **Surjective (Onto):** No, because not every subset of $\{1, 2, 3\}$ is mapped.
  - **Bijective:** No, because it is not injective or surjective.
  - **Order Preserving:** No, because the order of elements in the domain is not preserved in the codomain.
  - **Order Reflecting:** No because order is not preserved as previously stated.
  - **Order Embedding:** No, as it does not preserve or reflect order.
  - **Isomorphism:** No, because it is not bijective and does not preserve structure.

Problem 8 Binary Relations, Functions and Orderings 2 (10 pts)

(a) The variable $map2$ is a function. It is a total function because every element in the domain has a corresponding element in the codomain. The domain is $\{\{1,2,3\},\{1,2\},\{1,3\},\{2,3\},\{1\},\{2\},\{3\}\}$, and the codomain is $\{2,3,4,6,24,36,72\}$.

(b)

- **Injective (One-to-One):** No, because different subsets map to the same value.
- **Surjective (Onto):** Yes, because every element in the codomain is mapped to by some subset in the domain.
- **Bijective:** No, because it is not injective
- **Order Preserving:** No, because the order of subsets in the domain is not preserved in the codomain.
- **Order Reflecting:** No, for similar reasons as order preserving.
- **Order Embedding:** No, as it does not preserve or reflect order.
- **Isomorphism:** No, because it is not bijective.

Problem 9: Construction Techniques (10 pts)

9.1 (5 pts)

- a)

$$\text{cons}(f(a), \text{map}(f, \langle b, c \rangle))$$

- b)

$$\text{map}(f, \Lambda) = \langle \rangle$$

$$\text{map}(f, \Lambda) = \text{cons}(f(\Lambda), \text{map}(f, (\Lambda))$$

- c)

$$\begin{aligned} &= \text{cons}(f(a), \text{map}(f, \langle b, c \rangle)) \\ &= \text{cons}(f(a), \text{cons}(f(b), \text{cons}(f(c), \langle \rangle))) \\ &= \langle f(a), f(b), f(c) \rangle \end{aligned}$$

9.2 (5 pts)

- a)

$$\text{cons}(a, \text{insert}(c, \langle b, c, d \rangle))$$

- b)

$$\text{insert}(c, \Lambda) = \langle c \rangle, \quad \text{if } \Lambda \text{ is empty *LACK OF TIME/NOT SURE}$$

- c) LACK OT TIME/NOT SURE