Formal Reasoning 2020
Solutions Exam
(14/01/21)

There are six sections, with each three multiple choice questions and one open question. Each multiple choice question is worth 3 points, and the open questions are worth 6 points. The first ten points are free. Good luck!

Propositional logic: multiple choice questions

1. We use the dictionary:

W	it is winter
S	it snows

What is the best formalization of the sentence:

It only snows in winter.

(a) $W \to S$

(b) is correct

(b) $S \to W$

(c) $S \leftrightarrow W$

(d) $W \land S$

Answer (b) is correct. If 'It only snows in winter', it means that if 'it snows' is true, it 'must be winter'.

The first answer makes no sense because this states that 'if it is winter, it must snow', which is not even close to the original sentence.

The third answer makes no sense because this, besides the proper part that 'if it snows, it must be winter', also states that 'if it is winter, it must snow'.

The fourth answer makes no sense because this states that 'it is winter and it snows', which is also completely different from the original sentence.

1. We use the dictionary:

W	it is winter
S	it snows

What is the best formalization of the sentence:

It snows because it is winter.

(a) $W \to S$

(b) $S \to W$

(c) $S \leftrightarrow W$

(d) is correct

(d) $W \land S$

Answer (d) is correct. The sentence 'It snows because it is winter' in particularly states that it is winter and that it snows.

The other answers make no sense because none of them state that it is actually snowing.

2. What is the syntax according to the official grammar from the course notes of the formula:

$$\neg a \to b$$

(b) is correct

(a) $\neg(a \to b)$
(b) $(\neg a \to b)$
(c) $(\neg(a \to b))$
(d) $((\neg a) \to b)$

Answer (b) is correct. The negation binds stronger than the implication, so we should read it as $((\neg a) \to b)$. However, the negation does not get any parentheses, so only the outer parentheses for the implication remain.

It is clear that none of the other options can be correct at the same time.

2. What is the syntax according to the official grammar from the course notes of the formula:

$$\neg a \vee \neg b$$

(b) is correct

(a) $((\neg a) \vee (\neg b))$
(b) $(\neg a \vee \neg b)$
(c) $\neg(a \vee \neg b)$
(d) $(\neg(a \vee \neg b))$

Answer (b) is correct. The negation binds stronger than the disjunction, so we should read it as $((\neg a) \vee (\neg b))$. However, the negations do not get any parentheses, so only the outer parentheses for the disjunction remain.

It is clear that none of the other options can be correct at the same time.

3. Does the following hold?

$$(a \to b) \to c \equiv a \to (b \to c)$$

(c) is correct

(a) Yes, because both formulas are true in the same models.
(b) No, because there is a model in which the left formula is true and the right formula is false (but not the other way around).
(c) No, because there is a model in which the right formula is true and the left formula is false (but not the other way around).
(d) No, because there is a model in which the left formula is true and the right formula is false, and there is a model in which the right formula is true and the left formula is false.

Answer (c) is correct. If we create a combined truth table for the formulas we get:

a	b	c	$a \to b$	$(a \to b) \to c$	$b \to c$	$a \to (b \to c)$
0	0	0	1	0	1	1
0	0	1	1	1	1	1
0	1	0	1	0	0	1
0	1	1	1	1	1	1
1	0	0	0	1	1	1
1	0	1	0	1	1	1
1	1	0	1	0	0	0
1	1	1	1	1	1	1

So we see that there are models for which the right formula is true and the left formula is false (for instance in the first row), but there are no models for which the left formula is true and the right formula is false.

It is clear that none of the other options can be correct at the same time.

3. Does the following hold?

$$(a \leftrightarrow b) \leftrightarrow c \equiv a \leftrightarrow (b \leftrightarrow c)$$

(a) is correct

(a) Yes, because both formulas are true in the same models.
(b) No, because there is a model in which the left formula is true and the right formula is false (but not the other way around).
(c) No, because there is a model in which the right formula is true and the left formula is false (but not the other way around).
(d) No, because there is a model in which the left formula is true and the right formula is false, and there is a model in which the right formula is true and the left formula is false.

Answer (a) is correct. If we create a combined truth table for the formulas we get:

a	b	c	$a \leftrightarrow b$	$(a \leftrightarrow b) \leftrightarrow c$	$b \leftrightarrow c$	$a \leftrightarrow (b \leftrightarrow c)$
0	0	0	1	0	1	0
0	0	1	1	1	0	1
0	1	0	0	1	0	1
0	1	1	0	0	1	0
1	0	0	0	1	1	1
1	0	1	0	0	0	0
1	1	0	1	0	0	0
1	1	1	1	1	1	1

So we see that the truth tables are the same, so in particular both formulas are true in the same models.

It is clear that none of the other options can be correct at the same time.

4. Consider the sentence:

$$f \equiv g \text{ iff } \models f \leftrightarrow g$$

(where 'iff' is an abbreviation of 'if and only if'). What in this sentence can occur in a formula of propositional logic?

(a) $\equiv$

(b) $\leftrightarrow$ and $\equiv$

(c) is correct (c) $\leftrightarrow$

(d) $\leftrightarrow$ and $\equiv$ and 'iff'

Answer (c) is correct. Besides atomic variables, formulas can only contain the symbols for the logical operators $\neg$, $\wedge$, $\vee$, $\rightarrow$, and $\leftrightarrow$. Symbols like $\equiv$ and $\vDash$ are used to state things about formulas, but are not part of the formulas. The same holds for constructions like 'iff' and 'if ... then'.

4. Consider the sentence:

$$\text{if } f \vDash g\text{, then } \vDash f \rightarrow g$$

What in this sentence can occur in a formula of propositional logic?

(a) $\vDash$

(b) $\rightarrow$ and $\vDash$

(c) is correct (c) $\rightarrow$

(d) $\rightarrow$ and $\vDash$ and 'if ... then'

Answer (c) is correct. Besides atomic variables, formulas can only contain the symbols for the logical operators $\neg$, $\wedge$, $\vee$, $\rightarrow$, and $\leftrightarrow$. Symbols like $\equiv$ and $\vDash$ are used to state things about formulas, but are not part of the formulas. The same holds for constructions like 'iff' and 'if ... then'.

Propositional logic: open question

5. We use the dictionary:

E	there is an epidemic
H	exams are at home

Give an English sentence without negations that clearly describes the meaning of the propositional formula:

$$\neg(\neg E \vee \neg H)$$

A literal translation would be: *It is not the case that there is not an epidemic or the exams are not at home.* However, using De Morgan, we get that the formula $\neg(\neg E \vee \neg H)$ is logically equivalent to the formula $E \wedge H$, which translates to the easier: *There is an epidemic and the exams are at home.*

5. We use the dictionary:

E	there is an epidemic
H	exams are at home

Give an English sentence without negations that clearly describes the meaning of the propositional formula:

$$\neg(H \to \neg E)$$

A literal translation would be: *It is not the case that if the exams are at home, then there is not an epidemic.* However, using rewrite rules, we get that the formula $\neg(H \to \neg E)$ is logically equivalent to the formula $\neg(\neg H \vee \neg E)$, which is logically equivalent to $H \wedge E$, which translates to the easier: *The exams are at home and there is an epidemic.*

Predicate logic: multiple choice questions

6. We use the dictionary:

M	domain of men
s	Sharon
$N(x)$	x is nice
$L(x, y)$	x loves y

Which of the following formulas corresponds to the sentence:

There is a nice man who loves Sharon.

(a) is correct

(a) $\exists x \in M[N(x) \wedge L(x, s)]$
(b) $\exists x \in M[N(x) \to L(x, s)]$
(c) $\forall x \in M[N(x) \wedge L(x, s)]$
(d) $\forall x \in M[N(x) \to L(x, s)]$

Answer (a) is correct.

The second answer makes no sense because this formula does not state that there actually exists a nice man.

The third answer makes no sense because this formula states that all men are nice and all men love Sharon.

The fourth answer makes no sense because this formula states that all nice men love Sharon.

6. We use the dictionary:

M	domain of men
s	Sharon
$N(x)$	x is nice
$L(x, y)$	x loves y

Which of the following formulas corresponds to the sentence:

All nice men love Sharon.

(a) $\exists x \in M[N(x) \wedge L(x, s)]$
(b) $\exists x \in M[N(x) \to L(x, s)]$
(c) $\forall x \in M[N(x) \wedge L(x, s)]$

(d) is correct

(d) $\forall x \in M[N(x) \to L(x, s)]$

Answer (d) is correct.

The first answer makes no sense because this formula states that there exists at least one man who loves Sharon, but it doesn't impose that all nice men love Sharon.

The second answer makes no sense because this formula states that there exists a man for which it holds that if he is nice, then he loves Sharon, but it doesn't say anything about all nice men.

The third answer makes no sense because this formula states that all men are nice and all men love Sharon.

7. Which of the following formulas does not express that there is exactly one element of D that has property $P(x)$?

   (a) $\exists x \in D\, \forall y \in D\, [P(y) \leftrightarrow y = x]$

   (b) $\exists x \in D\, [P(x) \wedge \forall y \in D\, [y \neq x \to \neg P(y)]]$

   (c) $\exists x \in D\, P(x) \wedge \forall x_1, x_2 \in D\, [P(x_1) \wedge P(x_2) \to x_1 = x_2]$

   (d) is correct

   (d) $\exists x \in D\, \forall y \in D\, [P(y) \to y = x]$

   Answer (d) is correct. This option allows for the possibility that there are no elements in D having property P.

7. Which of the following formulas does not express that there is at most one element of D that has property $P(x)$?

   (a) $\forall x_1, x_2 \in D\, [P(x_1) \wedge P(x_2) \to x_1 = x_2]$

   (b) $\forall x_1, x_2 \in D\, [x_1 \neq x_2 \to \neg P(x_1) \vee \neg P(x_2)]$

   (c) $\neg \exists x_1, x_2 \in D\, [x_1 \neq x_2 \wedge P(x_1) \wedge P(x_2)]$

   (d) is correct

   (d) $\neg \exists x_1, x_2 \in D\, [P(x_1) \wedge x_1 \neq x_2 \to \neg P(x_2)]$

   Answer (d) is correct. Because of the combination of the $\exists$ and the $\to$, it is hard to explain what this formally really means. It is easier to state why the other three are correct:

   - The first one states that if there are two elements having property P, then they must be the same.
   - The second one states that if there are two different elements, then they cannot both have property P.
   - The third one states that there are no two different elements that both have property P.

   So in all cases, there cannot be two different ones, but it is possible that there exists no element with property P. Hence there is at most one element having property P.

8. Consider the following two logical equivalences that express 'logical laws':

   - $\neg \forall x \in D\, f \equiv \exists x \in D\, \neg f$
   - $\neg(f \leftrightarrow g) \equiv \neg f \leftrightarrow \neg g$

Which of these hold?

(b) is correct

(a) both
(b) only the first
(c) only the second
(d) none

Answer (b) is correct. The first one holds because it is the law of De Morgan for predicate logic.

To show that the second one doesn't hold, we reason as follows. The $f \leftrightarrow g$ leads to true exactly when f and g have the same truth values. However, if f and g have the same truth values, then $\neg f$ and $\neg g$ also have the same truth values. So $f \leftrightarrow g \equiv \neg f \leftrightarrow \neg g$. Hence if we negate only the left formula, it is clear that $\neg(f \leftrightarrow g) \not\equiv \neg f \leftrightarrow \neg g$.

Of course, it is also possible to see this in a table with all four possibilities for the truth values of the arbitrary formulas f and g. So although this looks like a truth table, technically it is not.

f	g	$f \leftrightarrow g$	$\neg(f \leftrightarrow g)$	$\neg f$	$\neg g$	$\neg f \leftrightarrow \neg g$
0	0	1	0	1	1	1
0	1	0	1	1	0	0
1	0	0	1	0	1	0
1	1	1	0	0	0	1

In the table we see that in all situations the truth values of both formulas are the opposite, so the formulas are not equivalent.

8. Consider the following two logical equivalences that express 'logical laws':

- $\neg\exists x \in D\, f \equiv \forall x \in D\, \neg f$
- $\neg(f \to g) \equiv \neg g \to \neg f$

Which of these hold?

(b) is correct

(a) both
(b) only the first
(c) only the second
(d) none

The first one holds because it is the law of De Morgan for predicate logic.

To show that the second one doesn't hold, we reason as follows. Any implication is true in three out of the four possibilities. So this holds for $f \to g$ and $\neg g \to \neg f$. However, the negation of a formula that is true in three out of the four possibilities, is only true in one of the four possibilities. So on the left we have a formula that is only true once, whereas the formula on the right is true three times. So they cannot be equivalent.

Of course, it is also possible to see this in a table with all four possibilities for the truth values of the arbitrary formulas f and g. So although this looks like a truth table, technically it is not.

f	g	$f \to g$	$\neg(f \to g)$	$\neg g$	$\neg f$	$\neg g \to \neg f$
0	0	1	0	1	1	1
0	1	1	0	0	1	1
1	0	0	1	1	0	0
1	1	1	0	0	0	1

In the table we see that in all situations the truth values of both formulas are the opposite, so the formulas are not equivalent.

9. Consider the model (men, women, parent_of), and an interpretation in this model:

M	domain of men
W	domain of women
$P(x, y)$	y is a parent of x

You may assume that both parents of all people in $(M \cup W)$ are also in $(M \cup W)$.

Which of the following formulas is not true in this model?

(a) is correct

(a) $\forall x \in (M \cup W)\, \exists y_1 \in M\, \exists y_2 \in W\, [P(y_1, x) \wedge P(y_2, x)]$
(b) $\forall x \in W\, \exists x', x'' \in W\, [P(x, x') \wedge P(x, x'')]$
(c) $\forall x \in W\, \exists x', x'' \in W\, [P(x, x') \wedge P(x', x'')]$
(d) $\forall x \in (M \cup W)\, \exists y \in (M \cup W)\, [P(x, y) \vee P(y, x)]$

Answer (a) is correct. This states that every person is parent of (at least) one man and one woman, which is certainly not true in this model.

The second formula seems to state that every woman has two mothers, however, these 'two' mothers may be the same. So it actually states that every woman has at least one mother.

The third formula states that every woman has a mother and a grandmother.

The fourth formula states that for all persons there is a person who is either his or her parent or his or her child. The first part of this disjunction holds. The second part does not.

9. Consider the model (men, women, parent_of), and an interpretation in this model:

M	domain of men
W	domain of women
$P(x, y)$	y is a parent of x

You may assume that both parents of all people in $(M \cup W)$ are also in $(M \cup W)$.

Which of the following formulas is not true in this model?

(a) $\forall x \in (M \cup W)\, \exists y_1 \in M\, \exists y_2 \in W\, [P(x, y_1) \wedge P(x, y_2)]$

(b) is correct

(b) $\forall x \in M\, \exists x', x'' \in M\, [P(x, x') \wedge P(x'', x)]$

(c) $\forall x \in M\, \exists x', x'' \in M\, [P(x, x') \wedge P(x'', x')]$

(d) $\forall x \in (M \cup W)\, \exists y \in (M \cup W)\, [P(x, y) \leftrightarrow P(y, x)]$

Answer (b) is correct. This states that every man has both a father and a son, which is certainly not true in this model.

The first formula states that every person has at least one parent.

The third formula states that every man has a parent, and this parent has a son.

The fourth formula states that for every person there exists a person for which it holds that the second person is a parent of the first person if and only if the first person is a parent of the second person. This may seem incorrect, but for the second person, we can take the first person again. And then we have that both sides of the $\leftrightarrow$ are false, so the $\leftrightarrow$ itself will be true.

Predicate logic: open question

10. Show that:

$$\not\models \forall x, y \in D\, [R(x, y) \vee x = y \vee R(y, x)]$$

Explain your answer.

Consider a model with as domain the integers and as relation equality. Take as interpretation

D	domain of integers $\mathbb{Z}$
$R(x, y)$	$x^2 = y$

Then if we take $x = 3$, $y = 7$, then $x, y \in \mathbb{Z}$, however, $3^2 = 7$ does not hold, $3 = 7$ does not hold, and $7^2 = 3$ does not hold. So the formula does not hold for all models and all interpretations.

10. Show that:

$$\not\models [\forall x \in D\, R(x, x)] \vee [\forall x \in D\, \neg R(x, x)]$$

Explain your answer.

Consider a model with as domain the integers and as relation equality. Take as interpretation

D	domain of integers $\mathbb{Z}$
$R(x, y)$	$x^2 = y$

Then if we take $x = 3$, then $x \in \mathbb{Z}$, however, $3^2 = 3$ does not hold, so $\forall x \in D\, R(x, x)$ does not hold. And if we take $x = 0$, then $x \in \mathbb{Z}$, and $0^2 = 0$ does hold, so $\forall x \in D\, \neg R(x, x)$ does not hold. Hence $[\forall x \in D\, R(x, x)] \vee [\forall x \in D\, \neg R(x, x)]$ does not hold. So the formula does not hold for all models and all interpretations.

Languages: multiple choice questions

11. Let be given a language L. Which of the following languages is not necessarily equal to the others?

(a) is correct

(a) LL^*
(b) $LL^* \cup \{\lambda\}$
(c) L^*L^*
(d) L^*

Answer (a) is correct. The idea is that any positive number of copies of a language L will not make a difference with L^*. However, L^* allows for zero copies and then you automatically get λ, which you may not get when doing at least one copy.

A bit more formal. Let $w \in L^*$. Then there exist $k \in \mathbb{N}$ such that $w = w_1 \dots w_k$ where $w_i \in L$ for $i \in \{1, \dots, k\}$. However, $w = w\lambda$, which is in L^*L^* because $w \in L^*$ and $\lambda \in L^*$. Hence $L^* \subseteq L^*L^*$.

We continue by proving that $L^*L^* \subseteq LL^* \cup \{\lambda\}$. Let $w \in L^*L^*$. Then there exist $k, l \in \mathbb{N}$ such that $w = w_1 \dots w_k w'_1 \dots w'_l$ where $w_i, w'_j \in L$ for $i \in \{1, \dots, k\}$ and $j \in \{1, \dots, l\}$. Now we make a case distinction on $k+l$. If $k + l = 0$, then $w = \lambda$ and clearly $w \in LL^* \cup \{\lambda\}$. If $k = 0$ and $l \geq 1$, then $w \in LL^*$, because $w = w'_1 w'_2 \dots w'_l$, and $w'_1 \in L$ and $w'_2 \dots w'_l \in L^*$. Note that the last part may be λ if $l = 1$. If $k \geq 1$, then $w \in LL^*$, because $w = w_1 w_2 \dots w_k w'_1 \dots w'_l$, and $w_1 \in L$ and $w_2 \dots w_k w'_1 \dots w'_l \in L^*$. Note that the last part may be λ if $k = 1$ and $l = 0$. So we also have that $L^*L^* \subseteq LL^* \cup \{\lambda\}$.

Finally, we show that $LL^* \cup \{\lambda\} \subseteq L^*$. Let $w \in LL^* \cup \{\lambda\}$. If $w = \lambda$, then immediately $w \in L^*$, because $\lambda \in L^*$ regardless of the language L. If $w \neq \lambda$ then there exists $k \in \mathbb{N}$ such that $w = w_1 w'_1 \dots w'_k$ where $w_1 \in L$ and $w'_i \in L$ for $i \in \{1, \dots, k\}$. But then there exists $l \in \mathbb{N}$ such that $w = w''_1 w''_2 \dots w''_l$ such that $w''_i \in L$ for $i \in \{1, \dots, l\}$. Namely, take $l = k + 1$, $w''_1 := w_1$, $w''_i := w'_{i-1}$ for $i \in \{2, \dots, k + 1\}$. So $w \in L^*$. And hence $LL^* \cup \{\lambda\} \subseteq L^*$.

So we get

$$L^* \subseteq L^*L^* \subseteq LL^* \cup \{\lambda\} \subseteq L^*$$

But this implies that these three languages are all necessarily equal, independent of the definition of L.

Note that if $\lambda \notin L$, then LL^* does not include λ and therefore LL^* is not necessarily equal to the other three languages.

11. Let be given a language L. Which of the following languages is not necessarily equal to the others?

(a) L
(b) $(L^R)^R$
(c) $L \cup (L^R)^R$

(d) is correct

(d) $((L^R)^R)^R$

Answer (d) is correct. First note that $(L^R)^R = L$, because reversing all words twice, we get back all original words.

A bit more formal. Let $w \in L$. Then $w^R \in L^R$ and $(w^R)^R \in (L^R)^R$. So $L \subseteq (L^R)^R$.

Now let $w \in (L^R)^R$. Then there exists a word $v \in L^R$ such that $w = v^R$. However, if $v \in L^R$ then there exists a word $u \in L$ such that $v = u^R$. So we get that $w = v^R = (u^R)^R = u$. So $w \in L$. So $(L^R)^R \subseteq L$.

And from

$$L \subseteq (L^R)^R \subseteq L$$

it indeed follows that $L = (L^R)^R$. And because $L = L \cup L$, it also follows that $L = L \cup L = L \cup (L^R)^R$.

Hence the three languages L, $(L^R)^R)$, and $L \cup (L^R)^R$ are necessarily equal, independent of the definition of L.

Now if $L \neq L^R$ we have that $((L^R)^R)^R = L^R$ and by assumption, L^R is not equal to L.

If $L = L^R$ then all languages are equal.

12. Which of the following regular expressions describes a language different from the others?

(a) $(a \cup b)^*$

(b) $(a^*b^*)^*$

(c) $(b^*a^*)^*$

(d) is correct (d) $(a^* \cup b^*)$

Answer (d) is correct. Note that

$$\mathcal{L}((a \cup b)^*) = \mathcal{L}((a^*b^*)^*) = \mathcal{L}((b^*a^*)^*) = \{a, b\}^*$$

and that

$$\mathcal{L}(a^* \cup b^*) = \{a\}^* \cup \{b\}^*$$

So in particular $ab \notin \{a\}^* \cup \{b\}^*$, but $ab \in \{a, b\}^*$.

12. Which of the following regular expressions describes a language different from the others?

(a) a^*b^*

(b) is correct (b) $(\lambda \cup aa)^*(\lambda \cup bb)^*$

(c) $(a \cup aa)^*(b \cup bb)^*$

(d) $(\lambda \cup a)^*(\lambda \cup b)^*$

Answer (b) is correct. Note that

$$\mathcal{L}(a^*b^*) = \mathcal{L}((a \cup aa)^*(b \cup bb)^*) = \mathcal{L}((\lambda \cup a)^*(\lambda \cup b)^*) = \{a^n b^m \mid n, m \in \mathbb{N}\}$$

and that

$$\mathcal{L}((\lambda \cup aa)^*(\lambda \cup bb)^*) = \left\{a^{2n}b^{2m} \mid n, m \in \mathbb{N}\right\}$$

So in particular $ab \notin \left\{a^{2n}b^{2m} \mid n, m \in \mathbb{N}\right\}$, but $ab \in \{a^n b^m \mid n, m \in \mathbb{N}\}$.

13. Consider the following context-free grammar for a fragment of English:

$$\begin{aligned} S &\to N \textit{ walks} \mid N \textit{ loves } N \mid S \textit{ and } S \\ N &\to \textit{the } A\, M \\ A &\to \textit{tall} \mid \textit{small} \mid \lambda \\ M &\to \textit{man} \mid \textit{woman} \end{aligned}$$

Which of the following statements about the language produced by this grammar is not true?

(a) The shortest sentences in the language have three words.

(b) The sentences in this language can be arbitrarily long.

(c) is correct

(c) This language contains an infinite sentence.

(d) For each $n \geq 3$ this language contains a sentence of exactly n words.

Answer (c) is correct. The sentences of this language, are the 'words' in a general language. And words in languages are never infinite. So the claim that the language contains infinite words is incorrect for any language.

The shortest sentences in the language are *the man walks* and *the woman walks*, and indeed they consist of three words. Note that any M adds exactly one word. Note that any A adds zero or one word, so if we are looking for the shortest sentences, A needs to be replaced by λ. Note that any N adds at least two words: 'the', zero words for A, and one word for M. This means that S adds at least three words: two words for N and the word 'walks'.

In order to show that sentences can be arbitrarily long, we show that for each $n \geq 3$ the language contains sentences of exactly n words.

- $n = 3$: *the man walks*
- $n = 4$: *the small woman walks*
- $n = 5$: *the man loves the woman*
- $n = 6$: *the small woman loves the man*

Note that any sentence can be extended with four words by adding *and the woman walks*. So this means that the sentences above can be extended to sentences of length $n \in \{7, 8, 9, 10\}$. And then, those newly created sentences can be extended to sentences of length $n \in 11, 12, 13, 14$. And so on.

Note that 'arbitrarily long' does not mean that any length is possible, but that if someone picks a natural number n, it will be possible to create a sentence that has a length that is at least n.

13. Consider the following context-free grammar for a fragment of a programming language, with alphabet $\Sigma = \{\texttt{:}, \texttt{=}, \texttt{;}, \texttt{0}, \texttt{1}, \texttt{+}, \texttt{*}, \texttt{(}, \texttt{)}, \texttt{x}, \texttt{y}\}$:

$$\begin{aligned} S &\to V \texttt{:=} E \mid S \texttt{;} S \\ E &\to \texttt{0} \mid \texttt{1} \mid V \mid E \texttt{+} E \mid E \texttt{*} E \mid \texttt{(} E \texttt{)} \\ V &\to \texttt{x} \mid \texttt{y} \end{aligned}$$

Which of the following statements about the language produced by this grammar is not true?

(a) The shortest programs in this language have four symbols.
(b) The programs in this language can be arbitrarily long.

(c) is correct

(c) This language contains an infinite program.
(d) This language does not contain a program of five symbols.

Answer (c) is correct. The programs of this language, are the 'words' in a general language. And words in languages are never infinite. So the claim that the language contains infinite words is incorrect for any language.

The shortest programs are of the type `x := 0`, so these programs indeed have four symbols. Note that any V adds exactly one symbol. Note that any E adds at least one symbol. This means that any S adds at least four symbols: one for V, two for `:=`, and one for E.

Note that 'arbitrarily long' does not mean that any length is possible, but that if someone picks a natural number n, it will be possible to create a sentence that has a length that is at least n.

There is an algorithm for creating a program of at least n symbols. This is due to the fact that any correct program can be extended with five symbols: '`; x := 0`'. First compute $q = \lfloor \frac{n}{5} \rfloor$ and then take the program `x := 0` followed by q times '`; x := 0`'. This will provide a valid program which length is at least n.

We have already seen that the shortest programs have length four. Obviously, if we want to use the production $S \to S\texttt{;}\ S$, we will get at least nine symbols. So we have to use the production $S \to V\texttt{:=}E$. As we have seen V will always give one symbol. So if we want to have five symbols in total, we have to find a way to expand E into two symbols. However, every E can be replaced by either one symbol, or by at least three. So there is no way we can create a program of five symbols.

14. Which of the following requirements does not necessarily hold for an invariant of a context-free grammar G?

(a) An invariant is a predicate on words from $(\Sigma \cup V)^*$.
(b) An invariant holds for the word $S \in (\Sigma \cup V)^*$.
(c) If an invariant holds for a word, and one symbol in the word is replaced according to a rule from the grammar, the invariant still holds.
(d) The invariant does not hold for the word of which we want to show that it is not in the language $\mathcal{L}(G)$.

(d) is correct

Answer (d) is correct. If we want to prove that some word is not in the language generated by a grammar, we do want to have that the invariant doesn't hold for this specific word. However, this is not part of the definition of an invariant.

The other three options are consequences of the definition of an invariant.

14. Which of the following requirements does not necessarily hold for an invariant of a context-free grammar G?

(a) The invariant holds for all words in $\mathcal{L}(G)$.
(b) The invariant holds for all words $S, w_1, w_2, \ldots$ in any production $S \to w_1 \to w_2 \to \ldots$ of the language.

(c) There is a word in $(\Sigma \cup V)^*$ for which the invariant holds.

(d) is correct (d) $\mathcal{L}(G)$ consists of the words in Σ^* that satisfy the invariant.

Answer (d) is correct. The most trivial example of an invariant is $P(w) :=$ true. It is clear that this is indeed an invariant. However, this invariant also holds for the words in Σ^* which are not in $\mathcal{L}(G)$.

The other three options are consequences of the definition of an invariant.

Languages: open question

15. Give a right linear grammar that produces the same language as the context-free grammar:

$$\begin{aligned} S &\to AB \\ A &\to aA \mid \lambda \\ B &\to bB \mid \lambda \end{aligned}$$

Note that the language generated by the given grammar is $\{a^n b^m \mid n, m \in \mathbb{N}\}$. This language can be generated with the following right linear grammar:

$$\begin{aligned} S &\to A \\ A &\to aA \mid B \\ B &\to bB \mid \lambda \end{aligned}$$

First, we add all the a's, then all the b's.

15. Give a right linear grammar that produces the same language as the context-free grammar:

$$\begin{aligned} S &\to AA \mid \lambda \\ A &\to aaA \mid S \end{aligned}$$

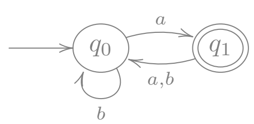
Note that the language generated by the given grammar is $\left\{a^{2n} \mid n \in \mathbb{N}\right\}$. This language can be generated with the following right linear grammar:

$$S \to aaS \mid \lambda$$

In every step, we add two a's.

Automata: multiple choice questions

16. Consider the following DFA:



How many words of four letters does this automaton accept?

(a) three

(b) four

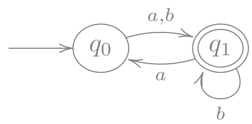
(c) less than three

(d) is correct (d) more than four

Answer (d) is correct. These are all options:

1. $q_0 \xrightarrow{b} q_0 \xrightarrow{b} q_0 \xrightarrow{b} q_0 \xrightarrow{a} q_1$
2. $q_0 \xrightarrow{b} q_0 \xrightarrow{a} q_1 \xrightarrow{a} q_0 \xrightarrow{a} q_1$
3. $q_0 \xrightarrow{b} q_0 \xrightarrow{a} q_1 \xrightarrow{b} q_0 \xrightarrow{a} q_1$
4. $q_0 \xrightarrow{a} q_1 \xrightarrow{a} q_0 \xrightarrow{b} q_0 \xrightarrow{a} q_1$
5. $q_0 \xrightarrow{a} q_1 \xrightarrow{b} q_0 \xrightarrow{b} q_0 \xrightarrow{a} q_1$

16. Consider the following DFA:



How many words of four letters does this automaton accept?

(a) three

(b) four

(c) less than three

(d) is correct (d) more than four

Answer (d) is correct. These are all options:

1. $q_0 \xrightarrow{a} q_1 \xrightarrow{b} q_1 \xrightarrow{b} q_1 \xrightarrow{b} q_1$
2. $q_0 \xrightarrow{b} q_1 \xrightarrow{b} q_1 \xrightarrow{b} q_1 \xrightarrow{b} q_1$
3. $q_0 \xrightarrow{a} q_1 \xrightarrow{a} q_0 \xrightarrow{a} q_1 \xrightarrow{b} q_1$
4. $q_0 \xrightarrow{b} q_1 \xrightarrow{a} q_0 \xrightarrow{a} q_1 \xrightarrow{b} q_1$
5. $q_0 \xrightarrow{a} q_1 \xrightarrow{a} q_0 \xrightarrow{b} q_1 \xrightarrow{b} q_1$
6. $q_0 \xrightarrow{b} q_1 \xrightarrow{a} q_0 \xrightarrow{b} q_1 \xrightarrow{b} q_1$
7. $q_0 \xrightarrow{a} q_1 \xrightarrow{b} q_1 \xrightarrow{a} q_0 \xrightarrow{a} q_1$
8. $q_0 \xrightarrow{b} q_1 \xrightarrow{b} q_1 \xrightarrow{a} q_0 \xrightarrow{a} q_1$
9. $q_0 \xrightarrow{a} q_1 \xrightarrow{b} q_1 \xrightarrow{a} q_0 \xrightarrow{b} q_1$
10. $q_0 \xrightarrow{b} q_1 \xrightarrow{b} q_1 \xrightarrow{a} q_0 \xrightarrow{b} q_1$

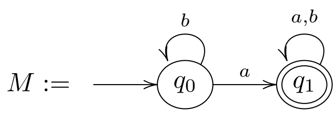
17. What is the minimum number of states for a DFA that accepts the language

$$\mathcal{L}\big((a \cup b)^* a (a \cup b)^*\big)$$

(a) is correct (a) two

(b) three
(c) less than two
(d) more than three

The language accepted here is the language $\{w \in \{a,b\}^* \mid w \text{ contains at least one } a\}$. A DFA that accepts this language is:



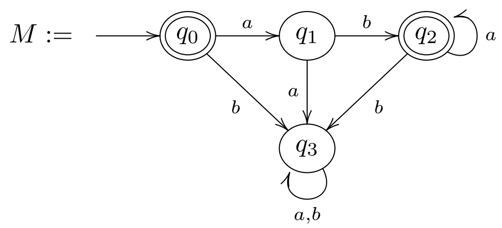
So it can be done with two states. And because λ is not accepted, it cannot be done with just one state.

17. What is the minimum number of states for a DFA that accepts the language

$$\mathcal{L}(\lambda \cup aba^*)$$

(a) two
(b) three
(c) less than two
(d) is correct (d) more than three

The language accepted here is the language $\{\lambda\} \cup \{aba^n \mid n \in \mathbb{N}\}$. A DFA that accepts this language is:



Note that because λ is accepted q_0 needs to be a final state. Note that from q_0 we need an outgoing arrow for a to q_1 because if we would send it back to q_0 then a would be accepted. And because words cannot start with a b, we need an outgoing arrow for b to a sink q_3. In q_1 we need an outgoing arrow with a b to a final state, but this cannot be q_0, because that would imply that $abab$ would be accepted. So we need a new final state q_2. In q_1 we need an arrow for a to the sink q_3 because it is not allowed to have two a's following each other before the obligatory b. In q_3 we need a loop for a back to q_2, because any number of a's here is fine. In q_3 we need an arrow for b to the sink q_3 because we cannot have more than one b in the word. So we see that the automaton given above is minimal and hence we need at least four states.

18. Consider the NFA:

$$M := \langle \Sigma, Q, q_0, F, \delta \rangle$$

with:

$$
\begin{aligned}
\Sigma &= \{a, b\} \\
Q &= \{q_0, q_1\} \\
F &= \{q_0\}
\end{aligned}
$$

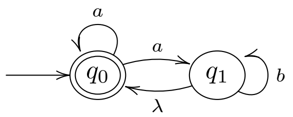
$$
\begin{aligned}
\delta(q_0, a) &= \{q_0, q_1\} \\
\delta(q_0, b) &= \varnothing \\
\delta(q_0, \lambda) &= \varnothing \\
\delta(q_1, a) &= \varnothing \\
\delta(q_1, b) &= \{q_1\} \\
\delta(q_1, \lambda) &= \{q_0\}
\end{aligned}
$$

What is true?

(b) is correct

(a) $aab \in L(M)$ and $baa \in L(M)$
(b) $aab \in L(M)$ and $baa \notin L(M)$
(c) $aab \notin L(M)$ and $baa \in L(M)$
(d) $aab \notin L(M)$ and $baa \notin L(M)$

Answer (b) is correct. This is the corresponding automaton represented graphically:

$M :=$ 

We have a production for aab: $q_0 \xrightarrow{a} q_0 \xrightarrow{a} q_1 \xrightarrow{b} q_1 \xrightarrow{\lambda} q_0$. However, since $\delta(q_0, b) = \varnothing$ and $\delta(q_0, \lambda) = \varnothing$, the automaton doesn't accept any words starting with b, so baa is not accepted.

18. Consider the NFA:

$$M := \langle \Sigma, Q, q_0, F, \delta \rangle$$

with:

$$
\begin{aligned}
\Sigma &= \{a, b\} \\
Q &= \{q_0, q_1\} \\
F &= \{q_0\}
\end{aligned}
$$

$$
\begin{aligned}
\delta(q_0, a) &= \{q_0, q_1\} \\
\delta(q_0, b) &= \varnothing \\
\delta(q_0, \lambda) &= \varnothing \\
\delta(q_1, a) &= \varnothing \\
\delta(q_1, b) &= \{q_1\} \\
\delta(q_1, \lambda) &= \{q_0\}
\end{aligned}
$$

What is true?

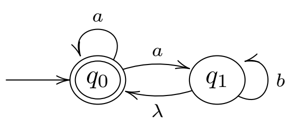
(a) $aba \in L(M)$ and $bab \in L(M)$

(b) is correct (b) $aba \in L(M)$ and $bab \notin L(M)$

(c) $aba \notin L(M)$ and $bab \in L(M)$

(d) $aba \notin L(M)$ and $bab \notin L(M)$

Answer (b) is correct. This is the corresponding automaton represented graphically:

$M :=$ 

We have a production for *aba*: $q_0 \xrightarrow{a} q_1 \xrightarrow{b} q_1 \xrightarrow{\lambda} q_0 \xrightarrow{a} q_1 \xrightarrow{\lambda} q_0$. However, since $\delta(q_0, b) = \varnothing$ and $\delta(q_0, \lambda) = \varnothing$, the automaton doesn't accept any words starting with b, so *bab* is not accepted.

19. There exist DFAs with 2020 states that accept the word $a^{2021}b^{2021}$. Does such a DFA always accept a word $a^n b^{2021}$ as well, for some $n > 2021$?

(a) Yes, because while processing the a's in $a^{2021}b^{2021}$, there has to be a state that occurs twice, which means there is a loop while processing

(a) is correct the a's.

(b) No, because it only accepts words of the form $a^n b^n$.

(c) No, because the language $\{a^n b^n \mid n \in \mathbb{N}\}$ is not regular, so it is not accepted by a DFA.

(d) You cannot know this, this is the case for some of these automata, but not for all.

Answer (a) is correct. This is basically the proof of the *pumping lemma for regular languages*, which is not really discussed in this course, but the explanation given above should be convincing.

The second answer makes no sense because we don't know anything about the words being accepted, besides that $a^{2021}b^{2021}$ is accepted.

The third answer makes no sense because this claim is not about this language.

The fourth answer makes no sense because we do know this: the first answer is correct.

19. There exist DFAs with 2020 states that accept the word $a^{2021}b^{2021}$. Does such a DFA always accept a word $a^{2021}b^n$ as well, for some $n > 2021$?

(a) Yes, because while processing the b's in $a^{2021}b^{2021}$, there has to be a state that occurs twice, which means there is a loop while processing

(a) is correct the b's.

(b) No, because it only accepts words of the form $a^n b^n$.

(c) No, because the language $\{a^n b^n \mid n \in \mathbb{N}\}$ is not regular, so it is not accepted by a DFA.

(d) You cannot know this, this is the case for some of these automata, but not for all.

Answer (a) is correct. This is basically the proof of the *pumping lemma for regular languages*, which is not really discussed in this course, but the explanation given above should be convincing.

The second answer makes no sense because we don't know anything about the words being accepted, besides that $a^{2021}b^{2021}$ is accepted.

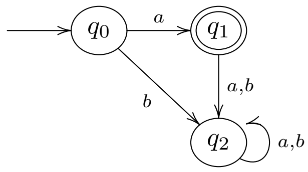
The third answer makes no sense because this claim is not about this language.

The fourth answer makes no sense because we do know this: the first answer is correct.

Automata: open question

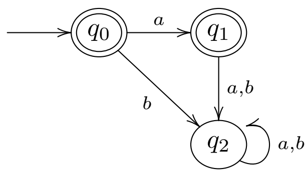
20. Give a regular language with alphabet $\Sigma = \{a, b\}$ that does not contain the empty word, and for which there does not exist a DFA with at most two states.

    Take the language $\mathcal{L}(a)$. It contains only the word a and in particular not the word λ. Because λ is not in the language, we know that the starting state q_0 is not a final state. However, since a is accepted, there has to be a final state, so we need at least a q_1 that is a final state. In addition, because all words with b are not accepted, we need a sink q_2. So we need at least three states. In fact, it can be done with three states:

$M :=$ 

20. Give a regular language with alphabet $\Sigma = \{a, b\}$ that does contain the empty word, and for which there does not exist a DFA with at most two states.

    Take the language $\mathcal{L}(a \cup \lambda)$. It contains only the words a and λ. Because λ is in the language, we know that the starting state q_0 is a final state. But this cannot be the state that accepts a by having a loop to this same state, because that would imply that also aa, aaa, and so on. So we need at least a q_1 that is also a final state. In addition, because all words with b are not accepted, we need a sink q_2. So we need at least three states. In fact, it can be done with three states:

$M :=$ 

Discrete mathematics: multiple choice questions

21. For which $n \geq 1$ is K_n a tree?

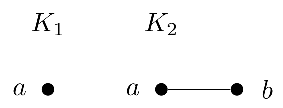
    (a) For $n = 1$.

    (b) For $n = 2$.

    (c) is correct (c) For $n = 1$ and $n = 2$.

    (d) This is never a tree.

    Answer (c) is correct. These are the two graphs:



A tree needs to be connected and it should not have any cycles. Both properties hold for these two graphs.

21. For which $n, m \geq 1$ is $K_{n,m}$ a tree?

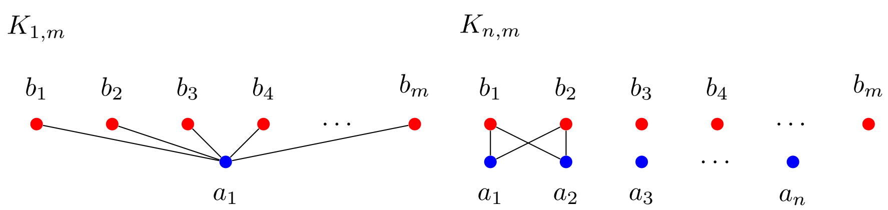
    (a) is correct (a) For $n = 1$ and any m, or $m = 1$ and any n.

    (b) For $n = 1$ and $m = 1$.

    (c) For $n = m$.

    (d) This is never a tree.

    Answer (a) is correct. These are the (partial) graphs $K_{1,m}$ and $K_{n,m}$ where $n \geq 2$:



It it clear that $K_{1,m}$ is connected and has no cycles, so it is a tree. However, although most of the edges in $K_{n,m}$ are not drawn, we see a cycle $a_1 \to b_2 \to a_2 \to b_1 \to a_1$, so it cannot be a tree.

For symmetry reasons, the results also follow for $K_{n,1}$ and $K_{n,m}$ with $m \geq 2$.

22. We define:

$$\begin{aligned} a_1 &= 1 \\ a_{n+1} &= a_n + n + 1 && \text{for } n \geq 1 \end{aligned}$$

What is a_4?

(a) 8

(b) 11

(c) 13

(d) is correct (d) None of the above.

Answer (d) is correct.

$$\begin{array}{lllllllll} a_1 & = & 1 \\ a_2 & = & a_{1+1} & = & a_1 + 1 + 1 & = & 1 + 1 + 1 & = & 3 \\ a_3 & = & a_{2+1} & = & a_2 + 2 + 1 & = & 3 + 2 + 1 & = & 6 \\ a_4 & = & a_{3+1} & = & a_3 + 3 + 1 & = & 6 + 3 + 1 & = & 10 \end{array}$$

22. We define:

$$a_1 = 0$$
$$a_{n+1} = a_n + n - 1 \qquad \text{for } n \geq 1$$

What is a_4?

(a) is correct (a) 3

(b) 5

(c) 6

(d) None of the above.

Answer (a) is correct.

$$\begin{array}{lllllllll} a_1 & = & 0 \\ a_2 & = & a_{1+1} & = & a_1 + 1 - 1 & = & 0 + 1 - 1 & = & 0 \\ a_3 & = & a_{2+1} & = & a_2 + 2 - 1 & = & 0 + 2 - 1 & = & 1 \\ a_4 & = & a_{3+1} & = & a_3 + 3 - 1 & = & 1 + 3 - 1 & = & 3 \end{array}$$

23. We have a proof by induction that shows that a predicate $P(n)$ holds for all n starting at zero. This proof follows the standard scheme, in which the base case just is about $P(0)$. What from this proof is used to establish that $P(3)$ holds?

(a) is correct (a) The proof of the base case, and the induction steps for $k = 0$, $k = 1$ and $k = 2$.

(b) The proof of the base case, and the induction steps for $k = 0$, $k = 1$, $k = 2$ and $k = 3$.

(c) The base case and the induction step for $k = 2$.

(d) The base case and the induction step for $k = 3$.

Answer (a) is correct. We need the base case to prove $P(0)$. Then, we use the induction step for $k = 0$ to prove $P(1)$ from $P(0)$. Next, we use the induction step for $k = 1$ to prove $P(2)$ from $P(1)$. Finally, we use the induction step for $k = 2$ to prove $P(3)$ from $P(2)$.

It is clear that none of the other options can be correct at the same time.

23. We want to prove that a predicate $P(n)$ holds for all $n \geq 0$, but we only manage to prove the induction step for $k \geq 2$. What can we do?

   (a) We cannot use induction to prove this, because in an induction proof the induction step needs to start at the same index as the statement that we want to prove.

(b) is correct

   (b) We can still use induction to prove this, if next to the base case for $P(0)$ we manage to prove extra base cases for $P(1)$ and $P(2)$.

   (c) We can still use induction to prove this, if next to the base case for $P(0)$ we manage to prove an extra base case for $P(1)$.

   (d) We can still use induction to prove this, we just use the base case $P(2)$ instead of $P(0)$.

   Answer (b) is correct. If the induction step only holds for $k \geq 2$, it means that we can prove $P(3)$, $P(4)$, and so on once we have a proof for $P(2)$. So we need to prove three base cases $P(0)$, $P(1)$, and $P(2)$.

   It is clear that none of the other options can be correct at the same time.

24. We want to count the number of ways that one can divide nine distinguishable objects into four non-distinguishable (possibly empty) groups. What can we best use for this?

   (a) Binomial coefficients.

   (b) Stirling numbers of the first kind.

(c) is correct

   (c) Stirling numbers of the second kind.

   (d) Bell numbers.

   Answer (c) is correct. The Stirling number of the second kind $\left\{ {n \atop m} \right\}$ is the number of ways to divide n distinguishable objects over m indistinguishable groups. In this case, we want to divide nine distinguishable elements over four non-distinguishable groups, which are possibly empty. Therefore, we have to count the number of ways to divide nine distinguishable elements over one group, two groups, three groups, and four groups. So the answer here would be

$$\left\{ {9 \atop 1} \right\} + \left\{ {9 \atop 2} \right\} + \left\{ {9 \atop 3} \right\} + \left\{ {9 \atop 4} \right\} = 1 + 255 + 3025 + 7770 = 11051$$

24. We want to count the number of ways that one can select a non-empty selection of at most four objects out of nine distinguishable objects. What can we best use for this?

(a) is correct

   (a) Binomial coefficients.

   (b) Stirling numbers of the first kind.

   (c) Stirling numbers of the second kind.

   (d) Bell numbers.

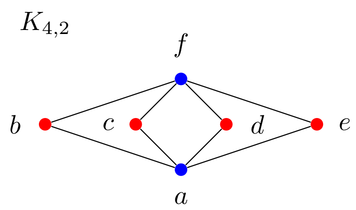
   Answer (a) is correct. The binomial coefficient $\binom{n}{m}$ is the number of ways to select m objects from n distinguishable objects. In this case, we want to select at most four out of nine distinguishable objects. So the answer here would be

$$\binom{9}{1} + \binom{9}{2} + \binom{9}{3} + \binom{9}{4} = 9 + 36 + 84 + 126 = 254$$

Discrete mathematics: open question

25. Give a planar connected graph that has an Eulerian circuit, but not a Hamiltonian circuit.

    Take $K_{4,2}$, drawn in a planar representation:

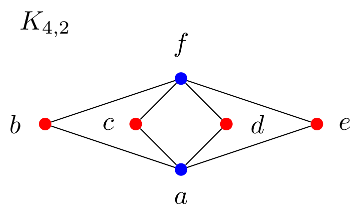
    

    Then

    - It is a connected graph, because there is a path between each pair of distinct vertices.
    - It is planar, because there are no crossing edges.
    - It has an Eulerian circuit because it is connected, it has at least two vertices, and the degrees of all vertices is even. For instance: $a \to b \to f \to c \to a \to d \to f \to e \to a$.
    - Because the graph is bipartite, we know that in any path, hence in particular in a Hamiltonian path, the red and blue vertices should change color after every edge. In our graph we have six vertices, so a Hamiltonian path should either be red-blue-red-blue-red-blue or blue-red-blue-red-blue-red, so we need three red and three blue vertices. But we only have two blue vertices, so it is not possible to create a Hamiltonian path in this graph.

25. Give a planar connected graph that has an Eulerian path, but not a Hamiltonian path.

    Take $K_{4,2}$, drawn in a planar representation:

    Then

- It is a connected graph, because there is a path between each pair of distinct vertices.
- It is planar, because there are no crossing edges.
- It has an Eulerian path because it is connected, it has at least two vertices, and the degrees of all vertices is even, so it has at most two vertices with odd degree. For instance: $a \to b \to f \to c \to a \to d \to f \to e$. Obviously, by adding the edge (e, a) this actually is even an Eulerian circuit.
- Now let us try to construct a Hamiltonian path. For reasons of symmetry between the four red vertices and the two blue vertices, we may assume that the path starts in the blue vertex a or in the red vertex b and that the edge (a, b) is the first edge of the path. Also because of symmetry, if we have to choose a new vertex to visit, we always choose the first unused vertex in the order of the alphabet.
  - Now if the path starts in a, then the path should go like this: $a \to b \to f \to c \to a$. However, we now have visited a twice, but didn't visit the vertices d and e. So this is not a Hamiltonian path.
  - Now if the path starts in b, then the path should go like this: $b \to a \to c \to f \to d \to a$. However, we now have visited a twice, but didn't visit the vertex e. So this is not a Hamiltonian path either.

  So both possibilities do not lead to a Hamiltonian path, so such a path doesn't exist in this graph.

Modal logic: multiple choice questions

26. What is the appropriate logic to formalize a statement of the form:

    This is allowed, but not required.

    (a) Epistemic logic.
    (b) Doxastic logic.
    (c) is correct — (c) Deontic logic.
    (d) Alethic logic (the logic of necessity and possibility).

Answer (c) is correct. Deontic logic is about obligations: something ought to be done, or something is permissible. Something that is 'allowed' is permissible. And something that is 'not required' is not obligatory.

Epistemic logic is about knowledge. Doxastic logic is about belief. And Alethic logic is about necessity and possibility. None of these are applicable.

26. What is the appropriate logic to formalize a statement of the form:

    This might be the case, but it also might not be the case.

    (a) Epistemic logic.
    (b) Doxastic logic.

(c) Deontic logic.

(d) is correct (d) Alethic logic (the logic of necessity and possibility).

Answer (d) is correct. Alethic logic is about necessity and possibility. If something 'might be the case' then it is possible, but not necessary.

Epistemic logic is about knowledge. Doxastic logic is about belief. Deontic logic is about obligation. None of these are applicable.

27. In the logic T, the axiom scheme $\Box f \to f$ holds for all formulas f. Is this logic appropriate for doxastic logic?

(a) is correct

(a) No, because it is possible to believe false things.
(b) No, because it is possible not to do things that are obligatory.
(c) Yes, because it is not possible to believe contradictory things.
(d) Yes, because it is not possible to require something that is forbidden.

Answer (a) is correct. Doxastic logic is about belief. So the axiom means *If I believe that f holds, then f holds*, which is certainly not the case due to the reason explained in the answer.

The second answer makes no sense because this is a claim that would be appropriate for deontic logic.

The third and the fourth answer make no sense because the answer is 'no'.

27. In the logic T, the axiom scheme $\Box f \to f$ holds for all formulas f. Is this logic appropriate for deontic logic?

(b) is correct

(a) No, because it is possible to believe false things.
(b) No, because it is possible not to do things that are obligatory.
(c) Yes, because it is not possible to believe contradictory things.
(d) Yes, because it is not possible to require something that is forbidden.

Answer (b) is correct. Deontic logic is about obligation. So the axiom means *If f ought to be done, then f is done*, which is certainly not the case due to the reason explained in the answer.

The first answer makes no sense because this is a claim that would be appropriate for doxastic logic.

The third and the fourth answer make no sense because the answer is 'no'.

28. If in a Kripke model we have $V(x_i) = \varnothing$, then what is necessarily the case?

(a) is correct

(a) $x_i \Vdash \Box\neg\Box f$ for all formulas f.
(b) $x_i \Vdash \Diamond\neg\Box f$ for all formulas f.
(c) $x_i \Vdash \neg\Box\Diamond f$ for all formulas f.
(d) $x_i \Vdash \Diamond\Diamond\neg f$ for all formulas f.

Answer (a) is correct. If $V(x_i) = \varnothing$ then automatically, all formulas starting with a $\Box$ hold, and all formulas starting with a $\Diamond$ do not hold. So $x_i \Vdash \Box\neg\Box f$ holds, $x_i \Vdash \Diamond\neg\Box f$ does not hold, and $x_i \Vdash \Diamond\Diamond\neg f$ does not hold. And because $\neg\Box\Diamond f$ is logically equivalent to $\Diamond\neg\Diamond f$, $x_i \Vdash \neg\Box\Diamond f$ also does not hold.

28. If in a Kripke model we have $V(x_i) = \varnothing$, then what is necessarily the case?

(a) $x_i \Vdash \neg\Box\Box f$ for all formulas f.

(b) $x_i \Vdash \Diamond\Box\neg f$ for all formulas f.

(c) is correct (c) $x_i \Vdash \Box\Diamond\neg f$ for all formulas f.

(d) $x_i \Vdash \Diamond\neg\Diamond f$ for all formulas f.

Answer (c) is correct. If $V(x_i) = \varnothing$ then automatically, all formulas starting with a $\Box$ hold, and all formulas starting with a $\Diamond$ do not hold. So $x_i \Vdash \Box\Diamond\neg f$ holds, $x_i \Vdash \Diamond\Box\neg f$ does not hold, and $x_i \Vdash \Diamond\neg\Diamond f$ does not hold. And because $\neg\Box\Box f$ is logically equivalent to $\Diamond\neg\Box f$, $x_i \Vdash \neg\Box\Box f$ also does not hold.

29. Which of the following is true in every LTL model?

(a) $a\,\mathcal{U}\,b \to a$

(b) is correct (b) $a\,\mathcal{U}\,b \to \mathcal{F}b$

(c) $a \to a\,\mathcal{U}\,b$

(d) $\mathcal{F}b \to a\,\mathcal{U}\,b$

Answer (b) is correct. Note that the formula should hold in all worlds of all LTL models.

Let $i \in \mathbb{N}$ and x_i be a world in an LTL model $\mathcal{M}$. If we assume $x_i \Vdash a\,\mathcal{U}\,b$, then it follows that there exists $j \in \mathbb{N}$ with $j \geq i$ such that $x_j \Vdash b$. But that is exactly the definition of $x_i \Vdash \mathcal{F}b$. So $x_i \Vdash a\,\mathcal{U}\,b \to \mathcal{F}b$ holds. And since we made no assumptions on x_i, this holds for all worlds in model $\mathcal{M}$. And since we also made no assumptions on model $\mathcal{M}$ this in fact holds for all LTL models.

For the three incorrect formulas f_j, it suffices to provide one model $\mathcal{M}$ containing at least one world x_i for which $x_i \Vdash f_j$ does not hold. So that is what we will do here.

The first answer makes no sense because we can take the model $\mathcal{M}$ defined by

$$V(x_0) = \{b\}, \quad V(x_i) = \varnothing \quad \text{for all } i \geq 1$$

Then $x_0 \Vdash a\,\mathcal{U}\,b$ because $x_0 \Vdash b$ and for all $k \in \{0, 1, \ldots, -1\} = \varnothing$, $x_k \Vdash a$ holds vacuously. However, $x_0 \Vdash a$ clearly doesn't hold. So $x_0 \Vdash a\,\mathcal{U}\,b \to a$ also doesn't hold.

The third answer makes no sense because we can take the model $\mathcal{M}$ defined by

$$V(x_0) = \{a\}, \quad V(x_i) = \varnothing \quad \text{for all } i \geq 1$$

Then $x_0 \Vdash a$ holds. But since there is no $j \geq 0$ such that $x_j \Vdash b$, automatically $x_0 \Vdash a\,\mathcal{U}\,b$ does not hold. Hence $x_0 \Vdash a \to a\,\mathcal{U}\,b$ does not hold.

The fourth answer makes no sense because we can take the model defined by

$$V(x_0) = \varnothing, \quad V(x_1) = \{b\}, \quad V(x_i) = \varnothing \quad \text{for all } i \geq 2$$

Then $x_0 \Vdash \mathcal{F}b$ holds, but $x_0 \Vdash a\,\mathcal{U}\,b$ does not hold, because $x_0 \Vdash a$ does not hold.

29. Which of the following is true in every LTL model?

(a) $a \,\mathcal{W}\, b \to \mathcal{F}a$

(b) $a \,\mathcal{W}\, b \to \mathcal{G}a$

(c) $\mathcal{F}a \to a \,\mathcal{W}\, b$

(d) is correct (d) $\mathcal{G}a \to a \,\mathcal{W}\, b$

Answer (d) is correct. Note that the formula should hold in all worlds of all LTL models.

Let $i \in \mathbb{N}$ and x_i be a world in an LTL model $\mathcal{M}$. If we assume $x_i \Vdash \mathcal{G}a$, then it follows that for all $j \in \mathbb{N}$ with $j \geq i$ that $x_j \Vdash a$ holds. So in particular it holds that for all $j \in \mathbb{N}$ with $j \geq i$ that $x_j \Vdash a$ or there exists a $j \in \mathbb{N}$ with $j \geq i$ such that $x_j \Vdash b$ and for all $k \in \{i, i+1, \ldots, j-1\}$ $x_k \Vdash a$ holds. But that implies that $x_i \Vdash a \,\mathcal{W}\, b$ holds. So $x_i \Vdash \mathcal{G}a \to a \,\mathcal{W}\, b$ holds. And since we made no assumptions on x_i, this holds for all worlds in model $\mathcal{M}$. And since we also made no assumptions on model $\mathcal{M}$ this in fact holds for all LTL models.

For the three incorrect formulas f_j, it suffices to provide one model $\mathcal{M}$ containing at least one world x_i for which $x_i \Vdash f_j$ does not hold. So that is what we will do here.

The first answer makes no sense because we can take the model $\mathcal{M}$ defined by

$$V(x_0) = \{b\}, \quad V(x_i) = \varnothing \quad \text{for all } i \geq 1$$

Then $x_0 \Vdash a \,\mathcal{W}\, b$ holds, but $x_0 \Vdash \mathcal{F}a$ doesn't hold because a never holds. So in particular $x_0 \Vdash a \,\mathcal{W}\, b \to \mathcal{F}a$ doesn't hold.

The second answer makes no sense because we can take the model $\mathcal{M}$ defined by

$$V(x_0) = \{b\}, \quad V(x_i) = \varnothing \quad \text{for all } i \geq 1$$

Then $x_0 \Vdash a \,\mathcal{W}\, b$ holds, but $x_0 \Vdash \mathcal{G}a$ doesn't hold because a never holds. So in particular $x_0 \Vdash a \,\mathcal{W}\, b \to \mathcal{G}a$ doesn't hold.

The third answer makes no sense because we can take the model $\mathcal{M}$ defined by

$$V(x_0) = \varnothing, \quad V(x_1) = \{a\}, \quad V(x_i) = \varnothing \quad \text{for all } i \geq 2$$

Then $x_0 \Vdash \mathcal{F}a$ holds in x_0, but $x_0 \Vdash a \,\mathcal{W}\, b$ does not hold, because b never holds and it is also not the case that a always holds. So in particular $x_0 \Vdash \mathcal{F}a \to a \,\mathcal{W}\, b$ doesn't hold.

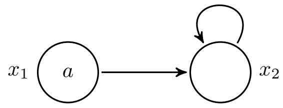
Modal logic: open question

30. Give a serial Kripke model $\mathcal{M}$ with:

$$\mathcal{M} \not\models a \to \Diamond a$$

Explain your answer, using both the symbols $\models$ and $\Vdash$.

$\mathcal{M} \quad :=$



Note that:

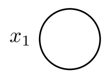
- This model is serial because every world has at least one outgoing arrow.
- Because $a \in L(x_1)$, it follows that $x_1 \Vdash a$.
- Because $a \notin L(x_2)$, and x_2 is the only accessible world from x_1, it follows that $x_1 \nVdash \Diamond a$.
- From this it follows that $x_1 \nVdash a \to \Diamond a$.
- And because with x_1 there is at least one world in $\mathcal{M}$ for which the formula doesn't hold, it follows that $\mathcal{M} \nvDash a \to \Diamond a$.

30. Give a non-reflexive Kripke model $\mathcal{M}$ with:

$$\mathcal{M} \vDash a \to \Diamond a$$

Explain your answer, using both the symbols $\vDash$ and $\Vdash$.

$\mathcal{M} \quad :=$



Note that:

- The model is non-reflexive because there exists a world, namely x_1, that does not have an outgoing arrow back to itself.
- Because $a \notin L(x_1)$, it follows that $x_1 \nVdash a$.
- From this, it immediately follows that $x_1 \Vdash a \to \Diamond a$.
- And because the formula holds for every world in the model, it follows that $\mathcal{M} \vDash a \to \Diamond a$.