Foundations of Software Reliability and Theoretical Computer Science
Informatik
Technical University of Munich

1
2
3
4
5
6
7
8
9
0

Registration number

Signature

Note:

- Cross your Registration number(with leading zero). It will be evaluated automatically.
- Sign in the corresponding signature field.

Logik

Exam:	IN2049 / Endterm	**Date:**	Wednesday 26th July, 2023
Examiner:	Prof. Dr. Javier Esparza	**Time:**	13:30 – 15:30

	P 1	P 2	P 3	P 4	P 5	P 6	P 7	P 8
I								
II								

Working instructions

- This exam consists of **20 pages** with a total of **8 problems**.
  Please make sure now that you received a complete copy of the exam.
- The total amount of achievable credits in this exam is 46 credits. Among them, 6 are bonus credits. In order to pass the exam, you will need at least 17 credits.
- Detaching pages from the exam is prohibited.
- Allowed resources: a single hand-written cheat-sheet (you can write on both sides of the sheet)
- **Answers are only accepted if the solution approach is documented.** Give a reason for each answer unless explicitly stated otherwise in the respective subproblem.
- Do not write with red or green colors nor use pencils.
- Physically turn off all electronic devices, put them into your bag and close the bag.

Left room from __________ to __________ / Early submission at __________

☐ Exam empty

cit-logik-1-20230726-E0100-01

Problem 1 Syntax of propositional logic (10 credits)

Recall that the syntax of propositional logic contains five logical operators: $\neg$ (negation), $\wedge$ (conjunction), $\vee$ (disjunction), $\rightarrow$ (implication) and $\leftrightarrow$ (bi-implication). **All the formulas in this problem are formulas over propositional logic**.

In every subproblem, you will be asked to prove or disprove a given claim. If the claim is true, give a proof. If the claim is false, give a counter-example and prove that it does not satisfy the claim.

0
1
2

a) Prove or disprove: For every formula F, there is an equivalent formula G which contains only $\vee$ and $\wedge$ as its logical operators.

The statement does not hold. Negation is not expressable by con- and disjunctions. Counterexample \neg A.

The truth table is

A | \neg A
0   1
1   0

Using a conjunction A & B, we would have
A | A & B
0 | 0
1 | B
and thus in particular no suitable constant for B.

And using a disjunction:
A | A | B
0 | B
1 | 1
which also does not give the desired table.

b) Prove or disprove: For every formula F, there is an equivalent formula G which contains only $\neg$ and $\rightarrow$ as its logical operators.

A | B = \neg A -> B
A & B = \neg (A -> \neg B)
A <-> B = A -> B & B -> A (which we can write as above)
Thus every formula can be written with only the two operators

c) Prove or disprove: For every formula F, there is an equivalent formula G which contains only $\leftrightarrow$ as its logical operator.

We show a stronger result in exercise d. If we could express formulae with only <->, we could also express them with <-> and negation, which we can not.

d) **Bonus subproblem**. Prove or disprove: For every formula *F*, there is an equivalent formula *G* which contains only ¬ and ↔ as its logical operators.
Hint: Consider the parity of the number of satisfying assignments.

Negation does not alter the parity of the number of satisfying assignments - if #sat is odd, 2^n - #sat is also odd (and that's the number of unsatisfying assignments, thus satisfying \neg F) and if #sat is even, 2^n - #sat is also even.

However, bi-implication always results in an even number of satisfying assignments.
W.l.o.g., consider A <-> B. We will do a case distinction on the parity of the number of satisfying assignments (#satA) of A.
Assume #satA is even.
Then there are #satA * #satB + #unsatA * #unsatB satisfying assignments for A <-> B. From #satA even we get #unsatA is even and thus #satA * #satB + #unsatA * #unsatB is even.

Assume #satA is odd. Now we need to do a case distinction on #satB.
Assume #satB is even, then the argument is symmetrical to the previous case.
Assume #satB is odd, then we get #satA * #satB is odd and #unsatA * #unsatB is odd but the sum of two odd number is even, which is what was to be proven.

Thus we can not express formulas with an odd number of satisfying assignments and more than one variable, e.g. A | B.

Problem 2 Sequent calculus (3 credits)

Give a proof tree using the rules of sequent calculus to prove that the following formula over propositional logic is a tautology.

$$(((P \wedge Q) \to R)) \to ((P \to R) \vee (Q \to R)))$$

```
---------Ax ----------Ax
P,Q => P P,Q => Q
-------------------- &R          -------------- Ax
P, Q => P & Q                R => R, R
---------------------------------------------------- -> L
(P & Q) -> R, P, Q=> R, R
--------------------------------------------------- -> R
(P & Q) -> R, P=> R, (Q -> R)
--------------------------------------------------- -> R
(P & Q) -> R => (P -> R), (Q -> R)
--------------------------------------------------- |R
(P & Q) -> R => (P -> R) | (Q -> R)
------------------------------------------------- -> R
=> (((P & Q) -> R) -> ((P -> R) | (Q -> R)))
```

Problem 3 Structures (6 credits)

Fix a signature $\tau = \{P, Q\}$ where P is a ternary relation symbol, and Q is a unary relation symbol. Consider the following formulas:

$$\varphi_1 = \forall x \exists y \, (P(x, x, y) \wedge \neg P(x, y, y))$$
$$\varphi_2 = \exists z \, \neg Q(z)$$
$$\varphi_3 = \forall x \forall y \exists z \, (P(x, z, y) \vee Q(z))$$

In each subproblem of this exercise, you will be asked to give a τ-structure satisfying some constraints. For each subproblem, if your solution is a τ-structure $\mathcal{E}$, present it in the following format:

- $\mathcal{U}^{\mathcal{E}} :=$
- $P^{\mathcal{E}} :=$
- $Q^{\mathcal{E}} :=$

0
1

a) Construct a τ-structure $\mathcal{A}$ such that $\mathcal{A} \models \varphi_1$, $\mathcal{A} \models \varphi_2$ and $\mathcal{A} \models \varphi_3$.

U := {a,b}
P := {(a,a,b), (b,b,a), (a,a,a), (b,b,a)}
Q := {}

0
1

b) Construct a τ-structure $\mathcal{B}$ such that $\mathcal{B} \models \varphi_1$, $\mathcal{B} \models \varphi_2$ and $\mathcal{B} \not\models \varphi_3$.

U := {a,b}
P := {(a,a,b), (b,b,a)}
Q := {}

c) Construct a τ-structure $\mathcal{C}$ such that $\mathcal{C} \not\models \varphi_1$, $\mathcal{C} \models \varphi_2$ and $\mathcal{C} \models \varphi_3$.

0
1

U := {a,b}
P := U³
Q := {}

d) Construct a τ-structure $\mathcal{D}$ such that $\mathcal{D} \not\models \varphi_1$, $\mathcal{D} \not\models \varphi_2$ and $\mathcal{D} \models \varphi_3$.

0
1

U := {a,b}
P := U³
Q := U

Page empty

cit-logik-1-20230726-E0100-07
cit-logik-1-20230726-E0100-07

Problem 4 First order logic modeling (10 credits)

Consider the following statements.

(S1) If a person has read somebody's work, then it was taught by somebody.

(S2) All wise philosophers have read Aristotle's work.

(S3) No philosopher is unwise.

(S4) There exists at least one philosopher.

(S5) Anybody who has taught somebody's work is a philosopher.

(S6) No wise person has taught Aristotle's work.

0 1 2 3

a) Formalize each of the statements S1, S2, S3, S4, S5 and S6 as closed formulas in first-order logic without equality. The only predicate symbols that you are allowed to use are two unary predicates P, W and two binary predicates R and T. The only constant symbol you can use is a.

$P(x)$ must be used to denote that x is a philosopher, $W(x)$ must be used to denote that x is wise, $R(x, y)$ must be used to denote that x has read y's work and $T(x, y)$ must be used to denote that x has taught y's work. a must be used for Aristotle.

S1: \forall p . (\exists s . R(p,s) --> \exists t . T(t,s))

S2: \forall p . (W(p) & P(p)) --> R(p, a)

S3: \neg \exists p . P(p) & \neg W(p)

S4: \exists p . P(p)

S5: \forall x . \exists y . T (x,y) --> P(x)

S6: \neg \exists p . (W(p) & T(p,a))

b) For every $i \in \{1, 2, 3, 4, 5, 6\}$, let F_i be the formalization of the statement Si in first-order logic without equality that you had obtained in the previous subproblem. Let $F = F_1 \wedge F_2 \wedge F_3 \wedge F_4 \wedge F_5 \wedge F_6$. Note that F is a closed formula. Convert F into an equivalent rectifed formula in prenex form F'.

0 1 2

```
F:
(\forall p . (\exists s . R(p,s)  --> \exists t . T(t,s))
& (\forall p . (W(p) & P(p)) --> R(p, a))
& (\neg \exists p . P(p) & \neg W(p))
& (\exists p . P(p))
& (\forall x . \exists y . T (x,y) --> P(x))
& (\neg \exists p . (W(p) & T(p,a)))

rectified:
(\forall p . (\exists s . R(p,s)  --> \exists t . T(t,s))
& (\forall d . (W(d) & P(d)) --> R(d, a))
& (\neg \exists e . P(e) & \neg W(e))
& (\exists f . P(f))
& (\forall x . \exists y . T (x,y) --> P(x))
& (\neg \exists g . (W(g) & T(g,a)))

prenex:
\exists f. \forall p. \forall s. \exists t. \forall d. \forall e. \forall x. \exists y. \forall g.
(\neg R (p,s) | T(t,s)) & ((W(d) & P(d)) --> R(d,a))) & (\neg(P(e) & \neg W(e))) & P(f) & (T(x,y) --> P(x)) & \neg (W(g) & T(g,a))
```


cit-logik-1-20230726-E0100-09
cit-logik-1-20230726-E0100-09

0
1
2

c) Consider the formula F' from the previous subproblem. Convert F' into an equisatisfiable closed formula in Skolem form G. **The matrix of the formula G must be in CNF where each clause is a Horn clause.**

Let b be a fresh constant, f be a fresh function.
\forall p. \forall s. \forall d. \forall e. \forall x. \forall g.
(\neg R (p,s) | T(f(p,s),s)) & ((W(d) & P(d)) --> R(d,a))) & (\neg P(e)) & (\neg W(e)) & P(b) & (T(x,(f(p,s,d,e,x)) --> P(x)) & \neg (W(g) & T(g,a))

Written as horn clauses:
(\neg R (p,s) | T(f(p,s),s)) & ((\neg W(d) | P(d)) | R(d,a))) & (\neg P(e)) & (\neg W(e)) & P(b) & (\negT(x,(f(p,s,d,e,x)) | P(x)) & (\neg W(g) | \neg T(g,a))

cit-logik-1-20230726-E0100-10
cit-logik-1-20230726-E0100-10
cit-logik-1-20230726-E0100-10

d) Consider the formula G from the previous subproblem. Construct a finite unsatisfiable subset K of the clause Herbrand expansion of G. You have to prove that K is not satisfiable by using the Horn satisfiability algorithm.

0 1 2 3

Notice that we have P(b) and \neg P(b) in the clause Herbrand expansion. (The second follows from using b as a term for e).

Using the algorithm, we want to set P(b) to true, and then get UNSAT as will not be able to set \neg P(b) to true.

Hence this indicates that the collection of statements S1, S2, S3, S4, S5 and S6 is unsatisfiable.

Problem 5 Herbrand theory (5 credits)

Let $\mathcal{A}$ be the structure defined as follows:

$$U_{\mathcal{A}} = \mathbb{N} \setminus \{0\}$$
$$c^{\mathcal{A}} = 5$$
$$f^{\mathcal{A}}(x, y) = x + y$$
$$(m, n) \in P^{\mathcal{A}} \Leftrightarrow m < n$$
$$n \in Q^{\mathcal{A}} \Leftrightarrow n \text{ is divisible by } 10$$

Let φ be the following formula:

$$\forall x \forall y \quad P(x, f(x, y)) \land \neg P(f(x, y), y) \land Q(f(c, c)).$$

0 1 2 3 4 5

Note that $\mathcal{A} \models \varphi$. Using the construction from the Fundamental theorem of predicate logic, construct a Herbrand structure $\mathcal{H}$ that is a model for φ based on $\mathcal{A}$.

U = {c,f(c,c),f(f(c,c))...} (or: {c} \cup {f^k(c,c) | k \in N})
where c=5
f^H = f^A
P^H = P^A
Q^H = Q^A

Problem 6 Predicate logic resolution (3 credits)

Let c and d be constant symbols, let P be a unary predicate symbol and let Q be a ternary predicate symbol. Consider the following clauses:

$$C_1 = \{\neg Q(c, x, y),\ P(g(x))\}$$
$$C_2 = \{Q(x, f(x), g(y))\}$$
$$C_3 = \{\neg Q(x, f(d), y),\ \neg P(y)\}$$

Use the predicate logic resolution to prove unsatisfiability of $C_1 \wedge C_2 \wedge C_3$. In each step explain which clauses you are considering, what is their most general unifier and what is their resolvent.

0
1
2
3

Unify C1, C3 with [g(x)/y] and get C4: {~Q(c,x,g(x)), ~Q(x,f(d),g(x))}
Unify C4, C2 with [x/d][x/y] and get C5: {~Q(c,x,g(x))}

Page empty

Problem 7 Compactness and completeness (5 credits)

0
1

a) Suppose Γ is a set of sentences in first-order logic with equality. For every $m \in \mathbb{N}$, construct a formula φ_m such that $\mathcal{A}$ is a model for $\Gamma \cup \{\varphi_m\}$ if and only if $\mathcal{A}$ is a model for Γ and the size of the universe of $\mathcal{A}$ is at least m.

\varphi_m := \exists x_m . \exists x_{m-1} ... \exists x_1 . &_i=1^m &_j=1^m (i != j) . ~neg (x_i = x_j)
(Read as: there are variables x_1 ... x_m s.t. for all i != j, x_i != x_j - i.e. there are m pairwise distinct variables)

0
1
2

b) Suppose Γ is a set of sentences in first-order logic with equality such that for every $m \in \mathbb{N}$, Γ has a model whose universe has at least m elements. Prove that Γ has a model with an infinite universe.
Hint: You are allowed to use the formulas $\{\varphi_m : m \in \mathbb{N}\}$ from the previous subproblem.

Assume that \Gamma would not have an infinite universe. Then there would be some largest universe of size n. Then, \Gamma \union {\varphi_{n+1}} would be unsatisfiable. However, this contradicts the statement that there is a model of size n+1 which we have from the assumption and thus \Gamma needs to have an infinite universe.

c) **Bonus subproblem**. Let T be a theory in first-order logic. We have shown in the lectures that if all models of T are elementary equivalent, then T is complete. Prove the following stronger statement here: If all *countable models* of T are elementary equivalent, then T is complete.

0
1
2

If all countable models of T are elementary equivalent, this also holds for all Herbrand models of T. We know that whenever a formula has an uncounctable model, it also has a corresponding countable Herbrand model, of which we assume the elementary equivalence.

If T has no uncountable models or all uncountable models are elementary equivalent, we have no proof obligation.

Thus, wlog, we can assume that there are two uncountable models A B of T, such that A and B are not elementary equivalent.

We know that we can construct corresponding countable Herbrand models of which we do have the elementary equivalence. Then A and B are equivalent "enough" so that we can use the argument from the lecture.

Problem 8 Consistency (4 credits)

Suppose T is a theory in first-order logic. T is said to be *consistent* if for every sentence S, it **does not** include both S and $\neg S$.

Consider the signature $\Sigma = \{+, *, \leq\}$. Recall that $(\mathbb{Z}, +, *, \leq)$ denotes the structure with universe $\mathbb{Z}$ and the standard intepretations for the symbols $+, *$ and $\leq$.

0
1

a) Give an example of a Σ-theory which is complete, decidable, contains $Th(\mathbb{Z}, +, *, \leq)$, and is not consistent.

Th(Z,+,*,\leq) \union {5+5 \leq 5}

0
1

b) Give an example of a Σ-theory which is complete, consistent, contains $Th(\mathbb{Z}, +, *, \leq)$, but is not decidable.

Th(R,+,*,\leq)

0
1
2

c) Give an example of a Σ-theory which is complete, decidable, consistent, but does not contain $Th(\mathbb{Z}, +, *, \leq)$.

Th(F_2,+,*,\leq) where F_2 = {0,1}

cit-logik-1-20230726-E0100-16

cit-logik-1-20230726-E0100-17

cit-logik-1-20230726-E0100-17

Additional space for solutions–clearly mark the (sub)problem your answers are related to and strike out invalid solutions.

☐ Page empty

cit-logik-1-20230726-E0100-18
cit-logik-1-20230726-E0100-18
cit-logik-1-20230726-E0100-18

cit-logik-1-20230726-E0100-19

☐ Page empty