

Bachelor/Master Exam Version V3B

First Name: _____

Last Name: _____

Immatriculation Number: _____

Course of Studies (please mark exactly one):

- Informatik Bachelor** **Mathematik Master**
 TK Master **Other:** _____

	Maximal Points	Achieved Points
Exercise 1	11	
Exercise 2	9	
Exercise 3	10	
Exercise 4	14	
Exercise 5	9	
Exercise 6	7	
Total	60	
Grade	-	

Instructions:

- On every sheet please give your **first name, last name**, and **immatriculation number**.
- You must solve the exam **without** consulting any **extra documents** (e.g., course notes).
- Make sure your answers are readable. Do not use **red or green pens or pencils**.
- Please answer the exercises on the **exercise sheets**. If needed, also use the back sides of the exercise sheets.
- Answers on extra sheets can only be accepted if they are clearly marked with your name, your immatriculation number, and the **exercise number**.
- **Cross out** text that should not be considered in the evaluation.
- Students that try to cheat **do not pass** the exam.
- At the end of the exam, please return **all sheets together with the exercise sheets**.

Exercise 1 (Theoretical Foundations):

(4 + 4 + 3 = 11 points)

Let $\varphi = p(0, s(0)) \wedge \forall X, Y (p(X, Y) \rightarrow p(s(X), s(s(Y)))) \wedge \neg p(s(0), s(s(0)))$ and $\psi = \exists Z p(Z, s(s(Z)))$ be formulas over the signature (Σ, Δ) with $\Sigma = \Sigma_0 \cup \Sigma_1$, $\Sigma_0 = \{0\}$, $\Sigma_1 = \{s\}$, and $\Delta = \Delta_2 = \{p\}$.

a) Prove that $\{\varphi\} \models \psi$ by means of SLD resolution.

Hint: First transform the formula $\varphi \wedge \neg\psi$ into an equivalent clause set.

b) Explicitly give a Herbrand model of the formula φ (i.e., specify a carrier and a meaning for all function and predicate symbols). You do not have to provide a proof for your answer.

c) Prove correctness of propositional resolution. You may assume that the following is correct: If \mathcal{K} is a set of clauses without variables, S is a model of \mathcal{K} , $K_1, K_2 \in \mathcal{K}$ and R is a resolvent of K_1 and K_2 , then S is a model of $\mathcal{K} \cup \{R\}$.

Solution: _____

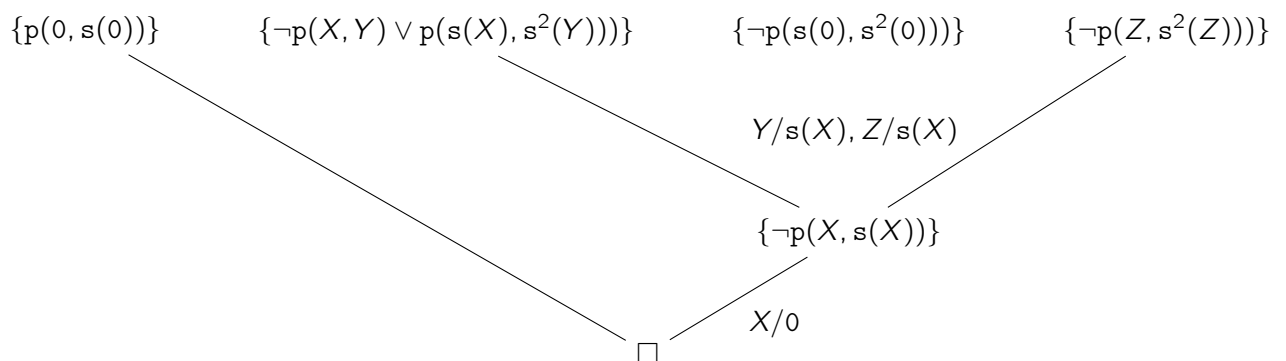
a)

$$\begin{aligned} \varphi \wedge \neg\psi &\Leftrightarrow p(0, s(0)) \wedge \forall X, Y (p(X, Y) \rightarrow p(s(X), s^2(Y))) \wedge \neg p(s(0), s^2(0)) \wedge \neg \exists Z (p(Z, s^2(Z))) \\ &\Leftrightarrow p(0, s(0)) \wedge \forall X, Y (\neg p(X, Y) \vee p(s(X), s^2(Y))) \wedge \neg p(s(0), s^2(0)) \wedge \neg \exists Z (p(Z, s^2(Z))) \\ &\Leftrightarrow p(0, s(0)) \wedge \forall X, Y (\neg p(X, Y) \vee p(s(X), s^2(Y))) \wedge \neg p(s(0), s^2(0)) \wedge \forall Z (\neg p(Z, s^2(Z))) \\ &\Leftrightarrow \forall X, Y, Z (p(0, s(0)) \wedge (\neg p(X, Y) \vee p(s(X), s^2(Y))) \wedge \neg p(s(0), s^2(0)) \wedge \neg p(Z, s^2(Z))) \end{aligned}$$

Thus, the equivalent clause set for $\varphi \wedge \neg\psi$ is

$$\{p(0, s(0))\}, \{\neg p(X, Y) \vee p(s(X), s^2(Y))\}, \{\neg p(s(0), s^2(0))\}, \{\neg p(Z, s^2(Z))\}.$$

We perform SLD resolution on this clause set to show $\{\varphi\} \models \psi$.



Hence, we have proven $\{\varphi\} \models \psi$.

□

b) We have $S \models \varphi$ for the Herbrand structure $S = (\mathcal{T}(\Sigma), \alpha)$ with $\alpha_0 = 0$, $\alpha_s(t) = s(t)$, and $\alpha_p = \{(s^i(0), s^{2i+1}(0)) \mid i \geq 0\}$

- c) We have to show that if \mathcal{K} is a set of clauses without variables and $\square \in Res^*(\mathcal{K})$, then \mathcal{K} is unsatisfiable. Let \mathcal{K} be a set of clauses without variables and $\square \in Res^*(\mathcal{K})$. Then we know that $Res^*(\mathcal{K})$ is unsatisfiable. Assume there is a model S for \mathcal{K} . By the given lemma we have that if S is a model of \mathcal{K} , then S is a model of $Res(\mathcal{K})$ and by induction it follows that S is a model of $Res^*(\mathcal{K})$. This is a contradiction to $Res^*(\mathcal{K})$ being unsatisfiable. Therefore, no such model S exists, so \mathcal{K} is unsatisfiable.
-

Exercise 2 (Procedural Semantics, SLD tree):

(5 + 4 = 9 points)

Consider the following Prolog program \mathcal{P} .

$a(X, Y) : \neg b(s(X)).$
 $a(X, Y) : \neg b(Y), !, c(X).$
 $a(s(X), s(Y)) : \neg a(X, Y).$
 $c(s(0)).$
 $b(0).$
 $b(1).$

a) Consider the following query:

?- $a(A, B).$

For the logic program \mathcal{P}' that results by **removing the cut** from \mathcal{P} , please show a successful computation for the query above (i.e., a computation of the form $(G, \emptyset) \vdash_{\mathcal{P}'}^+ (\square, \sigma)$ where $G = \{\neg a(A, B)\}$). It suffices to give substitutions only for those variables which are used to define the value of the variables A and B in the query.

b) Please give a graphical representation of the SLD tree for the query

?- $a(A, B).$

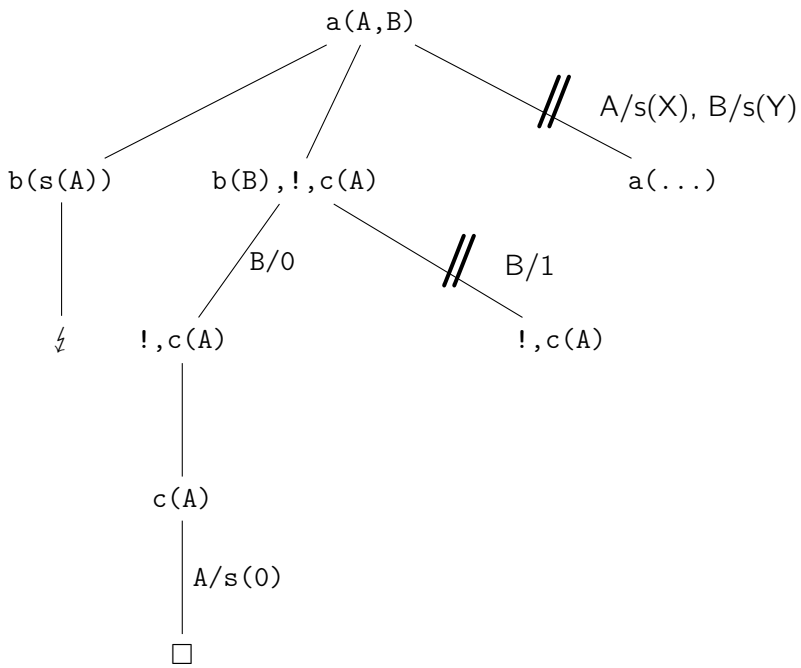
in the program \mathcal{P} **with the cut**. For every part of the tree that is cut off by evaluating **!**, please indicate the cut by marking the corresponding edge. For the cut-off parts only indicate the first cut-off goal, but do not evaluate further.

Solution: _____

a)

$$\begin{aligned}
 & (\{\neg a(A, B)\}, \emptyset) \\
 & \vdash_{\mathcal{P}} (\{\neg b(B), \neg c(A)\}, \emptyset) \\
 & \vdash_{\mathcal{P}} (\{\neg c(A)\}, \{B/0\}) \\
 & \vdash_{\mathcal{P}} (\square, \{A/s(0), B/0\})
 \end{aligned}$$

b)



Exercise 3 (Fixpoint Semantics):

(5 + 2 + 3 = 10 points)

Consider the following logic program \mathcal{P} over the signature (Σ, Δ) with $\Sigma = \Sigma_0 \cup \Sigma_1$, $\Sigma_0 = \{0\}$, $\Sigma_1 = \{s\}$, and $\Delta = \Delta_3 = \{p\}$.

$p(0, s(X), X)$.
 $p(s(X), s(Y), s(s(Z))) \text{ :- } p(X, Y, Z)$.

- a) For each $n \in \mathbb{N}$ explicitly give $\text{trans}_{\mathcal{P}}^n(\emptyset)$ in closed form, i.e., using a non-recursive definition.
- b) Compute the set $\text{lfp}(\text{trans}_{\mathcal{P}})$.
- c) Give $F[\mathcal{P}, \{\neg p(s(s(0))), s(s(X)), Y\}]$.

Solution: _____

Let G be the set of all ground terms, i.e., $G = \{s^i(0) \mid i \in \mathbb{N}\} = \mathcal{T}(\Sigma)$.

a)

$$\begin{aligned} \text{trans}_{\mathcal{P}}^0(\emptyset) &= \emptyset \\ \text{trans}_{\mathcal{P}}^1(\emptyset) &= \{p(0, s(t), t) \mid t \in G\} \\ \text{trans}_{\mathcal{P}}^2(\emptyset) &= \{p(s(0), s^2(t), s^2(t)) \mid t \in G\} \cup \text{trans}_{\mathcal{P}}^1(\emptyset) \\ \text{trans}_{\mathcal{P}}^3(\emptyset) &= \{p(s^2(0), s^3(t), s^4(t)) \mid t \in G\} \cup \text{trans}_{\mathcal{P}}^2(\emptyset) \\ &\vdots \\ \text{trans}_{\mathcal{P}}^n(\emptyset) &= \{p(s^i(0), s^{i+1}(t), s^{2i}(t)) \mid t \in G, 0 \leq i < n\} \end{aligned}$$

b)

$$\begin{aligned} \text{lfp}(\text{trans}_{\mathcal{P}}) &= \{p(s^i(0), s^{i+1}(t), s^{2i}(t)) \mid t \in G, i \geq 0\} \\ &= \{p(s^i(0), s^{i+j+1}(0), s^{2i+j}(0)) \mid i, j \geq 0\} \end{aligned}$$

c)

$$\begin{aligned} F[\mathcal{P}, \{\neg p(s(s(0))), p(s(s(X))), Y\}] &= \{p(s^2(0), s^3(t), s^4(t)) \mid t \in G\} \\ &= \{p(s^2(0), s^{3+i}(0), s^{4+i}(0)) \mid i \geq 0\} \end{aligned}$$

Exercise 4 (Definite Logic Programming):
(8 + 6 = 14 points)

- a) Implement the predicate `incr/2` in Prolog. This predicate can be used to identify the longest increasing prefix $[a_0, \dots, a_n]$ of a list $[a_0, \dots, a_n, a_{n+1}, \dots, a_m]$ such that for all $i \in \{0, \dots, n\}$ it holds that $a_i = a_0 + i$. The first argument of `incr` is the list to analyze. The second argument is the increasing prefix as described above.

As an example, for the list $[1, 2, 3, 2, 1]$ the result $[1, 2, 3]$ is computed (because $a_3 = 2$ is not equal to $1 + 3$). In Prolog, the corresponding call

```
incr([s(0), s(s(0)), s(s(s(0))), s(s(0)), s(0)], Res)
```

should return the only answer `Res = [s(0), s(s(0)), s(s(s(0)))]`.

Important: You may not use the cut, negation or any other predefined predicates in your implementation! However, you may implement auxiliary predicates.

- b) The Collatz sequence n_0, n_1, \dots for some initial value $n_0 > 0$ is defined as

$$n_{i+1} = \begin{cases} n_i/2, & \text{if } n_i \text{ is even} \\ n_i * 3 + 1, & \text{otherwise} \end{cases}$$

It can easily be seen that if $n_i = 1$ then the sequence will continue: $4, 2, 1, 4, 2, 1, \dots$. We define the function `collatz_len(n_0)` for a start value n_0 as the smallest i such that $n_i = 1$. Note that it is a famous open problem if all start values will eventually reach 1 or if there are other loops or diverging sequences.

Some examples for the length of the Collatz sequence:

- `collatz_len(1) = 0`, since $n_0 = 1$
- `collatz_len(2) = 1`, since $n_0 = 2, n_1 = 1$
- `collatz_len(3) = 7`, due to the Collatz sequence $3, 10, 5, 16, 8, 4, 2, 1$
- `collatz_len(4) = 2`

Implement the predicate `collatz_len/2` in Prolog that calculates the length of the Collatz sequence for a given initial value. It may behave arbitrarily if the length of the sequence starting in n_0 is not defined or $n_0 \leq 0$.

As an example, `collatz_len(3, Z)` gives the answer substitution `Z = 7`.

Hints:

- You may only use the built-in predicate `is/2`, the cut and the usual arithmetic operators such as `mod`, `+`, `*`, `-`, `/`.

Solution: _____

a) `incr([], []).`
`incr([X], [X]).`
`incr([X,s(X)|XS], [X|YS]) :- incr([s(X)|XS], YS).`
`incr([X,Y|_], [X]) :- neq(s(X), Y).`

`neq(0, s(_)).`
`neq(s(_), 0).`
`neq(s(X), s(Y)) :- neq(X, Y).`

b) `collatz(1,0) :- !.`
`collatz(X,Y) :- 0 is X mod 2, XP is X/2, collatz(XP,Z), Y is Z+1.`
`collatz(X,Y) :- 1 is X mod 2, XP is X*3+1, collatz(XP,Z), Y is Z+1.`

Exercise 5 (Meta-Programming):
(9 points)

Consider the function `havoc(t)`, which takes a ground term `t` and reverses the order of all arguments in `t` and any subterm of `t`. Please implement the predicate `havoc/2` that implements this function. For example, the query `?- havoc(a(b,c(d,e)), X)` yields the answer substitution `X = a(c(e,d),b)`.

Hints:

- You may use the built-in predicate `=.. /2`.
- You may use the built-in predicate `reverse/2` which reverses lists.

Solution: _____

```
havoc(Term,RTerm):- Term =.. [Head|ARGS],
    reverse(ARGS,RARGS), havoc_all(RARGS,RRARGS),
    RTerm =.. [Head|RRARGS].
```

```
havoc_all([], []).
havoc_all([T|XS], [TR|YS]):-havoc_all(XS,YS), havoc(T,TR).
```

Exercise 6 (Difference Lists):

(7 points)

Consider the following logic program \mathcal{P} .

$q(X) \text{ :- } p(X - [], [1,2,3,4] - []).$

$p([_ | X] - X, Y - Y).$

$p([A | X] - X2, Y - Y2) \text{ :- } p(X - X2, Y - [A | Y2]).$

Explicitly give the set of all ground terms t for which the query $?- q(t)$ succeeds. You do not have to provide a proof for your answer.

Solution: _____

$\{[4, 3, 2, 1, t] \mid t \text{ is a ground term}\}$

_____.