Logic and Verification
Computer Science
Technical University of Munich

Eexam
Place student sticker here

Note:

- During the attendance check a sticker containing a unique code will be put on this exam.
- This code contains a unique number that associates this exam with your registration number.
- This number is printed both next to the code and to the signature field in the attendance check list.

Lambda Calculus

Exam:	IN2358 / Endterm	**Date:**	Saturday 25th February, 2023
Examiner:	Prof. Tobias Nipkow, PhD	**Time:**	10:00 – 11:30

	P 1	P 2	P 3	P 4	P 5
I					

Working instructions

- This exam consists of **12 pages** with a total of **5 problems**.
  Please make sure now that you received a complete copy of the exam.
- The total amount of achievable credits in this exam is 40 credits.
- Detaching pages from the exam is prohibited.
- Allowed resources:
  - one **analog dictionary** English ↔ native language
  - one piece of A4 paper with hand-written notes on both sides
- Subproblems marked by * can be solved without results of previous subproblems.
- **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 _____

Problem 1 Programming with λ-terms (9 credits)

The goal of this exercise is to define a function height that computes the height of a tree in untyped λ-calculus. We use the fold encoding of trees, i.e. we have

$$\begin{aligned} \mathsf{tip} &= (\lambda f\ c.\ c) \\ \mathsf{node} &= (\lambda l\ a\ r.\ (\lambda f\ c.\ f\ (l\ f\ c)\ a\ (r\ f\ c))) \end{aligned}$$

where node $l\ a\ r$ represents a tree node with left subtree l, value a, and right subtree r.
In order to define height, we need a max function on Church numerals. You may assume that you are given the function iszero that reduces to a boolean which indicates whether the given Church numeral is $\underline{0}$. For booleans we use the encoding from the lecture. Furthermore, you may use succ and pred that compute the successor respectively the predecessor of a Church numeral. Note that the predecessor of $\underline{0}$ is $\underline{0}$.

0 1 2 3 a)* Write a function that computes the maximum of two church numerals.

0 1 2 3 b)* Implement the max function as a recursive function using fix without relying on the representation of numerals.

0 1 2 3 c)* Define the function height that calculates the height a tree in the encoding given above.

Problem 2 **Rewriting (8 credits)**

Let $\to \subseteq A \times A$ and $m : A \to \mathbb{N}$ such that $a \to a'$ implies $m(a) > m(a')$. Prove that if $\to$ is locally confluent, it is also confluent. Do not prove it by contradiction (as in the lecture notes) but by induction.

The proof must be given in the standard verbal style. However, it may be helpful to draw a diagram, in particular as a starting point.

0
1
2
3
4
5
6
7
8

Problem 3 **Reductions, Terms, and Types (9 credits)**

0 1 2

a)* Give a lambda term t that is well-typed in $\lambda^{\to}$ such that $t \to^{n}_{\text{cbv}} t'$ with t' in normal form and for all sequences of reductions $t \to^{m}_{\text{cbn}} t'$ it holds that $m > n$.

0 1 2 3

b)* Consider intuitionistic logic as introduced in the lecture where $\neg A$ is an abbreviation for $A \to \bot$. Give a lambda term that corresponds to the proof of the proposition

$$(\neg A \vee \neg B) \wedge A \to \neg B.$$

0 1 2

c)* The preservation property states that $\Gamma \vdash t : \tau$ and $t \to_\beta t'$ imply $\Gamma \vdash t' : \tau$. Does the converse, i.e. $\Gamma \vdash t' : \tau \wedge t \to_\beta t' \Longrightarrow \Gamma \vdash t : \tau$ also hold? Justify your answer with a proof or a counterexample.

0 1 2

d)* Using only the combinators S and K, find a combinatory logic term A for which it holds that $\mathsf{A}\ x\ y \to^{*}_{w} x\ y$.

Problem 4 **Let Polymorphism (8 credits)**

Consider $\lambda^{\rightarrow}$ extended with `let` and consider the following the typing problem

$$y : ?1 \vdash \texttt{let}\ \ f = \lambda x\ z.\ ((x\ z)\ (y\ z))\ \ \texttt{in}\ \ (f\ (\lambda u\ v.\, u)) : ?2$$

where both ?1 and ?2 have to be found.

Note: You may simply write type environments as $x_1 : \sigma_1, \ldots, x_n : \sigma_n$, without the square brackets.

a)* Find a most general type (no quantifiers) ?1 such that the above typing problem has a solution. We call this type τ.

0
1
2

b) Find a most general type schema σ with $y : \tau \vdash \lambda x\ z.\ ((x\ z)\ (y\ z)) : \sigma$. You may, but you do not need to draw a type derivation tree.

0
1
2
3

0 1 2 3

c) Draw the type derivation tree for

$$y : \tau, f : \sigma \vdash (f\ (\lambda u\ v.\, u)) : ?2$$

Of course, with the correct type for ?2.
Use only the introduction and elimination rules for $\rightarrow$ and $\forall$ and the standard assumption rule

$$\Gamma \vdash x : \Gamma(x)$$

Problem 5 Transforming Proof Trees (6 credits)

In this exercise, we consider classical logic with negation, disjunction, and implication. In particular, we use the following rules for negation

$$\frac{\Gamma, B \vdash A \qquad \Gamma, B \vdash \neg A}{\Gamma \vdash \neg B}\,\neg\mathrm{I} \qquad\qquad \frac{\Gamma \vdash \neg A \qquad \Gamma \vdash A}{\Gamma \vdash B}\,\neg\mathrm{E}$$

and we have the law of excluded middle

$$\frac{}{\Gamma \vdash A \vee \neg A}\,\mathrm{LEM}$$

It is common to represent $A \to B$ as $\neg A \vee B$ so we define a function $-^*$ that performs this replacement for a given formula:

$$\begin{aligned} A^* &= A && \text{for atomic A} \\ (\neg A)^* &= \neg A^* \\ (A \vee B)^* &= A^* \vee B^* \\ (A \to B)^* &= \neg A^* \vee B^* \end{aligned}$$

Prove that $\Gamma \vdash A \implies \Gamma^* \vdash A^*$ by induction on the derivation of $\Gamma \vdash A$. You only need to consider the cases where $\Gamma \vdash A$ was proved by $\to$E or $\to$I. You may use the weakening rule

$$\frac{\Gamma \vdash A}{\Gamma, B \vdash A}$$

but need to do so explicitly.

0
1
2
3
4
5
6

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