Logic and Verification
Computer Science
Technical University of Munich

**Esolution**
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.

$$\begin{aligned}\mathsf{minus} &= (\lambda m\ n.\ n\ \mathsf{pred}\ m)\\ \mathsf{max} &= (\lambda m\ n.\ (\mathsf{iszero}\ (\mathsf{minus}\ m\ n)\ n\ m))\end{aligned}$$

0 1 2 3

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

$$\mathsf{max} = (\lambda m\ n.\ \mathsf{fix}\ (\lambda f.\ \lambda m\ n.\ \mathsf{iszero}\ m\ n\ (\mathsf{iszero}\ n\ m\ (\mathsf{succ}\ (f\ (\mathsf{pred}\ n)\ (\mathsf{pred}\ m)))))\ m\ n)$$

0 1 2 3

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

$$\mathsf{height} = (\lambda t.\ t\ (\lambda l\ a\ r.\ \mathsf{succ}\ (\mathsf{max}\ l\ r))\ \underline{0})$$

## 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

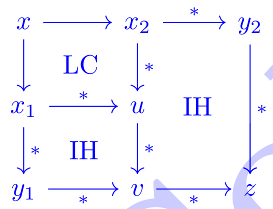
We prove

$$x \to^* y_1 \wedge x \to^* y_2 \Rightarrow \exists z.\, y_1 \to^* z \wedge y_2 \to^* z \qquad (2.1)$$

for all $y_1, y_2$ by complete or strong induction on $m(x)$ . We can assume that the property holds for all $x'$ with $m(x) > m(x')$ and need to show that it also holds for $x$ .

If either $x = y_1$ or $x = y_2$ then $z = y_2$ or $z = y_1$ do the job.

Otherwise there must be $x_1, x_2$ such that $x \to x_i \to^* y_i$ for $i = 1, 2$ . Local confluence implies that there is a $u$ such that $x_i \to^* u$ for $i = 1, 2$. Thus $m(x) > m(x_i)$ for $i = 1, 2$ . By IH, because $m(x) > m(x_1)$, there is a $v$ such that $y_1 \to^* v$ and $u \to^* v$ . By IH, because $m(x) > m(x_2)$, there is a $z$ such that $v \to^* z$ and $y_2 \to^* z$ . Because $y_1 \to^* z$ this concludes the proof.




where LC = local confluence.

## 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 $t \to_{\text{cbv}}^{n} t'$ with $t'$ in normal form and for all sequences of reductions $t \to_{\text{cbn}}^{m} t'$ it holds that $m > n$.

Consider $t = (\lambda f\ x.\ f\ (f\ x))\ ((\lambda g.\ g)\ (\lambda y.\ y))\ (\lambda z.\ z)$. Then, we have

$$\begin{aligned}
t &\to_{\text{cbv}} (\lambda f\ x.\ f\ (f\ x))\ (\lambda y.\ y)\ (\lambda z.\ z)\\
&\to_{\text{cbv}} (\lambda x.\ (\lambda y.\ y)\ ((\lambda y.\ y)\ x))\ (\lambda z.\ z)\\
&\to_{\text{cbv}} (\lambda y.\ y)\ ((\lambda y.\ y)\ (\lambda z.\ z))\\
&\to_{\text{cbv}} (\lambda y.\ y)\ (\lambda z.\ z)\\
&\to_{\text{cbv}} (\lambda z.\ z)
\end{aligned}$$

$$\begin{aligned}
t &\to_{\text{cbn}} (\lambda x.\ (\lambda g.\ g)\ (\lambda y.\ y)\ ((\lambda g.\ g)\ (\lambda y.\ y)\ x))\ (\lambda z.\ z)\\
&\to_{\text{cbn}} (\lambda g.\ g)\ (\lambda y.\ y)\ ((\lambda g.\ g)\ (\lambda y.\ y)\ (\lambda z.\ z))\\
&\to_{\text{cbn}} (\lambda y.\ y)\ ((\lambda g.\ g)\ (\lambda y.\ y))\ (\lambda z.\ z)\\
&\to_{\text{cbn}} ((\lambda g.\ g)\ (\lambda y.\ y))\ (\lambda z.\ z)\\
&\to_{\text{cbn}} (\lambda y.\ y)\ (\lambda z.\ z)\\
&\to_{\text{cbn}} (\lambda z.\ z)
\end{aligned}$$

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

$$(\neg A \lor \neg B) \land A \to \neg B.$$

$\lambda x\ y.\ \mathsf{case}\ (\pi_1\ x)\ (\lambda z.\ z\ (\pi_2\ x))\ (\lambda z.\ z\ y)$

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. $\Gamma \vdash t' : \tau \land t \to_\beta t' \implies \Gamma \vdash t : \tau$ also hold? Justify your answer with a proof or a counterexample.

This does not hold. Consider for example $t' = (\lambda x.\ x)$ and $t = (\lambda y\ x.\ x)\ (\lambda x.\ x\ x)$.

0 1 2

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

Use $\mathsf{S\ K\ K}$ which behaves like $\mathsf{I}$, then $\mathsf{S\ K\ K}\ x\ y \to_w^* x\ y$.

## Problem 4 Let Polymorphism (8 credits)

Consider $\lambda^{\to}$ 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 $\tau$.

0
1
2

$?1 = \tau = A \to B$

b) Find a most general type schema $\sigma$ 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

$\sigma = \forall C.\ (A \to B \to C) \to A \to C$

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

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

Let $\Gamma = y : \tau, f : \sigma$:

$$\dfrac{\dfrac{\Gamma \vdash f : \sigma}{\Gamma \vdash f : (A \rightarrow B \rightarrow A) \rightarrow A \rightarrow A}\ \forall E \qquad \dfrac{\dfrac{\Gamma, u : A, v : B \vdash u : A}{\Gamma, u : A \vdash \lambda v.\ u : A \rightarrow B \rightarrow A}\ {\rightarrow}I}{\Gamma \vdash \lambda u\ v.\ u : A \rightarrow B \rightarrow A}\ {\rightarrow}I}{\Gamma \vdash (f\ (\lambda u\ v.\ u)) : A \rightarrow A}\ {\rightarrow}E$$

## 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.

We prove $\Gamma \vdash A \implies \Gamma^* \vdash A^*$ by induction on the derivation of $\Gamma \vdash A$.

- **Case** $\to$E
  We get $\Gamma^* \vdash (A \to B)^* \iff \Gamma^* \vdash \neg A^* \vee B^*$ and $\Gamma^* \vdash A^*$ as induction hypotheses. We need to show that $\Gamma^* \vdash B^*$ holds .

$$\dfrac{\dfrac{}{\Gamma^* \vdash \neg A^* \vee B^*} \qquad \dfrac{\dfrac{}{\Gamma^*, \neg A^* \vdash \neg A^*} \qquad \dfrac{\dfrac{\text{I.H.}}{\Gamma^* \vdash A^*}}{\Gamma^*, \neg A^* \vdash A^*}}{\Gamma^*, \neg A^* \vdash B^*} \qquad \dfrac{}{\Gamma^*, B^* \vdash B^*}}{\Gamma^* \vdash B^*}$$

- **Case** $\to$I
  We get $\Gamma^*, A^* \vdash B^*$ as an induction hypothesis and need to show that $\Gamma^* \vdash (A \to B)^* \iff \Gamma^* \vdash \neg A^* \vee B^*$ holds .

$$\dfrac{\dfrac{}{\Gamma^* \vdash A^* \vee \neg A^*} \qquad \dfrac{\dfrac{\text{I.H.}}{\Gamma^*, A^* \vdash B^*}}{\Gamma^*, A^* \vdash \neg A^* \vee B^*} \qquad \dfrac{\dfrac{}{\Gamma^*, \neg A^* \vdash \neg A^*}}{\Gamma^*, \neg A^* \vdash \neg A^* \vee B^*}}{\Gamma^* \vdash \neg A^* \vee B^*}$$

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