Λάμβδα Λογισμός με Τύπους - I

Γιάννης Κασσιός

Δομή της Σημερινής Διάλεξης

- Συστήματα τύπων - Εισαγωγή
- Συστήματα χωρίς τύπους
- λ-λογισμός με απλούς τύπους
- Απλές επεκτάσεις του λ-λογισμού (εγγραφές, εναλλακτικοί τύποι, αναδρομή)
- Subtyping
- Επισκόπηση

Δομή της Σημερινής Διάλεξης

- Συστήματα τύπων - Εισαγωγή
- Συστήματα χωρίς τύπους
- λ-λογισμός με απλούς τύπους
- Απλές επεκτάσεις του λ-λογισμού (εγγραφές, εναλλακτικοί τύποι, αναδρομή)
- Subtyping
- Επισκόπηση

Δομή της Σημερινής Διάλεξης

- Συστήματα τύπων - Εισαγωγή
- Συστήματα χωρίς τύπους
- λ-λογισμός με απλούς τύπους
- Απλές επεκτάσεις του λ-λογισμού (εγγραφές, εναλλακτικοί τύποι, αναδρομή)
- Subtyping
- Επισκόπηση

Δομή της Σημερινής Διάλεξης

- Συστήματα τύπων - Εισαγωγή
- Συστήματα χωρίς τύπους
- λ-λογισμός με απλούς τύπους
- Απλές επεκτάσεις του λ-λογισμού (εγγραφές, εναλλακτικοί τύποι, αναδρομή)
- Subtyping
- Επισκόπηση

Δομή της Σημερινής Διάλεξης

- Συστήματα τύπων - Εισαγωγή
- Συστήματα χωρίς τύπους
- λ-λογισμός με απλούς τύπους
- Απλές επεκτάσεις του λ-λογισμού (εγγραφές, εναλλακτικοί τύποι, αναδρομή)
- Subtyping
- Επισκόπηση

Δομή της Σημερινής Διάλεξης

- Συστήματα τύπων - Εισαγωγή
- Συστήματα χωρίς τύπους
- λ-λογισμός με απλούς τύπους
- Απλές επεκτάσεις του λ-λογισμού (εγγραφές, εναλλακτικοί τύποι, αναδρομή)
- Subtyping
- Επισκόπηση

Συστήματα Τύπων

Ένα <u>σύστημα τύπων (type system)</u> είναι μία συντακτική μέθοδος απόδειξης ότι κάποιες υπολογιστικές συμπεριφορές εκλείπουν, με την κατηγοριοποίηση εκφράσεων ανάλογα με το τι είδους τιμές υπολογίζουν.

Benjamin Pierce: *Types and Programming Languages*, 2002

- Στατικά (static) συστήματα τύπων
- Μη πλήρης μέθοδος απόδειξης ορθότητας
- Προσέγγιση των αποτελεσμάτων ενός υπολογισμού

Συστήματα Τύπων

Ένα σύστημα τύπων (type system) είναι μία συντακτική μέθοδος απόδειξης ότι κάποιες υπολογιστικές συμπεριφορές εκλείπουν, με την κατηγοριοποίηση εκφράσεων ανάλογα με το τι είδους τιμές υπολογίζουν.

Benjamin Pierce: *Types and Programming Languages*, 2002

- Στατικά (static) συστήματα τύπων
- Μη πλήρης μέθοδος απόδειξης ορθότητας
- Προσέγγιση των αποτελεσμάτων ενός υπολογισμού

Ένα σύστημα τύπων (type system) είναι μία συντακτική μέθοδος απόδειξης ότι κάποιες υπολογιστικές συμπεριφορές εκλείπουν, με την κατηγοριοποίηση εκφράσεων ανάλογα με το τι είδους τιμές υπολογίζουν.

Benjamin Pierce: *Types and Programming Languages*, 2002

- Στατικά (static) συστήματα τύπων
- Μη πλήρης μέθοδος απόδειξης ορθότητας
- Προσέγγιση των αποτελεσμάτων ενός υπολογισμού

Συστήματα Τύπων

Ένα σύστημα τύπων (type system) είναι μία συντακτική μέθοδος απόδειξης ότι κάποιες υπολογιστικές συμπεριφορές εκλείπουν, με την κατηγοριοποίηση εκφράσεων ανάλογα με το τι είδους τιμές υπολογίζουν.

Benjamin Pierce: *Types and Programming Languages*, 2002

- Στατικά (static) συστήματα τύπων
- Μη πλήρης μέθοδος απόδειξης ορθότητας
- Προσέγγιση των αποτελεσμάτων ενός υπολογισμού

- Ο έλεγχος τύπων αποσκοπεί στο να βρει 'λανθασμένες' χρήσεις τιμών σε προγράμματα.
- Παράδειγμα: συνάρτηση `not`
  - έχει σχεδιαστεί για να εφαρμόζεται μόνο σε αληθοτιμές, π.χ.: `not true`
  - η χρήση της σε κάποια άλλη τιμή είναι 'λανθασμένη' και απαγορεύεται από ένα σύστημα τύπων: `not 3`
  - ένα λάθος τύπων σε ένα πρόγραμμα σημαίνει πιθανότατα ότι ο προγραμματιστής έχει κάνει ένα σοβαρό σημασιολογικό σφάλμα
- Ο έλεγχος τύπων είναι συντηρητικός (conservative): `if true then 5 else not 3`
- Ο έλεγχος τύπων δεν πιάνει όλα τα λάθη
  - διαίρεση με το μηδέν, δείκτης πίνακα εκτός ορίων
  - λεπτά σημασιολογικά λάθη ('bugs')

- Ο έλεγχος τύπων αποσκοπεί στο να βρει 'λανθασμένες' χρήσεις τιμών σε προγράμματα.
- Παράδειγμα: συνάρτηση `not`
  - έχει σχεδιαστεί για να εφαρμόζεται μόνο σε αληθοτιμές, π.χ.: `not true`
  - η χρήση της σε κάποια άλλη τιμή είναι 'λανθασμένη' και απαγορεύεται από ένα σύστημα τύπων: `not 3`
  - ένα λάθος τύπων σε ένα πρόγραμμα σημαίνει πιθανότατα ότι ο προγραμματιστής έχει κάνει ένα σοβαρό σημασιολογικό σφάλμα
- Ο έλεγχος τύπων είναι συντηρητικός (conservative): `if true then 5 else not 3`
- Ο έλεγχος τύπων δεν πιάνει όλα τα λάθη
  - διαίρεση με το μηδέν, δείκτης πίνακα εκτός ορίων
  - λεπτά σημασιολογικά λάθη ('bugs')

- Ο έλεγχος τύπων αποσκοπεί στο να βρει 'λανθασμένες' χρήσεις τιμών σε προγράμματα.
- Παράδειγμα: συνάρτηση `not`
  - έχει σχεδιαστεί για να εφαρμόζεται μόνο σε αληθοτιμές, π.χ.: `not true`
  - η χρήση της σε κάποια άλλη τιμή είναι 'λανθασμένη' και απαγορεύεται από ένα σύστημα τύπων: `not 3`
  - ένα λάθος τύπων σε ένα πρόγραμμα σημαίνει πιθανότατα ότι ο προγραμματιστής έχει κάνει ένα σοβαρό σημασιολογικό σφάλμα
- Ο έλεγχος τύπων είναι συντηρητικός (conservative): `if true then 5 else not 3`
- Ο έλεγχος τύπων δεν πιάνει όλα τα λάθη
  - διαίρεση με το μηδέν, δείκτης πίνακα εκτός ορίων
  - λεπτά σημασιολογικά λάθη ('bugs')

- Ο έλεγχος τύπων αποσκοπεί στο να βρει 'λανθασμένες' χρήσεις τιμών σε προγράμματα.
- Παράδειγμα: συνάρτηση `not`
  - έχει σχεδιαστεί για να εφαρμόζεται μόνο σε αληθοτιμές, π.χ.: `not true`
  - η χρήση της σε κάποια άλλη τιμή είναι 'λανθασμένη' και απαγορεύεται από ένα σύστημα τύπων: `not 3`
  - ένα λάθος τύπων σε ένα πρόγραμμα σημαίνει πιθανότατα ότι ο προγραμματιστής έχει κάνει ένα σοβαρό σημασιολογικό σφάλμα
- Ο έλεγχος τύπων είναι συντηρητικός (conservative): `if true then 5 else not 3`
- Ο έλεγχος τύπων δεν πιάνει όλα τα λάθη
  - διαίρεση με το μηδέν, δείκτης πίνακα εκτός ορίων
  - λεπτά σημασιολογικά λάθη ('bugs')

- Ο έλεγχος τύπων αποσκοπεί στο να βρει 'λανθασμένες' χρήσεις τιμών σε προγράμματα.
- Παράδειγμα: συνάρτηση `not`
  - έχει σχεδιαστεί για να εφαρμόζεται μόνο σε αληθοτιμές, π.χ.: `not true`
  - η χρήση της σε κάποια άλλη τιμή είναι 'λανθασμένη' και απαγορεύεται από ένα σύστημα τύπων: `not 3`
  - ένα λάθος τύπων σε ένα πρόγραμμα σημαίνει πιθανότατα ότι ο προγραμματιστής έχει κάνει ένα σοβαρό σημασιολογικό σφάλμα
- Ο έλεγχος τύπων είναι συντηρητικός (conservative): `if true then 5 else not 3`
- Ο έλεγχος τύπων δεν πιάνει όλα τα λάθη
  - διαίρεση με το μηδέν, δείκτης πίνακα εκτός ορίων
  - λεπτά σημασιολογικά λάθη ('bugs')

- Ο έλεγχος τύπων αποσκοπεί στο να βρει ‘λανθασμένες’ χρήσεις τιμών σε προγράμματα.
- Παράδειγμα: συνάρτηση `not`
  - έχει σχεδιαστεί για να εφαρμόζεται μόνο σε αληθοτιμές, π.χ.:
    `not true`
  - η χρήση της σε κάποια άλλη τιμή είναι ‘λανθασμένη’ και απαγορεύεται από ένα σύστημα τύπων:
    `not 3`
  - ένα λάθος τύπων σε ένα πρόγραμμα σημαίνει πιθανότατα ότι ο προγραμματιστής έχει κάνει ένα σοβαρό σημασιολογικό σφάλμα
- Ο έλεγχος τύπων είναι συντηρητικός (conservative):
  `if true then 5 else not 3`
- Ο έλεγχος τύπων δεν πιάνει όλα τα λάθη
  - διαίρεση με το μηδέν, δείκτης πίνακα εκτός ορίων
  - λεπτά σημασιολογικά λάθη (‘bugs’)

- Εντοπισμός σημασιολογικών σφαλμάτων
- Αφαίρεση
- Τεκμηρίωση
- Ασφάλεια (= κανένας λανθασμένος υπολογισμός δε συμβαίνει σε προγράμματα που επιτρέπει το σύστημα τύπων)
  - σύστημα τύπων + δυναμικός έλεγχος

- Εντοπισμός σημασιολογικών σφαλμάτων
- Αφαίρεση
- Τεκμηρίωση
- Ασφάλεια (= κανένας λανθασμένος υπολογισμός δε συμβαίνει σε προγράμματα που επιτρέπει το σύστημα τύπων)
  - σύστημα τύπων + δυναμικός έλεγχος

- Εντοπισμός σημασιολογικών σφαλμάτων
- Αφαίρεση
- Τεκμηρίωση
- Ασφάλεια (= κανένας λανθασμένος υπολογισμός δε συμβαίνει σε προγράμματα που επιτρέπει το σύστημα τύπων)
    - σύστημα τύπων + δυναμικός έλεγχος

- Εντοπισμός σημασιολογικών σφαλμάτων
- Αφαίρεση
- Τεκμηρίωση
- Ασφάλεια (= κανένας λανθασμένος υπολογισμός δε συμβαίνει σε προγράμματα που επιτρέπει το σύστημα τύπων)
    - σύστημα τύπων + δυναμικός έλεγχος

- Ένα σύστημα τύπων απορρίπτει εκφράσεις της γλώσσας
- Πολλές φορές ένα απλό σύστημα τύπων είναι πολύ συντηρητικό
  - π.χ. λίστες χωρίς πολυμορφισμό
- Εκφραστικότητα ενός συστήματος τύπων: πόσες μη λανθασμένες εκφράσεις επιτρέπει το σύστημα
- Η εκφραστικότητα είναι το κύριο μέλημα της έρευνας στους τύπους

- Ένα σύστημα τύπων απορρίπτει εκφράσεις της γλώσσας
- Πολλές φορές ένα απλό σύστημα τύπων είναι πολύ συντηρητικό
  - π.χ. λίστες χωρίς πολυμορφισμό
- Εκφραστικότητα ενός συστήματος τύπων: πόσες μη λανθασμένες εκφράσεις επιτρέπει το σύστημα
- Η εκφραστικότητα είναι το κύριο μέλημα της έρευνας στους τύπους

- Ένα σύστημα τύπων απορρίπτει εκφράσεις της γλώσσας
- Πολλές φορές ένα απλό σύστημα τύπων είναι πολύ συντηρητικό
  - π.χ. λίστες χωρίς πολυμορφισμό
- Εκφραστικότητα ενός συστήματος τύπων: πόσες μη λανθασμένες εκφράσεις επιτρέπει το σύστημα
- Η εκφραστικότητα είναι το κύριο μέλημα της έρευνας στους τύπους

- Ένα σύστημα τύπων απορρίπτει εκφράσεις της γλώσσας
- Πολλές φορές ένα απλό σύστημα τύπων είναι πολύ συντηρητικό
  - π.χ. λίστες χωρίς πολυμορφισμό
- Εκφραστικότητα ενός συστήματος τύπων: πόσες μη λανθασμένες εκφράσεις επιτρέπει το σύστημα
- Η εκφραστικότητα είναι το κύριο μέλημα της έρευνας στους τύπους

- Ο λ-λογισμός που είδαμε + ‘τιμές’ + κανόνες αποτίμησης
  - <u>λειτουργική σημασιολογία (operational semantics)</u>
- Σύνταξη:

$t ::=$		όρος	$v ::=$		τιμή
	x	μεταβλητή		$\lambda x.t$	λ-αφαίρεση
	$\lambda x.t$	λ-αφαίρεση			
	$t\ t$	εφαρμογή			

- Αποτίμηση (κανόνες πρόθυμης αποτίμησης):

$$\frac{t_1 \longrightarrow t_1'}{t_1\ t_2 \longrightarrow t_1'\ t_2}(\text{εφ.1}) \qquad \frac{t \longrightarrow t'}{v\ t \longrightarrow v\ t'}(\text{εφ.2})$$

$$\frac{}{(\lambda x.t)v \longrightarrow t[x := v]}(\beta\text{-μετ.})$$

- Αναγωγή:

$$\longrightarrow^* = \bigcup_{i=0}^{\infty} \longrightarrow^i$$

- Ο λ-λογισμός που είδαμε + ‘τιμές’ + κανόνες αποτίμησης
  - λειτουργική σημασιολογία (operational semantics)
- Σύνταξη:

$$
\begin{array}{llr}
t ::= & & \text{όρος} \\
 & x & \text{μεταβλητή} \\
 & \lambda x.t & \lambda\text{-αφαίρεση} \\
 & t\ t & \text{εφαρμογή}
\end{array}
\qquad
\begin{array}{llr}
v ::= & & \text{τιμή} \\
 & \lambda x.t & \lambda\text{-αφαίρεση}
\end{array}
$$

- Αποτίμηση (κανόνες πρόθυμης αποτίμησης):

$$\frac{t_1 \longrightarrow t_1'}{t_1\ t_2 \longrightarrow t_1'\ t_2}(\text{εφ.}1) \qquad \frac{t \longrightarrow t'}{v\ t \longrightarrow v\ t'}(\text{εφ.}2)$$

$$\frac{}{(\lambda x.t)v \longrightarrow t[x := v]}(\beta\text{-μετ.})$$

- Αναγωγή:

$$\longrightarrow^* = \bigcup_{i=0}^{\infty} \longrightarrow^i$$

- Ο λ-λογισμός που είδαμε + 'τιμές' + κανόνες αποτίμησης
  - λειτουργική σημασιολογία (operational semantics)
- Σύνταξη:

$$
\begin{array}{llr}
t ::= & & \text{όρος} \\
 & x & \text{μεταβλητή} \\
 & \lambda x.t & \lambda\text{-αφαίρεση} \\
 & t\ t & \text{εφαρμογή}
\end{array}
\qquad
\begin{array}{llr}
v ::= & & \text{τιμή} \\
 & \lambda x.t & \lambda\text{-αφαίρεση}
\end{array}
$$

- Αποτίμηση (κανόνες πρόθυμης αποτίμησης):

$$\frac{t_1 \longrightarrow t_1'}{t_1\ t_2 \longrightarrow t_1'\ t_2}(\text{εφ.1}) \qquad \frac{t \longrightarrow t'}{v\ t \longrightarrow v\ t'}(\text{εφ.2})$$

$$\frac{}{(\lambda x.t)v \longrightarrow t[x := v]}(\beta\text{-μετ.})$$

- Αναγωγή:

$$\longrightarrow^* = \bigcup_{i=0}^{\infty} \longrightarrow^i$$

- Ο λ-λογισμός που είδαμε + 'τιμές' + κανόνες αποτίμησης
  - λειτουργική σημασιολογία (operational semantics)
- Σύνταξη:

$t ::=$		όρος	$v ::=$		τιμή
	x	μεταβλητή		$\lambda x.t$	λ-αφαίρεση
	$\lambda x.t$	λ-αφαίρεση			
	$t\ t$	εφαρμογή			

- Αποτίμηση (κανόνες πρόθυμης αποτίμησης):

$$\frac{t_1 \longrightarrow t_1'}{t_1\ t_2 \longrightarrow t_1'\ t_2}(\text{εφ.}1) \qquad \frac{t \longrightarrow t'}{v\ t \longrightarrow v\ t'}(\text{εφ.}2)$$

$$\frac{}{(\lambda x.t)v \longrightarrow t[x := v]}(\beta\text{-μετ.})$$

- Αναγωγή:

$$\longrightarrow^* = \bigcup_{i=0}^{\infty} \longrightarrow^i$$

- Ο λ-λογισμός που είδαμε + 'τιμές' + κανόνες αποτίμησης
  - λειτουργική σημασιολογία (operational semantics)
- Σύνταξη:

$$
\begin{array}{llr}
t ::= & & \text{όρος} \\
 & x & \text{μεταβλητή} \\
 & \lambda x.t & \lambda\text{-αφαίρεση} \\
 & t\ t & \text{εφαρμογή}
\end{array}
\qquad
\begin{array}{llr}
v ::= & & \text{τιμή} \\
 & \lambda x.t & \lambda\text{-αφαίρεση}
\end{array}
$$

- Αποτίμηση (κανόνες πρόθυμης αποτίμησης):

$$
\frac{t_1 \longrightarrow t_1'}{t_1\ t_2 \longrightarrow t_1'\ t_2}(\text{εφ.}1)
\qquad
\frac{t \longrightarrow t'}{v\ t \longrightarrow v\ t'}(\text{εφ.}2)
$$

$$
\frac{}{(\lambda x.t)v \longrightarrow t[x := v]}(\beta\text{-μετ.})
$$

- Αναγωγή:

$$
\longrightarrow^* = \bigcup_{i=0}^{\infty} \longrightarrow^i
$$

- Σύνταξη:

t ::=	…		v ::=	…
	`true`	αληθές		`true`
	`false`	ψευδές		`false`
	`if` t `then` t `else` t	έλεγχος		

- Αποτίμηση:

$$\frac{}{\texttt{if true then } t_1 \texttt{ else } t_2 \longrightarrow t_1}\ (\text{αλ.})$$

$$\frac{}{\texttt{if false then } t_1 \texttt{ else } t_2 \longrightarrow t_2}\ (\text{ψευδ.})$$

$$\frac{t_1 \longrightarrow t_1'}{\texttt{if } t_1 \texttt{ then } t_2 \texttt{ else } t_3 \longrightarrow \texttt{if } t_1' \texttt{ then } t_2 \texttt{ else } t_3}\ (\text{έλ.})$$

- Σύνταξη:

$$
\begin{array}{rlll}
t ::= & \ldots & & v ::= \ldots \\
 & \texttt{true} & \text{αληθές} & \quad \texttt{true} \\
 & \texttt{false} & \text{ψευδές} & \quad \texttt{false} \\
 & \texttt{if } t \texttt{ then } t \texttt{ else } t & \text{έλεγχος} &
\end{array}
$$

- Αποτίμηση:

$$\frac{}{\texttt{if true then } t_1 \texttt{ else } t_2 \longrightarrow t_1}\ (\text{αλ.})$$

$$\frac{}{\texttt{if false then } t_1 \texttt{ else } t_2 \longrightarrow t_2}\ (\text{ψευδ.})$$

$$\frac{t_1 \longrightarrow t_1'}{\texttt{if } t_1 \texttt{ then } t_2 \texttt{ else } t_3 \longrightarrow \texttt{if } t_1' \texttt{ then } t_2 \texttt{ else } t_3}\ (\text{έλ.})$$

- Σύνταξη:

t ::=	...		v ::=	...
	`0`	μηδέν		`0`
	`succ` t	διάδοχος		`succ` v
	`iszero` t	έλεγχος μηδέν		

- Αποτίμηση:

$$\frac{t \longrightarrow t'}{\texttt{succ}\ t \longrightarrow \texttt{succ}\ t'}\text{(διάδ.)} \qquad \frac{t \longrightarrow t'}{\texttt{iszero}\ t \longrightarrow \texttt{iszero}\ t'}\text{(έλ.0)}$$

$$\frac{}{\texttt{iszero 0} \longrightarrow \texttt{true}}\text{(0)} \qquad \frac{}{\texttt{iszero (succ}\ v\texttt{)} \longrightarrow \texttt{false}}\text{(μη 0)}$$

- Γράφουμε `0,1,2...` αντί για `0,succ 0,succ(succ 0)...`
- Υποθέτουμε την ύπαρξη συνηθισμένων πράξεων + - * / κτλ.

- Σύνταξη:

$$
\begin{array}{llr}
t ::= & \ldots & \\
 & \texttt{0} & \text{μηδέν} \\
 & \texttt{succ } t & \text{διάδοχος} \\
 & \texttt{iszero } t & \text{έλεγχος μηδέν}
\end{array}
\qquad
\begin{array}{ll}
v ::= & \ldots \\
 & \texttt{0} \\
 & \texttt{succ } v
\end{array}
$$

- Αποτίμηση:

$$
\frac{t \longrightarrow t'}{\texttt{succ } t \longrightarrow \texttt{succ } t'}(\text{διάδ.})
\qquad
\frac{t \longrightarrow t'}{\texttt{iszero } t \longrightarrow \texttt{iszero } t'}(\text{έλ.0})
$$

$$
\frac{}{\texttt{iszero 0} \longrightarrow \texttt{true}}(0)
\qquad
\frac{}{\texttt{iszero (succ } v\texttt{)} \longrightarrow \texttt{false}}(\text{μη } 0)
$$

- Γράφουμε `0,1,2...` αντί για `0,succ 0,succ(succ 0)...`
- Υποθέτουμε την ύπαρξη συνηθισμένων πράξεων + - * / κτλ.

- Σύνταξη:

$$t ::= \quad \ldots \qquad\qquad v ::= \quad \ldots$$

`0`	μηδέν	`0`
`succ` t	διάδοχος	`succ` v
`iszero` t	έλεγχος μηδέν	

- Αποτίμηση:

$$\frac{t \longrightarrow t'}{\texttt{succ}\ t \longrightarrow \texttt{succ}\ t'}\,(\text{διάδ.}) \qquad \frac{t \longrightarrow t'}{\texttt{iszero}\ t \longrightarrow \texttt{iszero}\ t'}\,(\text{έλ.0})$$

$$\frac{}{\texttt{iszero 0} \longrightarrow \texttt{true}}\,(0) \qquad \frac{}{\texttt{iszero (succ}\ v\texttt{)} \longrightarrow \texttt{false}}\,(\text{μη } 0)$$

- Γράφουμε `0,1,2...` αντί για `0,succ 0,succ(succ 0)...`
- Υποθέτουμε την ύπαρξη συνηθισμένων πράξεων + - * / κτλ.

- Σύνταξη:

$$
\begin{array}{llr}
t ::= & \dots & \\
 & \mathtt{0} & \text{μηδέν} \\
 & \mathtt{succ}\ t & \text{διάδοχος} \\
 & \mathtt{iszero}\ t & \text{έλεγχος μηδέν}
\end{array}
\qquad
\begin{array}{ll}
v ::= & \dots \\
 & \mathtt{0} \\
 & \mathtt{succ}\ v
\end{array}
$$

- Αποτίμηση:

$$
\frac{t \longrightarrow t'}{\mathtt{succ}\ t \longrightarrow \mathtt{succ}\ t'}(\text{διάδ.})
\qquad
\frac{t \longrightarrow t'}{\mathtt{iszero}\ t \longrightarrow \mathtt{iszero}\ t'}(\text{έλ.0})
$$

$$
\frac{}{\mathtt{iszero}\ \mathtt{0} \longrightarrow \mathtt{true}}(0)
\qquad
\frac{}{\mathtt{iszero}\ (\mathtt{succ}\ v) \longrightarrow \mathtt{false}}(\text{μη } 0)
$$

- Γράφουμε `0,1,2...` αντί για `0,succ 0,succ(succ 0)...`
- Υποθέτουμε την ύπαρξη συνηθισμένων πράξεων + - * / κτλ.

$$\mathtt{A} = \mathtt{if\ iszero((}\lambda\mathtt{x.x)1)\ then\ 2\ else\ 3}$$

$$\cfrac{\cfrac{\cfrac{}{(\lambda\mathtt{x.x})\mathtt{1} \longrightarrow \mathtt{1}}(\beta\text{-μετ.})}{\mathtt{iszero((}\lambda\mathtt{x.x)1)} \longrightarrow \mathtt{iszero\ 1}}(\text{έλ.0})}{\mathtt{A} \longrightarrow \mathtt{if\ iszero\ 1\ then\ 2\ else\ 3}}(\text{έλ.})$$

$$\cfrac{\cfrac{}{\mathtt{iszero\ 1} \longrightarrow \mathtt{false}}(\text{μη 0})}{\mathtt{if\ iszero\ 1\ then\ 2\ else\ 3} \longrightarrow \mathtt{if\ false\ then\ 2\ else\ 3}}(\text{έλ.})$$

$$\cfrac{}{\mathtt{if\ false\ then\ 2\ else\ 3} \longrightarrow \mathtt{3}}(\text{ψευδ.})$$

$$\mathtt{A} \longrightarrow^* \mathtt{3}$$

$$\mathtt{A} = \mathtt{if\ iszero((\lambda x.x)1)\ then\ 2\ else\ 3}$$

$$\dfrac{\dfrac{}{\mathtt{(\lambda x.x)1} \longrightarrow \mathtt{1}}(\beta\text{-μετ.})}{\dfrac{\mathtt{iszero((\lambda x.x)1)} \longrightarrow \mathtt{iszero\ 1}}{\mathtt{A} \longrightarrow \mathtt{if\ iszero\ 1\ then\ 2\ else\ 3}}(\text{έλ.})}(\text{έλ.0})$$

$$\dfrac{\dfrac{}{\mathtt{iszero\ 1} \longrightarrow \mathtt{false}}(\text{μη 0})}{\mathtt{if\ iszero\ 1\ then\ 2\ else\ 3} \longrightarrow \mathtt{if\ false\ then\ 2\ else\ 3}}(\text{έλ.})$$

$$\dfrac{}{\mathtt{if\ false\ then\ 2\ else\ 3} \longrightarrow \mathtt{3}}(\text{ψευδ.})$$

$$\mathtt{A} \longrightarrow^{*} \mathtt{3}$$

$$\mathtt{A} = \mathtt{if\ iszero((}\lambda\mathtt{x.x)1)\ then\ 2\ else\ 3}$$

$$\dfrac{\dfrac{}{(\lambda\mathtt{x.x)1} \longrightarrow \mathtt{1}}\,(\beta\text{-μετ.})}{\dfrac{\mathtt{iszero((}\lambda\mathtt{x.x)1)} \longrightarrow \mathtt{iszero\ 1}\,(\text{έλ.0})}{\mathtt{A} \longrightarrow \mathtt{if\ iszero\ 1\ then\ 2\ else\ 3}}\,(\text{έλ.})}$$

$$\dfrac{\dfrac{}{\mathtt{iszero\ 1} \longrightarrow \mathtt{false}}\,(\text{μη 0})}{\mathtt{if\ iszero\ 1\ then\ 2\ else\ 3} \longrightarrow \mathtt{if\ false\ then\ 2\ else\ 3}}\,(\text{έλ.})$$

$$\dfrac{}{\mathtt{if\ false\ then\ 2\ else\ 3} \longrightarrow \mathtt{3}}\,(\text{ψευδ.})$$

$$\mathtt{A} \longrightarrow^{*} \mathtt{3}$$

$$A = \texttt{if iszero}((\lambda x.x)1)\ \texttt{then 2 else 3}$$

$$\dfrac{\dfrac{\dfrac{}{(\lambda x.x)1 \longrightarrow 1}(\beta\text{-μετ.})}{\texttt{iszero}((\lambda x.x)1) \longrightarrow \texttt{iszero 1}}(\text{έλ.0})}{A \longrightarrow \texttt{if iszero 1 then 2 else 3}}(\text{έλ.})$$

$$\dfrac{\dfrac{}{\texttt{iszero 1} \longrightarrow \texttt{false}}(\text{μη 0})}{\texttt{if iszero 1 then 2 else 3} \longrightarrow \texttt{if false then 2 else 3}}(\text{έλ.})$$

$$\dfrac{}{\texttt{if false then 2 else 3} \longrightarrow 3}(\text{ψευδ.})$$

$$A \longrightarrow^* 3$$

$$\mathtt{A} = \mathtt{if\ iszero((}\lambda\mathtt{x.x)1)\ then\ 2\ else\ 3}$$

$$\dfrac{\dfrac{}{(\lambda\mathtt{x.x})\mathtt{1} \longrightarrow \mathtt{1}}\,(\beta\text{-μετ.})}{\dfrac{\mathtt{iszero((}\lambda\mathtt{x.x)1)} \longrightarrow \mathtt{iszero\ 1}}{\mathtt{A} \longrightarrow \mathtt{if\ iszero\ 1\ then\ 2\ else\ 3}}\,(\text{έλ.})}\,(\text{έλ.0})$$

$$\dfrac{\dfrac{}{\mathtt{iszero\ 1} \longrightarrow \mathtt{false}}\,(\text{μη 0})}{\mathtt{if\ iszero\ 1\ then\ 2\ else\ 3} \longrightarrow \mathtt{if\ false\ then\ 2\ else\ 3}}\,(\text{έλ.})$$

$$\dfrac{}{\mathtt{if\ false\ then\ 2\ else\ 3} \longrightarrow \mathtt{3}}\,(\text{ψευδ.})$$

$$\mathtt{A} \longrightarrow^{*} \mathtt{3}$$

$$\dfrac{\dfrac{}{(\lambda \texttt{x.x})\texttt{1} \longrightarrow \texttt{1}}(\beta\text{-μετ.})}{\texttt{if } (\lambda \texttt{x.x})\texttt{1} \texttt{ then 2 else 3} \longrightarrow \texttt{if 1 then 2 else 3}}(\text{έλ.})$$

Σφάλμα υπολογισμού!

- Κανένας κανόνας αποτίμησης δεν εφαρμόζεται πλέον
- Δεν έχουμε φτάσει σε τιμή
- Δεν έχουμε περίπτωση μη τερματισμού
- Αυτό είναι το είδος σφαλμάτων που θέλουμε να `πιάνει΄ ένα σύστημα τύπων

$$\dfrac{\dfrac{}{(\lambda \texttt{x.x})\texttt{1} \longrightarrow \texttt{1}}(\beta\text{-μετ.})}{\texttt{if } (\lambda \texttt{x.x})\texttt{1} \texttt{ then 2 else 3} \longrightarrow \texttt{if 1 then 2 else 3}}(\text{έλ.})$$

Σφάλμα υπολογισμού!

- Κανένας κανόνας αποτίμησης δεν εφαρμόζεται πλέον
- Δεν έχουμε φτάσει σε τιμή
- Δεν έχουμε περίπτωση μη τερματισμού
- Αυτό είναι το είδος σφαλμάτων που θέλουμε να 'πιάνει' ένα σύστημα τύπων

$$\dfrac{\dfrac{}{(\lambda \mathtt{x.x})\mathtt{1} \longrightarrow \mathtt{1}}(\beta\text{-μετ.})}{\mathtt{if}\ (\lambda \mathtt{x.x})\mathtt{1}\ \mathtt{then}\ \mathtt{2}\ \mathtt{else}\ \mathtt{3} \longrightarrow \mathtt{if}\ \mathtt{1}\ \mathtt{then}\ \mathtt{2}\ \mathtt{else}\ \mathtt{3}}(\text{έλ.})$$

Σφάλμα υπολογισμού!

- Κανένας κανόνας αποτίμησης δεν εφαρμόζεται πλέον
- Δεν έχουμε φτάσει σε τιμή
- Δεν έχουμε περίπτωση μη τερματισμού
- Αυτό είναι το είδος σφαλμάτων που θέλουμε να 'πιάνει' ένα σύστημα τύπων

$$\dfrac{\overline{(\lambda \mathtt{x.x})\mathtt{1} \longrightarrow \mathtt{1}}\,(\beta\text{-μετ.})}{\mathtt{if}\ (\lambda \mathtt{x.x})\mathtt{1}\ \mathtt{then}\ \mathtt{2}\ \mathtt{else}\ \mathtt{3} \longrightarrow \mathtt{if}\ \mathtt{1}\ \mathtt{then}\ \mathtt{2}\ \mathtt{else}\ \mathtt{3}}\,(\text{έλ.})$$

Σφάλμα υπολογισμού!

- Κανένας κανόνας αποτίμησης δεν εφαρμόζεται πλέον
- Δεν έχουμε φτάσει σε τιμή
- Δεν έχουμε περίπτωση μη τερματισμού
- Αυτό είναι το είδος σφαλμάτων που θέλουμε να ‘πιάνει’ ένα σύστημα τύπων

$$\dfrac{\overline{(\lambda \mathtt{x}.\mathtt{x})\mathtt{1} \longrightarrow \mathtt{1}}^{(\beta\text{-μετ.})}}{\mathtt{if}\ (\lambda \mathtt{x}.\mathtt{x})\mathtt{1}\ \mathtt{then}\ \mathtt{2}\ \mathtt{else}\ \mathtt{3} \longrightarrow \mathtt{if}\ \mathtt{1}\ \mathtt{then}\ \mathtt{2}\ \mathtt{else}\ \mathtt{3}}(\text{έλ.})$$

Σφάλμα υπολογισμού!

- Κανένας κανόνας αποτίμησης δεν εφαρμόζεται πλέον
- Δεν έχουμε φτάσει σε τιμή
- Δεν έχουμε περίπτωση μη τερματισμού
- Αυτό είναι το είδος σφαλμάτων που θέλουμε να ‘πιάνει’ ένα σύστημα τύπων

- Το σύστημα διαχωρίζει τις εκφράσεις σε όρθες (well-typed) και εσφαλμένες (ill-typed)
- Ασφάλεια (safety): ένα βήμα αποτίμησης σε ορθή έκφραση δίνει μια νέα ορθή έκφραση
- Πρόοδος (progress): μία ορθή έκφραση είτε είναι τιμή είτε υπάρχει κανόνας αποτίμησης που μπορεί να εφαρμοστεί σε αυτήν

- Το σύστημα διαχωρίζει τις εκφράσεις σε όρθες (well-typed) και εσφαλμένες (ill-typed)
- Ασφάλεια (safety): ένα βήμα αποτίμησης σε ορθή έκφραση δίνει μια νέα ορθή έκφραση
- Πρόοδος (progress): μία ορθή έκφραση είτε είναι τιμή είτε υπάρχει κανόνας αποτίμησης που μπορεί να εφαρμοστεί σε αυτήν

- Το σύστημα διαχωρίζει τις εκφράσεις σε όρθες (well-typed) και εσφαλμένες (ill-typed)
- Ασφάλεια (safety): ένα βήμα αποτίμησης σε ορθή έκφραση δίνει μια νέα ορθή έκφραση
- Πρόοδος (progress): μία ορθή έκφραση είτε είναι τιμή είτε υπάρχει κανόνας αποτίμησης που μπορεί να εφαρμοστεί σε αυτήν

- Ορίζουμε δύο ακόμα συντακτικές κατηγορίες:
  - τύπος (type) (T)
  - περιβάλλον (environment,context) (Γ)
- Το περιβάλλον είναι ένα σύνολο υποθέσεων για τους τύπους μεταβλητών
  - υπόθεση: $x : T$
- $\Gamma \vdash t : T$ αν οι υποθέσεις στο Γ ισχύουν, τότε ο όρος t έχει τύπο T
  - γράφουμε $\vdash t : T$ για $\Gamma = \emptyset$

- Ορίζουμε δύο ακόμα συντακτικές κατηγορίες:
  - <u>τύπος (type)</u> (T)
  - <u>περιβάλλον (environment,context)</u> (Γ)
- Το περιβάλλον είναι ένα σύνολο υποθέσεων για τους τύπους μεταβλητών
  - υπόθεση: $x : T$
- $\Gamma \vdash t : T$ αν οι υποθέσεις στο Γ ισχύουν, τότε ο όρος t έχει τύπο T
  - γράφουμε $\vdash t : T$ για $\Gamma = \emptyset$

- Ορίζουμε δύο ακόμα συντακτικές κατηγορίες:
  - <u>τύπος (type)</u> (T)
  - <u>περιβάλλον (environment,context)</u> (Γ)
- Το περιβάλλον είναι ένα σύνολο υποθέσεων για τους τύπους μεταβλητών
  - υπόθεση: $x : T$
- $\Gamma \vdash t : T$ αν οι υποθέσεις στο Γ ισχύουν, τότε ο όρος t έχει τύπο T
  - γράφουμε $\vdash t : T$ για $\Gamma = \emptyset$

- Σύνταξη:

$$
\begin{array}{lll}
t ::= & \ldots & \\
 & \lambda x\colon T.t & \lambda\text{-αφαίρεση} \\
T ::= & & \text{τύποι} \\
 & \mathtt{Nat} & \text{φυσικοί} \\
 & \mathtt{Bool} & \text{αληθοτιμές} \\
 & T \rightarrow T & \text{συναρτήσεις}
\end{array}
\qquad
\begin{array}{lll}
v ::= & \ldots & \\
 & \lambda x\colon T.t & \\
\Gamma ::= & & \text{περιβάλλοντα} \\
 & \emptyset & \text{κενό περιβάλλον} \\
 & \Gamma, x : T & \text{ένωση}
\end{array}
$$

- Αποτίμηση:

$$\frac{}{(\lambda x\colon T.t)v \longrightarrow t[x := v]}\,(\beta\text{-μετ.})$$

- Σύνταξη:

$t ::=$	$\dots$		$v ::=$	$\dots$	
	$\lambda x\colon T.t$	λ-αφαίρεση		$\lambda x\colon T.t$	
$T ::=$		τύποι	$\Gamma ::=$		περιβάλλοντα
	`Nat`	φυσικοί		$\emptyset$	κενό περιβάλλον
	`Bool`	αληθοτιμές		$\Gamma, x : T$	ένωση
	$T \to T$	συναρτήσεις			

- Αποτίμηση:

$$\frac{}{(\lambda x\colon T.t)v \longrightarrow t[x := v]}\,(\beta\text{-μετ.})$$

$$\frac{}{\Gamma \vdash \mathtt{0}:\mathtt{Nat}}(\text{τ.0}) \quad \frac{}{\Gamma \vdash \mathtt{true}:\mathtt{Bool}}(\text{τ.αλ.}) \quad \frac{}{\Gamma \vdash \mathtt{false}:\mathtt{Bool}}(\text{τ.ψευδ.})$$

$$\frac{\Gamma \vdash t : \mathtt{Nat}}{\Gamma \vdash \mathtt{succ}\ \ t : \mathtt{Nat}}(\text{τ.διάδ.}) \quad \frac{\Gamma \vdash t : \mathtt{Nat}}{\Gamma \vdash \mathtt{iszero}\ \ t : \mathtt{Bool}}(\text{τ.έλ.0})$$

$$\frac{\Gamma \vdash t_1 : \mathtt{Bool}\ \ \Gamma \vdash t_2 : T\ \ \Gamma \vdash t_3 : T}{\Gamma \vdash \mathtt{if}\ \ t_1\ \ \mathtt{then}\ \ t_2\ \ \mathtt{else}\ \ t_3\ \ :\ \ T}(\text{τ.έλ})$$

$$\frac{(x : T) \in \Gamma}{\Gamma \vdash x : T}(\text{τ.μετ.}) \quad \frac{\Gamma \vdash t_1 : T_1 \to T_2 \quad \Gamma \vdash t_2 : T_1}{\Gamma \vdash t_1 t_2 : T_2}(\text{τ.εφαρμ.})$$

$$\frac{\Gamma, x : T_1 \vdash t : T_2}{\Gamma \vdash \lambda x : T_1 . t\ \ :\ \ T_1 \to T_2}(\text{τ.}\lambda\text{-αφ.})$$

$$\frac{}{\Gamma \vdash \texttt{0:Nat}}\text{(τ.0)} \quad \frac{}{\Gamma \vdash \texttt{true:Bool}}\text{(τ.αλ.)} \quad \frac{}{\Gamma \vdash \texttt{false:Bool}}\text{(τ.ψευδ.)}$$

$$\frac{\Gamma \vdash t : \texttt{Nat}}{\Gamma \vdash \texttt{succ}\ \ t : \texttt{Nat}}\text{(τ.διάδ.)} \quad \frac{\Gamma \vdash t : \texttt{Nat}}{\Gamma \vdash \texttt{iszero}\ \ t : \texttt{Bool}}\text{(τ.έλ.0)}$$

$$\frac{\Gamma \vdash t_1 : \texttt{Bool}\ \Gamma \vdash t_2 : T\ \Gamma \vdash t_3 : T}{\Gamma \vdash \texttt{if}\ \ t_1\ \ \texttt{then}\ \ t_2\ \ \texttt{else}\ \ t_3\ :\ T}\text{(τ.έλ)}$$

$$\frac{(x : T) \in \Gamma}{\Gamma \vdash x : T}\text{(τ.μετ.)} \quad \frac{\Gamma \vdash t_1 : T_1 \rightarrow T_2 \quad \Gamma \vdash t_2 : T_1}{\Gamma \vdash t_1 t_2 : T_2}\text{(τ.εφαρμ.)}$$

$$\frac{\Gamma, x : T_1 \vdash t : T_2}{\Gamma \vdash \lambda x : T_1.t\ :\ T_1 \rightarrow T_2}\text{(τ.λ-αφ.)}$$

$$\frac{}{\Gamma \vdash \mathtt{0}:\mathtt{Nat}}(\text{τ.0}) \quad \frac{}{\Gamma \vdash \mathtt{true}:\mathtt{Bool}}(\text{τ.αλ.}) \quad \frac{}{\Gamma \vdash \mathtt{false}:\mathtt{Bool}}(\text{τ.ψευδ.})$$

$$\frac{\Gamma \vdash t : \mathtt{Nat}}{\Gamma \vdash \mathtt{succ}\ t : \mathtt{Nat}}(\text{τ.διάδ.}) \quad \frac{\Gamma \vdash t : \mathtt{Nat}}{\Gamma \vdash \mathtt{iszero}\ t : \mathtt{Bool}}(\text{τ.έλ.0})$$

$$\frac{\Gamma \vdash t_1 : \mathtt{Bool} \quad \Gamma \vdash t_2 : T \quad \Gamma \vdash t_3 : T}{\Gamma \vdash \mathtt{if}\ t_1\ \mathtt{then}\ t_2\ \mathtt{else}\ t_3\ :\ T}(\text{τ.έλ})$$

$$\frac{(x : T) \in \Gamma}{\Gamma \vdash x : T}(\text{τ.μετ.}) \quad \frac{\Gamma \vdash t_1 : T_1 \rightarrow T_2 \quad \Gamma \vdash t_2 : T_1}{\Gamma \vdash t_1 t_2 : T_2}(\text{τ.εφαρμ.})$$

$$\frac{\Gamma, x : T_1 \vdash t : T_2}{\Gamma \vdash \lambda x : T_1.t \ :\ T_1 \rightarrow T_2}(\text{τ.}\lambda\text{-αφ.})$$

$$\dfrac{\dfrac{\dfrac{(\mathtt{x}:\mathtt{Bool}) \in \{(\mathtt{x}:\mathtt{Bool})\}}{\mathtt{x}:\mathtt{Bool} \vdash \mathtt{x}:\mathtt{Bool}}\text{(τ.μετ.)}}{\vdash \lambda\mathtt{x}{:}\mathtt{Bool}.\mathtt{x} : \mathtt{Bool}\to\mathtt{Bool}}\text{(τ.λ-αφ.)} \qquad \dfrac{}{\vdash \mathtt{true}:\mathtt{Bool}}\text{(τ.αλ.)}}{\vdash (\lambda\mathtt{x}{:}\mathtt{Bool}.\mathtt{x})\mathtt{true} : \mathtt{Bool}}\text{(τ.εφαρμ.)}$$

$$\dfrac{\dfrac{}{\mathtt{true}:\mathtt{Bool}}\text{(τ.αλ.)} \qquad \dfrac{}{\mathtt{1}:\mathtt{Nat}}\text{(τ.μη 0)} \qquad \dfrac{}{\mathtt{false}:\mathtt{Bool}}\text{(τ.ψευδ.)}}{\mathtt{if\ true\ then\ 1\ else\ false} : X}X$$

$$\dfrac{\dfrac{\dfrac{(\mathtt{x} : \mathtt{Bool}) \in \{(\mathtt{x} : \mathtt{Bool})\}}{\mathtt{x} : \mathtt{Bool} \vdash \mathtt{x} : \mathtt{Bool}}\,(\text{τ.μετ.})}{\vdash \lambda \mathtt{x}{:}\mathtt{Bool}.\mathtt{x} \;:\; \mathtt{Bool}{\rightarrow}\mathtt{Bool}}\,(\text{τ.}\lambda\text{-αφ.}) \qquad \dfrac{}{\vdash \mathtt{true} : \mathtt{Bool}}\,(\text{τ.αλ.})}{\vdash (\lambda \mathtt{x}{:}\mathtt{Bool}.\mathtt{x})\mathtt{true} \;:\; \mathtt{Bool}}\,(\text{τ.εφαρμ.})$$

$$\dfrac{\dfrac{}{\mathtt{true} : \mathtt{Bool}}\,(\text{τ.αλ.}) \qquad \dfrac{}{\mathtt{1} : \mathtt{Nat}}\,(\text{τ.μη 0}) \qquad \dfrac{}{\mathtt{false} : \mathtt{Bool}}\,(\text{τ.ψευδ.})}{\mathtt{if\ true\ then\ 1\ else\ false} \;:\; X}\,X$$

- Απάλειψη τύπων (erasure): σβήσιμο τύπων από τις εκφράσεις
  - $erase(x) = x$
  - $erase(\lambda x : T.t) = \lambda x.erase(t)$
  - $erase(t_1 t_2) = erase(t_1)erase(t_2)$
- Αν $t \longrightarrow t'$ τότε $erase(t) \longrightarrow erase(t')$
- Αν $erase(t) \longrightarrow m$ τότε υπάρχει όρος t' ώστε $erase(t') = m$ και $t \longrightarrow t'$
- Ο όρος λ-λογισμού χωρίς τύπους m έχει τυπο (is typable) σε ένα λογισμό L με τύπους αν υπάρχει περιβάλλον Γ, όρος t και τύπος T στο L, ώστε:
  - $\Gamma \vdash t : T$
  - $erase(t) = m$

- Απάλειψη τύπων (erasure): σβήσιμο τύπων από τις εκφράσεις
  - $erase(x) = x$
  - $erase(\lambda x : T.t) = \lambda x.erase(t)$
  - $erase(t_1 t_2) = erase(t_1)erase(t_2)$
- Αν $t \longrightarrow t'$ τότε $erase(t) \longrightarrow erase(t')$
- Αν $erase(t) \longrightarrow m$ τότε υπάρχει όρος t' ώστε $erase(t') = m$ και $t \longrightarrow t'$
- Ο όρος λ-λογισμού χωρίς τύπους m έχει τυπο (is typable) σε ένα λογισμό L με τύπους αν υπάρχει περιβάλλον Γ, όρος t και τύπος T στο L, ώστε:
  - $\Gamma \vdash t : T$
  - $erase(t) = m$

- Απάλειψη τύπων (erasure): σβήσιμο τύπων από τις εκφράσεις
  - $erase(x) = x$
  - $erase(\lambda x : T.t) = \lambda x.erase(t)$
  - $erase(t_1 t_2) = erase(t_1)erase(t_2)$
- Αν $t \longrightarrow t'$ τότε $erase(t) \longrightarrow erase(t')$
- Αν $erase(t) \longrightarrow m$ τότε υπάρχει όρος t' ώστε $erase(t') = m$ και $t \longrightarrow t'$
- Ο όρος λ-λογισμού χωρίς τύπους m έχει τυπο (is typable) σε ένα λογισμό L με τύπους αν υπάρχει περιβάλλον Γ, όρος t και τύπος T στο L, ώστε:
  - $\Gamma \vdash t : T$
  - $erase(t) = m$

- Απάλειψη τύπων (erasure): σβήσιμο τύπων από τις εκφράσεις
  - $erase(x) = x$
  - $erase(\lambda x : T.t) = \lambda x.erase(t)$
  - $erase(t_1 t_2) = erase(t_1)erase(t_2)$
- Αν $t \longrightarrow t'$ τότε $erase(t) \longrightarrow erase(t')$
- Αν $erase(t) \longrightarrow m$ τότε υπάρχει όρος t' ώστε $erase(t') = m$ και $t \longrightarrow t'$
- Ο όρος λ-λογισμού χωρίς τύπους m έχει τυπο (is typable) σε ένα λογισμό L με τύπους αν υπάρχει περιβάλλον Γ, όρος t και τύπος T στο L, ώστε:
  - $\Gamma \vdash t : T$
  - $erase(t) = m$

- Όροι με τύπο:
  - $(\lambda$x.x)true
    - $t =$ (λx:Bool.x)true
    - $T =$ Bool
    - $\Gamma = \emptyset$
  - if iszero x then false else y
    - $t =$ if iszero x then false else y
    - $T =$ Bool
    - $\Gamma = \{(x : \mathtt{Nat}), (y : \mathtt{Bool})\}$
- Όροι χωρίς τύπο:
  - if true then 1 else false
  - Y (δηλ. λf.(λx.f x x)(λx.f x x))

- Όροι με τύπο:
  - `(`λ`x.x)true`
    - $t =$ `(`λ`x:Bool.x)true`
    - $T =$ `Bool`
    - $\Gamma = \emptyset$
  - `if iszero x then false else y`
    - $t =$ `if iszero x then false else y`
    - $T =$ `Bool`
    - $\Gamma = \{(x : \texttt{Nat}), (y : \texttt{Bool})\}$
- Όροι χωρίς τύπο:
  - `if true then 1 else false`
  - Y (δηλ. λ`f.(`λ`x.f x x)(`λ`x.f x x)`)

- Όροι με τύπο:
  - $(\lambda \texttt{x}.\texttt{x})\texttt{true}$
    - $t = (\lambda \texttt{x:Bool.x})\texttt{true}$
    - $T = \texttt{Bool}$
    - $\Gamma = \emptyset$
  - `if iszero x then false else y`
    - $t = \texttt{if iszero x then false else y}$
    - $T = \texttt{Bool}$
    - $\Gamma = \{(\texttt{x} : \texttt{Nat}), (\texttt{y} : \texttt{Bool})\}$
- Όροι χωρίς τύπο:
  - `if true then 1 else false`
  - Y (δηλ. $\lambda \texttt{f}.(\lambda \texttt{x.f x x})(\lambda \texttt{x.f x x})$)

- Όροι με τύπο:
  - $(\lambda \mathtt{x.x})\mathtt{true}$
    - $t = (\lambda \mathtt{x{:}Bool.x})\mathtt{true}$
    - $T = \mathtt{Bool}$
    - $\Gamma = \emptyset$
  - `if iszero x then false else y`
    - $t =$ `if iszero x then false else y`
    - $T = \mathtt{Bool}$
    - $\Gamma = \{(\mathtt{x} : \mathtt{Nat}), (\mathtt{y} : \mathtt{Bool})\}$
- Όροι χωρίς τύπο:
  - `if true then 1 else false`
  - Y (δηλ. $\lambda \mathtt{f.}(\lambda \mathtt{x.f\ x\ x})(\lambda \mathtt{x.f\ x\ x})$)

$$Y = \lambda \mathtt{f}.(\lambda \mathtt{x}.\mathtt{f}\ \mathtt{x}\ \mathtt{x})(\lambda \mathtt{x}.\mathtt{f}\ \mathtt{x}\ \mathtt{x})$$

$$Y' = \lambda \mathtt{f}{:}T_f.\quad (\lambda \mathtt{x}{:}T_{x_1}.\mathtt{f}\ \mathtt{x}\ \mathtt{x})\ (\lambda \mathtt{x}{:}T_{x_2}.\mathtt{f}\ \mathtt{x}\ \mathtt{x})$$

$$T_f = T_{x_1} \longrightarrow T_{x_1} \longrightarrow T_{r_1}$$

$$T_f = T_{x_2} \longrightarrow T_{x_2} \longrightarrow T_{r_2}$$

$$T_{x_1} = T_{x_2} \longrightarrow T_{r_2}$$

$$T_{x_1} = T_{x_2} = T_x \qquad T_{r_1} = T_{r_2} = T_r$$

$$T_x = T_x \longrightarrow T_r$$

Δεν επιτρέπονται άπειροι τύποι στο σύστημα!

$$Y = \lambda \texttt{f.}(\lambda \texttt{x.f x x})(\lambda \texttt{x.f x x})$$

$$Y' = \lambda \texttt{f}:T_f.\quad (\lambda \texttt{x}:T_{x_1}.\texttt{f x x})\ (\lambda \texttt{x}:T_{x_2}.\texttt{f x x})$$

$$T_f = T_{x_1} \longrightarrow T_{x_1} \longrightarrow T_{r_1}$$

$$T_f = T_{x_2} \longrightarrow T_{x_2} \longrightarrow T_{r_2}$$

$$T_{x_1} = T_{x_2} \longrightarrow T_{r_2}$$

$$T_{x_1} = T_{x_2} = T_x \qquad T_{r_1} = T_{r_2} = T_r$$

$$T_x = T_x \longrightarrow T_r$$

Δεν επιτρέπονται άπειροι τύποι στο σύστημα!

$$Y = \lambda \mathtt{f}.(\lambda \mathtt{x}.\mathtt{f\ x\ x})(\lambda \mathtt{x}.\mathtt{f\ x\ x})$$

$$Y' = \lambda \mathtt{f}{:}T_f.\quad (\lambda \mathtt{x}{:}T_{x_1}.\mathtt{f\ x\ x})\ (\lambda \mathtt{x}{:}T_{x_2}.\mathtt{f\ x\ x})$$

$$T_f = T_{x_1} \longrightarrow T_{x_1} \longrightarrow T_{r_1}$$

$$T_f = T_{x_2} \longrightarrow T_{x_2} \longrightarrow T_{r_2}$$

$$T_{x_1} = T_{x_2} \longrightarrow T_{r_2}$$

$$T_{x_1} = T_{x_2} = T_x \qquad T_{r_1} = T_{r_2} = T_r$$

$$T_x = T_x \longrightarrow T_r$$

Δεν επιτρέπονται άπειροι τύποι στο σύστημα!

$$Y = \lambda \mathtt{f}.(\lambda \mathtt{x}.\mathtt{f}\ \mathtt{x}\ \mathtt{x})(\lambda \mathtt{x}.\mathtt{f}\ \mathtt{x}\ \mathtt{x})$$

$$Y' = \lambda \mathtt{f}{:}T_f.\quad (\lambda \mathtt{x}{:}T_{x_1}.\mathtt{f}\ \mathtt{x}\ \mathtt{x})\ (\lambda \mathtt{x}{:}T_{x_2}.\mathtt{f}\ \mathtt{x}\ \mathtt{x})$$

$$T_f = T_{x_1} \longrightarrow T_{x_1} \longrightarrow T_{r_1}$$

$$T_f = T_{x_2} \longrightarrow T_{x_2} \longrightarrow T_{r_2}$$

$$T_{x_1} = T_{x_2} \longrightarrow T_{r_2}$$

$$T_{x_1} = T_{x_2} = T_x \qquad T_{r_1} = T_{r_2} = T_r$$

$$T_x = T_x \longrightarrow T_r$$

Δεν επιτρέπονται άπειροι τύποι στο σύστημα!

$$Y = \lambda \mathtt{f}.(\lambda \mathtt{x}.\mathtt{f}\ \mathtt{x}\ \mathtt{x})(\lambda \mathtt{x}.\mathtt{f}\ \mathtt{x}\ \mathtt{x})$$

$$Y' = \lambda \mathtt{f}{:}T_f.\quad (\lambda \mathtt{x}{:}T_{x_1}.\mathtt{f}\ \mathtt{x}\ \mathtt{x})\ (\lambda \mathtt{x}{:}T_{x_2}.\mathtt{f}\ \mathtt{x}\ \mathtt{x})$$

$$T_f = T_{x_1} \longrightarrow T_{x_1} \longrightarrow T_{r_1}$$

$$T_f = T_{x_2} \longrightarrow T_{x_2} \longrightarrow T_{r_2}$$

$$T_{x_1} = T_{x_2} \longrightarrow T_{r_2}$$

$$T_{x_1} = T_{x_2} = T_x \qquad T_{r_1} = T_{r_2} = T_r$$

$$T_x = T_x \longrightarrow T_r$$

Δεν επιτρέπονται άπειροι τύποι στο σύστημα!

$$Y = \lambda\texttt{f.}(\lambda\texttt{x.f x x})(\lambda\texttt{x.f x x})$$

$$Y' = \lambda\texttt{f:}T_f\texttt{.}\quad(\lambda\texttt{x:}T_{x_1}\texttt{.f x x})\ (\lambda\texttt{x:}T_{x_2}\texttt{.f x x})$$

$$T_f = T_{x_1} \longrightarrow T_{x_1} \longrightarrow T_{r_1}$$

$$T_f = T_{x_2} \longrightarrow T_{x_2} \longrightarrow T_{r_2}$$

$$T_{x_1} = T_{x_2} \longrightarrow T_{r_2}$$

$$T_{x_1} = T_{x_2} = T_x \qquad T_{r_1} = T_{r_2} = T_r$$

$$T_x = T_x \longrightarrow T_r$$

Δεν επιτρέπονται άπειροι τύποι στο σύστημα!

- Ετικέττες (labels) (l)
- Μετασυμβολισμός: $X^{i \in 1..n} = X[i := 1], X[i := 2], .., X[i := n]$
- Νέα Σύνταξη:

$$
\begin{array}{lll}
t ::= & \ldots & \\
 & \{(l{=}t)^{i \in 1..n}\} & \text{εγγραφή} \\
v ::= & \ldots & \\
 & \{(l{=}v)^{i \in 1..n}\} & \\
T ::= & \ldots & \\
 & \{(l : T)^{i \in 1..n}\} & \text{τύπος εγγραφής}
\end{array}
$$

- Ετικέττες (labels) (l)
- Μετασυμβολισμός: $X^{i \in 1..n} = X[i := 1], X[i := 2], .., X[i := n]$
- Νέα Σύνταξη:

$$
\begin{array}{lll}
t ::= & \ldots & \\
 & \{(l{=}t)^{i \in 1..n}\} & \text{εγγραφή} \\
v ::= & \ldots & \\
 & \{(l{=}v)^{i \in 1..n}\} & \\
T ::= & \ldots & \\
 & \{(l : T)^{i \in 1..n}\} & \text{τύπος εγγραφής}
\end{array}
$$

- Ετικέττες (labels) (l)
- Μετασυμβολισμός: $X^{i \in 1..n} = X[i := 1], X[i := 2], .., X[i := n]$
- Νέα Σύνταξη:

$$
\begin{array}{rll}
t ::= & \ldots & \\
& \{(l{=}t)^{i \in 1..n}\} & \text{εγγραφή} \\
v ::= & \ldots & \\
& \{(l{=}v)^{i \in 1..n}\} & \\
T ::= & \ldots & \\
& \{(l : T)^{i \in 1..n}\} & \text{τύπος εγγραφής}
\end{array}
$$

- Νέοι κανόνες αποτίμησης :

$$\frac{}{\{(l_i{=}v_i)^{i\in 1..n}\}\,.\,l_j \;\longrightarrow\; v_j}\,(\text{αποαναφ.}1) \qquad \frac{t \longrightarrow t'}{t.l \;\longrightarrow\; t'.l}\,(\text{αποαναφ.}2)$$

$$\frac{j \in 1..n \qquad t_j \longrightarrow t_j'}{\{(l_i{=}v_i)^{i\in 1..j-1}, (l_i{=}t_i)^{i\in j..n}\} \;\longrightarrow\; \{(l_i{=}v_i)^{i\in 1..j-1}, (l_j{=}t_j'), (l_i{=}t_i)^{i\in j+1..n}\}}\,(\text{εγγρ.})$$

- Νέοι κανόνες τύπων :

$$\frac{\forall i. \quad \Gamma \vdash t_i : T_i}{\Gamma \vdash \{(l_i{=}t_i)^{i\in 1..n}\} : \{(l_i : T_i)^{i\in 1..n}\}}\,(\text{τ.εγγρ.})$$

$$\frac{\Gamma \vdash t : \{(l_i : T_i)^{i\in 1..n}\}}{\Gamma \vdash t.l_j : T_j}\,(\text{τ.αποαναφ.})$$

- Νέοι κανόνες αποτίμησης:

$$\frac{}{\{(l_i{=}v_i)^{i\in 1..n}\}\,.\,l_j \;\longrightarrow\; v_j}(\text{αποαναφ.1}) \qquad \frac{t \longrightarrow t'}{t.l \;\longrightarrow\; t'.l}(\text{αποαναφ.2})$$

$$\frac{j \in 1..n \qquad t_j \longrightarrow t_j'}{\begin{array}{c}\{(l_i{=}v_i)^{i\in 1..j-1}, (l_i{=}t_i)^{i\in j..n}\} \;\longrightarrow\\ \{(l_i{=}v_i)^{i\in 1..j-1}, (l_j{=}t_j'), (l_i{=}t_i)^{i\in j+1..n}\}\end{array}}(\text{εγγρ.})$$

- Νέοι κανόνες τύπων:

$$\frac{\forall i.\quad \Gamma \vdash t_i : T_i}{\Gamma \vdash \{(l_i{=}t_i)^{i\in 1..n}\} : \{(l_i : T_i)^{i\in 1..n}\}}(\text{τ.εγγρ.})$$

$$\frac{\Gamma \vdash t : \{(l_i : T_i)^{i\in 1..n}\}}{\Gamma \vdash t.l_j : T_j}(\text{τ.αποαναφ.})$$

- Η τιμή ενός εναλλακτικού τύπου (variant type) επιλέγεται από ένα πεπερασμένο σύνολο περιπτώσεων
- Κάθε περίπτωση έχει δική της ετικέτα (όπως οι αλγεβρικοί τύποι της Haskell)
- Σύνταξη:

t ::=	...	
	$\texttt{<}l\texttt{=}t\texttt{>}$ $\texttt{as}$ T	κόλλημα ετικέτας
	$\texttt{case}$ t $\texttt{of}$ $(\texttt{<}l\texttt{=}x\texttt{>}$ $\texttt{=>}t)^{i \in 1..n}$	έλεγχος ετικέτας
v ::=	...	
	$\texttt{<}l\texttt{=}v\texttt{>}$ $\texttt{as}$ T	

- Η τιμή ενός εναλλακτικού τύπου (variant type) επιλέγεται από ένα πεπερασμένο σύνολο περιπτώσεων
- Κάθε περίπτωση έχει δική της ετικέτα (όπως οι αλγεβρικοί τύποι της Haskell)
- Σύνταξη:

$$
\begin{array}{lll}
t ::= & \ldots & \\
 & \texttt{<}l\texttt{=}t\texttt{>}\ \texttt{as}\ T & \text{κόλλημα ετικέτας} \\
 & \texttt{case}\ t\ \texttt{of}\ (\texttt{<}l\texttt{=}x\texttt{>}\ \texttt{=>}t)^{i \in 1..n} & \text{έλεγχος ετικέτας} \\
v ::= & \ldots & \\
 & \texttt{<}l\texttt{=}v\texttt{>}\ \texttt{as}\ T &
\end{array}
$$

- Η τιμή ενός <u>εναλλακτικού τύπου (variant type</u>) επιλέγεται από ένα πεπερασμένο σύνολο περιπτώσεων
- Κάθε περίπτωση έχει δική της ετικέτα (όπως οι αλγεβρικοί τύποι της Haskell)
- Σύνταξη:

$$
\begin{array}{lll}
t ::= & \ldots & \\
 & \texttt{<}l\texttt{=}t\texttt{> as } T & \text{κόλλημα ετικέτας} \\
 & \texttt{case } t \texttt{ of } (\texttt{<}l\texttt{=}x\texttt{> =>}t)^{i \in 1..n} & \text{έλεγχος ετικέτας} \\
v ::= & \ldots & \\
 & \texttt{<}l\texttt{=}v\texttt{> as } T &
\end{array}
$$

- Νέοι κανόνες αποτίμησης :

$$\frac{}{\texttt{case}\ \ \texttt{<}l_j\texttt{=}v_j\texttt{>}\ \ \texttt{as}\ \ T\ \ \texttt{of}\ \ (\texttt{<}l_i\texttt{=}x_i\texttt{>=>}t_i)^{i\in 1..n}\ \longrightarrow\ t_j[x_j := v_j]}\ (\text{επιλ.})$$

$$\frac{t \longrightarrow t'}{\texttt{case}\ \ t\ \ \texttt{of}\ \ X\ \longrightarrow\ \texttt{case}\ \ t'\ \ \texttt{of}\ \ X}\ (\text{έλ.ετ.}) \qquad \frac{t \longrightarrow t'}{\texttt{<}l_i\texttt{=}t\texttt{>}\ \ \texttt{as}\ \ T\ \longrightarrow\ \texttt{<}l_i\texttt{=}t'\texttt{>}\ \ \texttt{as}\ \ T}\ (\text{ετ.})$$

- Νέοι κανόνες τύπων :

$$\frac{\Gamma \vdash t_j : T_j}{\Gamma \vdash \texttt{<}l_j\texttt{=}t_j\texttt{>}\ \ \texttt{as}\ \ \texttt{<}(l_i : T_i)^{i\in 1..n}\texttt{>}\ :\ \texttt{<}(l_i : T_i)^{i\in 1..n}\texttt{>}}\ (\text{τ.ετ.})$$

$$\frac{\Gamma \vdash t : \texttt{<}(l_i : T_i)^{i\in 1..n}\texttt{>} \qquad \forall i.\ \Gamma, x_i : T_i \vdash t_i : T}{\Gamma \vdash \texttt{case}\ \ t\ \ \texttt{of}\ \ (\texttt{<}l_i\texttt{=}x_i\texttt{>=>}t_i)^{i\in 1..n} : T}\ (\text{τ.έλ.ετ.})$$

- Νέοι κανόνες αποτίμησης:

$$\frac{}{\texttt{case}\ \ \texttt{<}l_j\texttt{=}v_j\texttt{>}\ \ \texttt{as}\ \ T\ \ \texttt{of}\ \ (\texttt{<}l_i\texttt{=}x_i\texttt{>=>}t_i)^{i\in 1..n} \ \longrightarrow\ t_j[x_j := v_j]}\,(\text{επιλ.})$$

$$\frac{t \longrightarrow t'}{\texttt{case}\ \ t\ \ \texttt{of}\ \ X \longrightarrow \texttt{case}\ \ t'\ \ \texttt{of}\ \ X}\,(\text{έλ.ετ.}) \qquad \frac{t \longrightarrow t'}{\texttt{<}l_i\texttt{=}t\texttt{>}\ \ \texttt{as}\ \ T \longrightarrow \texttt{<}l_i\texttt{=}t'\texttt{>}\ \ \texttt{as}\ \ T}\,(\text{ετ.})$$

- Νέοι κανόνες τύπων:

$$\frac{\Gamma \vdash t_j : T_j}{\Gamma \vdash \texttt{<}l_j\texttt{=}t_j\texttt{>}\ \ \texttt{as}\ \ \texttt{<}(l_i : T_i)^{i\in 1..n}\texttt{>}\ :\ \texttt{<}(l_i : T_i)^{i\in 1..n}\texttt{>}}\,(\text{τ.ετ.})$$

$$\frac{\Gamma \vdash t : \texttt{<}(l_i : T_i)^{i\in 1..n}\texttt{>} \qquad \forall i.\ \Gamma, x_i : T_i \vdash t_i : T}{\Gamma \vdash \texttt{case}\ \ t\ \ \texttt{of}\ \ (\texttt{<}l_i\texttt{=}x_i\texttt{>=>}t_i)^{i\in 1..n} : T}\,(\text{τ.έλ.ετ.})$$

- Το Y δε μπορεί να κατασκευαστεί προγραμματιστικά στη γλώσσα μας, μπορούμε όμως να επεκτείνουμε τη γλώσσα μας
- Θα ονομάζουμε `fix` τον τελεστή αναδρομή Y στην επέκταση
- Νέα σύνταξη:

$$\begin{aligned} t ::= \; & \ldots \\ & \texttt{fix}\ t \quad \text{σταθερό σημείο} \end{aligned}$$

- Νέοι κανόνες αποτίμησης:

$$\frac{}{\texttt{fix}(\lambda x{:}T.t) \longrightarrow t[x := \texttt{fix}(\lambda x{:}T.t)]}\text{(unfold)} \qquad \frac{t \longrightarrow t'}{\texttt{fix}\ t \longrightarrow \texttt{fix}\ t'}\text{(fix)}$$

- Νέος κανόνας τύπων:

$$\frac{\Gamma \vdash t : T \to T}{\Gamma \vdash \texttt{fix}\ t : T}\text{(τ.fix)}$$

- Το Y δε μπορεί να κατασκευαστεί προγραμματιστικά στη γλώσσα μας, μπορούμε όμως να επεκτείνουμε τη γλώσσα μας
- Θα ονομάζουμε `fix` τον τελεστή αναδρομή Y στην επέκταση
- Νέα σύνταξη:

$$t ::= \quad \ldots \qquad \mathtt{fix}\ t \quad \text{σταθερό σημείο}$$

- Νέοι κανόνες αποτίμησης:

$$\frac{}{\mathtt{fix}(\lambda x{:}T.t) \longrightarrow t[x := \mathtt{fix}(\lambda x{:}T.t)]}\text{(unfold)} \qquad \frac{t \longrightarrow t'}{\mathtt{fix}\ t \longrightarrow \mathtt{fix}\ t'}\text{(fix)}$$

- Νέος κανόνας τύπων:

$$\frac{\Gamma \vdash t : T \to T}{\Gamma \vdash \mathtt{fix}\ t : T}\text{(T.fix)}$$

- Το Y δε μπορεί να κατασκευαστεί προγραμματιστικά στη γλώσσα μας, μπορούμε όμως να επεκτείνουμε τη γλώσσα μας
- Θα ονομάζουμε `fix` τον τελεστή αναδρομή Y στην επέκταση
- Νέα σύνταξη:

$$\begin{array}{rll} t ::= & \ldots & \\ & \texttt{fix}\ t & \text{σταθερό σημείο} \end{array}$$

- Νέοι κανόνες αποτίμησης:

$$\frac{}{\texttt{fix}(\lambda x{:}T.t) \longrightarrow t[x := \texttt{fix}(\lambda x{:}T.t)]}\text{(unfold)} \qquad \frac{t \longrightarrow t'}{\texttt{fix}\ t \longrightarrow \texttt{fix}\ t'}\text{(fix)}$$

- Νέος κανόνας τύπων:

$$\frac{\Gamma \vdash t : T \rightarrow T}{\Gamma \vdash \texttt{fix}\ t : T}\text{(T.fix)}$$

- Το Y δε μπορεί να κατασκευαστεί προγραμματιστικά στη γλώσσα μας, μπορούμε όμως να επεκτείνουμε τη γλώσσα μας
- Θα ονομάζουμε `fix` τον τελεστή αναδρομή Y στην επέκταση
- Νέα σύνταξη:

$$t ::= \quad \ldots \qquad \texttt{fix}\ t \quad \text{σταθερό σημείο}$$

- Νέοι κανόνες αποτίμησης:

$$\frac{}{\texttt{fix}(\lambda x{:}T.t) \longrightarrow t[x := \texttt{fix}(\lambda x{:}T.t)]}\text{(unfold)} \qquad \frac{t \longrightarrow t'}{\texttt{fix}\ t \longrightarrow \texttt{fix}\ t'}\text{(fix)}$$

- Νέος κανόνας τύπων:

$$\frac{\Gamma \vdash t : T \to T}{\Gamma \vdash \texttt{fix}\ t : T}\text{(T.fix)}$$

- Το Y δε μπορεί να κατασκευαστεί προγραμματιστικά στη γλώσσα μας, μπορούμε όμως να επεκτείνουμε τη γλώσσα μας
- Θα ονομάζουμε `fix` τον τελεστή αναδρομή Y στην επέκταση
- Νέα σύνταξη:

$$
\begin{array}{rll}
t ::= & \ldots & \\
 & \texttt{fix}\ t & \text{σταθερό σημείο}
\end{array}
$$

- Νέοι κανόνες αποτίμησης:

$$
\frac{}{\texttt{fix}(\lambda x\texttt{:}T\texttt{.}t) \longrightarrow t[x := \texttt{fix}(\lambda x\texttt{:}T\texttt{.}t)]}\text{(unfold)}
\qquad
\frac{t \longrightarrow t'}{\texttt{fix}\ t \longrightarrow \texttt{fix}\ t'}\text{(fix)}
$$

- Νέος κανόνας τύπων:

$$
\frac{\Gamma \vdash t : T \rightarrow T}{\Gamma \vdash \texttt{fix}\ t : T}\text{(T.fix)}
$$

```
factorial = fix factdef

   factdef
=  λf:Nat→Nat.λn:Nat.
      if iszero n then 1 else n*f(n-1)
```

Τύπος:

$$\dfrac{\dfrac{\dfrac{\cdots}{n : \mathtt{Nat}, f : \mathtt{Nat}\to\mathtt{Nat} \vdash \texttt{if iszero n then 1 else n*f(n-1)} : \mathtt{Nat}}}{\vdash \texttt{factdef} : (\mathtt{Nat}\to\mathtt{Nat})\to(\mathtt{Nat}\to\mathtt{Nat})}\,(\text{τ.}\lambda\text{-αφ.})}{\vdash \texttt{factorial} : \mathtt{Nat}\to\mathtt{Nat}}\,(\text{τ.fix})$$

```
factorial = fix factdef
```

```
   factdef
=  λf:Nat→Nat.λn:Nat.
      if iszero n then 1 else n*f(n-1)
```

Τύπος:

$$\dfrac{\dfrac{\dfrac{\cdots}{\begin{array}{l}\mathtt{n} : \mathtt{Nat}, \mathtt{f} : \mathtt{Nat}{\rightarrow}\mathtt{Nat} \vdash \\ \quad \mathtt{if\ iszero\ n\ then\ 1\ else\ n*f(n-1)} : \mathtt{Nat}\end{array}}}{\vdash \mathtt{factdef} : (\mathtt{Nat}{\rightarrow}\mathtt{Nat}){\rightarrow}(\mathtt{Nat}{\rightarrow}\mathtt{Nat})}\,(\text{τ.}\lambda\text{-αφ.})}{\vdash \mathtt{factorial} : \mathtt{Nat}{\rightarrow}\mathtt{Nat}}\,(\text{τ.fix})$$

```
     factorial 1
-->  ( (λn:Nat.if iszero n then 1 else n*f(n-1))
       [f := factorial]) 1
=    (λn:Nat.
       if iszero n then 1 else n*factorial(n-1))1
-->  if iszero 1 then 1 else 1*factorial(1-1)
-->* 1*factorial 0
-->  1*(if iszero 0 then 1 else 0*factorial(0-1))
-->* 1*1
-->* 1
```

```
    factorial 1
⟶  ( (λn:Nat.if iszero n then 1 else n*f(n-1))
     [f := factorial]) 1
=   (λn:Nat.
      if iszero n then 1 else n*factorial(n-1))1
⟶  if iszero 1 then 1 else 1*factorial(1-1)
⟶* 1*factorial 0
⟶  1*(if iszero 0 then 1 else 0*factorial(0-1))
⟶* 1*1
⟶* 1
```

$$
\begin{array}{ll}
 & \texttt{factorial 1} \\
\longrightarrow & (\ \ (\lambda\texttt{n:Nat.if iszero n then 1 else n}*\texttt{f(n-1)}) \\
 & \quad [\texttt{f} := \texttt{factorial}])\ \texttt{1} \\
= & (\lambda\texttt{n:Nat.} \\
 & \quad \texttt{if iszero n then 1 else n}*\texttt{factorial(n-1))1} \\
\longrightarrow & \texttt{if iszero 1 then 1 else 1}*\texttt{factorial(1-1)} \\
\longrightarrow^* & \texttt{1}*\texttt{factorial 0} \\
\longrightarrow & \texttt{1}*\texttt{(if iszero 0 then 1 else 0}*\texttt{factorial(0-1))} \\
\longrightarrow^* & \texttt{1}*\texttt{1} \\
\longrightarrow^* & \texttt{1}
\end{array}
$$

```
      factorial 1
⟶     (  (λn:Nat.if iszero n then 1 else n*f(n-1))
         [f := factorial]) 1
=     (λn:Nat.
        if iszero n then 1 else n*factorial(n-1))1
⟶     if iszero 1 then 1 else 1*factorial(1-1)
⟶*    1*factorial 0
⟶     1*(if iszero 0 then 1 else 0*factorial(0-1))
⟶*    1*1
⟶*    1
```

```
        factorial 1
⟶       (  (λn:Nat.if iszero n then 1 else n*f(n-1))
           [f := factorial]) 1
=       (λn:Nat.
          if iszero n then 1 else n*factorial(n-1))1
⟶       if iszero 1 then 1 else 1*factorial(1-1)
⟶*      1*factorial 0
⟶       1*(if iszero 0 then 1 else 0*factorial(0-1))
⟶*      1*1
⟶*      1
```

$$
\begin{array}{ll}
 & \texttt{factorial 1} \\
\longrightarrow & \texttt{(}(\lambda\texttt{n:Nat.if iszero n then 1 else n}*\texttt{f(n-1))} \\
 & \quad [\texttt{f} := \texttt{factorial}]\texttt{) 1} \\
= & (\lambda\texttt{n:Nat.} \\
 & \quad \texttt{if iszero n then 1 else n}*\texttt{factorial(n-1))1} \\
\longrightarrow & \texttt{if iszero 1 then 1 else 1}*\texttt{factorial(1-1)} \\
\longrightarrow^* & \texttt{1}*\texttt{factorial 0} \\
\longrightarrow & \texttt{1}*\texttt{(if iszero 0 then 1 else 0}*\texttt{factorial(0-1))} \\
\longrightarrow^* & \texttt{1}*\texttt{1} \\
\longrightarrow^* & \texttt{1}
\end{array}
$$

$$
\begin{array}{ll}
 & \texttt{factorial 1} \\
\longrightarrow & \texttt{(} (\lambda\texttt{n:Nat.if iszero n then 1 else n}*\texttt{f(n-1))} \\
 & \quad [\texttt{f} := \texttt{factorial}]\texttt{) 1} \\
= & \texttt{(}\lambda\texttt{n:Nat.} \\
 & \quad \texttt{if iszero n then 1 else n}*\texttt{factorial(n-1))1} \\
\longrightarrow & \texttt{if iszero 1 then 1 else 1}*\texttt{factorial(1-1)} \\
\longrightarrow^* & \texttt{1}*\texttt{factorial 0} \\
\longrightarrow & \texttt{1}*\texttt{(if iszero 0 then 1 else 0}*\texttt{factorial(0-1))} \\
\longrightarrow^* & \texttt{1}*\texttt{1} \\
\longrightarrow^* & \texttt{1}
\end{array}
$$

$$
\begin{array}{ll}
 & \texttt{factorial 1} \\
\longrightarrow & \texttt{(}(\lambda\texttt{n:Nat.if iszero n then 1 else n}*\texttt{f(n-1))} \\
 & \quad [\texttt{f} := \texttt{factorial}]\texttt{) 1} \\
= & (\lambda\texttt{n:Nat.} \\
 & \quad \texttt{if iszero n then 1 else n}*\texttt{factorial(n-1))1} \\
\longrightarrow & \texttt{if iszero 1 then 1 else 1}*\texttt{factorial(1-1)} \\
\longrightarrow^* & \texttt{1}*\texttt{factorial 0} \\
\longrightarrow & \texttt{1}*\texttt{(if iszero 0 then 1 else 0}*\texttt{factorial(0-1))} \\
\longrightarrow^* & \texttt{1}*\texttt{1} \\
\longrightarrow^* & \texttt{1}
\end{array}
$$

Subtyping

- Το απλό σύστημα τύπων είναι πολύ αυστηρό:

$$(\lambda \texttt{r:\{x:Nat\}. r.x})\texttt{\{x=0,y=1\}}$$

το όρισμα πρέπει να είναι ακριβώς τύπου `{x:Nat}`:

$$\frac{\Gamma \vdash f : T_1 \rightarrow T_2 \quad \Gamma \vdash x : T_1}{\Gamma \vdash f\,x : T_2}(\text{τ.εφ.})$$

- Θέλουμε η έκφραση να επιτρέπεται: οποιαδήποτε εγγραφή που να περιέχει την ετικέτα x με τύπο `Nat` πρέπει να γίνεται δεκτή
- Σχέση συμπερίληψης στους τύπους: ο τύπος `{x:Nat}` θα πρέπει να περιέχει τις τιμές του τύπου `{x:Nat,y:Nat}`
- Subtyping: σχέση συμπερίληψης στους τύπους. Ένας τύπος S είναι υποτύπος (subtype) ενός άλλου τύπου T όταν οι τιμές του S είναι και τιμές του T. Ο T είναι υπερτύπος (supertype) του S.
- 'S υποτύπος του T' γράφεται $S <: T$

- Το απλό σύστημα τύπων είναι πολύ αυστηρό:

$$(\lambda\texttt{r:\{x:Nat\}.}\quad \texttt{r.x)\{x=0,y=1\}}$$

το όρισμα πρέπει να είναι ακριβώς τύπου `{x:Nat}`:

$$\frac{\Gamma \vdash f : T_1 \to T_2 \quad \Gamma \vdash x : T_1}{\Gamma \vdash f\ x : T_2}(\text{τ.εφ.})$$

- Θέλουμε η έκφραση να επιτρέπεται: οποιαδήποτε εγγραφή που να περιέχει την ετικέττα x με τύπο `Nat` πρέπει να γίνεται δεκτή
- Σχέση συμπερίληψης στους τύπους: ο τύπος `{x:Nat}` θα πρέπει να περιέχει τις τιμές του τύπου `{x:Nat,y:Nat}`
- Subtyping: σχέση συμπερίληψης στους τύπους. Ένας τύπος S είναι υποτύπος (subtype) ενός άλλου τύπου T όταν οι τιμές του S είναι και τιμές του T. Ο T είναι υπερτύπος (supertype) του S.
- 'S υποτύπος του T' γράφεται $S <: T$

Subtyping

- Το απλό σύστημα τύπων είναι πολύ αυστηρό:

$$(\lambda\texttt{r:\{x:Nat\}.}\quad \texttt{r.x)\{x=0,y=1\}}$$

το όρισμα πρέπει να είναι ακριβώς τύπου `{x:Nat}`:

$$\frac{\Gamma \vdash f : T_1 \rightarrow T_2 \quad \Gamma \vdash x : T_1}{\Gamma \vdash f\ x : T_2}(\text{τ.εφ.})$$

- Θέλουμε η έκφραση να επιτρέπεται: οποιαδήποτε εγγραφή που να περιέχει την ετικέττα x με τύπο `Nat` πρέπει να γίνεται δεκτή
- Σχέση συμπερίληψης στους τύπους: ο τύπος `{x:Nat}` θα πρέπει να περιέχει τις τιμές του τύπου `{x:Nat,y:Nat}`
- Subtyping: σχέση συμπερίληψης στους τύπους. Ένας τύπος S είναι υποτύπος (subtype) ενός άλλου τύπου T όταν οι τιμές του S είναι και τιμές του T. Ο T είναι υπερτύπος (supertype) του S.
- 'S υποτύπος του T' γράφεται $S <: T$

- Το απλό σύστημα τύπων είναι πολύ αυστηρό:

$$\texttt{(}\lambda\texttt{r:\{x:Nat\}. r.x)\{x=0,y=1\}}$$

το όρισμα πρέπει να είναι ακριβώς τύπου `{x:Nat}`:

$$\frac{\Gamma \vdash f : T_1 \to T_2 \quad \Gamma \vdash x : T_1}{\Gamma \vdash f\ x : T_2}(\text{τ.εφ.})$$

- Θέλουμε η έκφραση να επιτρέπεται: οποιαδήποτε εγγραφή που να περιέχει την ετικέττα x με τύπο `Nat` πρέπει να γίνεται δεκτή
- Σχέση συμπερίληψης στους τύπους: ο τύπος `{x:Nat}` θα πρέπει να περιέχει τις τιμές του τύπου `{x:Nat,y:Nat}`
- Subtyping: σχέση συμπερίληψης στους τύπους. Ένας τύπος S είναι υποτύπος (subtype) ενός άλλου τύπου T όταν οι τιμές του S είναι και τιμές του T. Ο T είναι υπερτύπος (supertype) του S.
- 'S υποτύπος του T' γράφεται $S <: T$

Subtyping

- Το απλό σύστημα τύπων είναι πολύ αυστηρό:

$$(\lambda \texttt{r:\{x:Nat\}.}\ \ \texttt{r.x)\{x=0,y=1\}}$$

το όρισμα πρέπει να είναι ακριβώς τύπου `{x:Nat}`:

$$\frac{\Gamma \vdash f : T_1 \to T_2 \quad \Gamma \vdash x : T_1}{\Gamma \vdash f\ x : T_2}(\text{τ.εφ.})$$

- Θέλουμε η έκφραση να επιτρέπεται: οποιαδήποτε εγγραφή που να περιέχει την ετικέττα x με τύπο `Nat` πρέπει να γίνεται δεκτή
- Σχέση συμπερίληψης στους τύπους: ο τύπος `{x:Nat}` θα πρέπει να περιέχει τις τιμές του τύπου `{x:Nat,y:Nat}`
- Subtyping: σχέση συμπερίληψης στους τύπους. Ένας τύπος S είναι υποτύπος (subtype) ενός άλλου τύπου T όταν οι τιμές του S είναι και τιμές του T. Ο T είναι υπερτύπος (supertype) του S.
- 'S υποτύπος του T' γράφεται $S <: T$

- Υπαγωγή (subsumption)

$$\frac{\Gamma \vdash t : S \qquad S <: T}{\Gamma \vdash t : T}(\text{υπαγ.})$$

- Ανακλαστικότητα

$$\frac{}{S <: S}(\text{ανακλ.})$$

- Μεταβατικότητα

$$\frac{S <: U \qquad U <: T}{S <: T}(\text{μεταβ.})$$

- Υπαγωγή (subsumption)

$$\frac{\Gamma \vdash t : S \qquad S <: T}{\Gamma \vdash t : T}(\text{υπαγ.})$$

- Ανακλαστικότητα

$$\frac{}{S <: S}(\text{ανακλ.})$$

- Μεταβατικότητα

$$\frac{S <: U \qquad U <: T}{S <: T}(\text{μεταβ.})$$

- Υπαγωγή (subsumption)

$$\frac{\Gamma \vdash t : S \qquad S <: T}{\Gamma \vdash t : T}(\text{υπαγ.})$$

- Ανακλαστικότητα

$$\frac{}{S <: S}(\text{ανακλ.})$$

- Μεταβατικότητα

$$\frac{S <: U \qquad U <: T}{S <: T}(\text{μεταβ.})$$

- Αντι-μονοτονική σχέση πλάτους :

$$\frac{}{\{(l_i : T_i)^{i \in 1..n+k}\} <: \{(l_i : T_i)^{i \in 1..n}\}}(\text{εγγ.πλ.})$$

- Μονοτονική σχέση βάθους :

$$\frac{\forall i. \quad S_i <: T_i}{\{(l_i : S_i)^{i \in 1..n}\} <: \{(l_i : T_i)^{i \in 1..n}\}}(\text{εγγ.βάθ.})$$

- Αντιμετάθεση ετικεττών :

$$\frac{X \text{ αντιμετάθεση του } Y}{\{X\} <: \{Y\}}(\text{εγγ.αντιμετ.})$$

- Αντι-μονοτονική σχέση πλάτους:

$$\frac{}{\{(l_i : T_i)^{i\in 1..n+k}\} <: \{(l_i : T_i)^{i\in 1..n}\}}\text{(εγγ.πλ.)}$$

- Μονοτονική σχέση βάθους:

$$\frac{\forall i.\quad S_i <: T_i}{\{(l_i : S_i)^{i\in 1..n}\} <: \{(l_i : T_i)^{i\in 1..n}\}}\text{(εγγ.βάθ.)}$$

- Αντιμετάθεση ετικεττών:

$$\frac{X \text{ αντιμετάθεση του } Y}{\{X\} <: \{Y\}}\text{(εγγ.αντιμετ.)}$$

- Αντι-μονοτονική σχέση πλάτους :

$$\frac{}{\{(l_i : T_i)^{i \in 1..n+k}\} <: \{(l_i : T_i)^{i \in 1..n}\}}(\text{εγγ.πλ.})$$

- Μονοτονική σχέση βάθους :

$$\frac{\forall i. \quad S_i <: T_i}{\{(l_i : S_i)^{i \in 1..n}\} <: \{(l_i : T_i)^{i \in 1..n}\}}(\text{εγγ.βάθ.})$$

- Αντιμετάθεση ετικεττών :

$$\frac{X \text{ αντιμετάθεση του } Y}{\{X\} <: \{Y\}}(\text{εγγ.αντιμετ.})$$

$$\dfrac{\dfrac{\dfrac{}{\texttt{\{y:Nat\}} <: \texttt{\{\}}}\,(\text{εγγ.πλ.})}{\texttt{\{x:\{y:Nat\}\}} <: \texttt{\{x:\{\}\}}}\,(\text{εγγ.βάθ.})}{\texttt{\{x:\{y:Nat\},z:Nat\}} <: \texttt{\{x:\{\}\}}}\,(\text{εγγ.πλ.})$$

Έστω:

$$\texttt{rec=\{x=0,y=1\}} \qquad \texttt{f} = \lambda\texttt{r:\{x:Nat\}.}\ \ \texttt{r.x}$$

$$\texttt{T=\{x:Nat\}} \qquad \texttt{S=\{x:Nat,y:Nat\}}$$

Τότε:

$$\dfrac{\dfrac{\dfrac{\dfrac{}{\vdash \texttt{0:Nat}} \quad \dfrac{\cdots}{\vdash \texttt{1:Nat}}}{\vdash \texttt{rec} : \texttt{S}}\,(\text{τ.εγγρ.}) \qquad \dfrac{}{\texttt{S} <: \texttt{T}}\,(\text{εγγ.πλ.})}{\vdash \texttt{rec} : \texttt{T}}\,(\text{υπαγ.}) \qquad \dfrac{\cdots}{\vdash \texttt{f} : \texttt{T} \to \texttt{Nat}}}{\vdash \texttt{f}\ \ \texttt{rec} : \texttt{Nat}}\,(\text{τ.εφαρμ.})$$

$$\dfrac{\dfrac{\dfrac{}{\texttt{\{y:Nat\}} <: \texttt{\{\}}}\;(\text{εγγ.πλ.})}{\texttt{\{x:\{y:Nat\}\}} <: \texttt{\{x:\{\}\}}}\;(\text{εγγ.βάθ.})}{\texttt{\{x:\{y:Nat\},z:Nat\}} <: \texttt{\{x:\{\}\}}}\;(\text{εγγ.πλ.})$$

Έστω:

$$\texttt{rec=\{x=0,y=1\}} \quad \texttt{f} = \lambda \texttt{r:\{x:Nat\}.} \quad \texttt{r.x}$$

$$\texttt{T=\{x:Nat\}} \quad \texttt{S=\{x:Nat,y:Nat\}}$$

Τότε:

$$\dfrac{\dfrac{\dfrac{}{\vdash \texttt{0:Nat}} \quad \dfrac{\cdots}{\vdash \texttt{1:Nat}}}{\vdash \texttt{rec} : \texttt{S}}\;(\text{τ.εγγρ.}) \qquad \dfrac{}{\texttt{S} <: \texttt{T}}\;(\text{εγγ.πλ.})}{\vdash \texttt{rec} : \texttt{T}}\;(\text{υπαγ.}) \qquad \dfrac{\cdots}{\vdash \texttt{f} : \texttt{T} \to \texttt{Nat}}$$

$$\dfrac{\vdash \texttt{rec} : \texttt{T} \qquad \vdash \texttt{f} : \texttt{T} \to \texttt{Nat}}{\vdash \texttt{f rec} : \texttt{Nat}}\;(\text{τ.εφαρμ.})$$

Subtyping

Εγγραφές - Παράδειγμα

$$\dfrac{\dfrac{\dfrac{}{\texttt{\{y:Nat\}} <: \texttt{\{\}}}\,(\text{εγγ.πλ.})}{\texttt{\{x:\{y:Nat\}\}} <: \texttt{\{x:\{\}\}}}\,(\text{εγγ.βάθ.})}{\texttt{\{x:\{y:Nat\},z:Nat\}} <: \texttt{\{x:\{\}\}}}\,(\text{εγγ.πλ.})$$

Έστω:

$$\texttt{rec=\{x=0,y=1\}} \quad \texttt{f} = \lambda \texttt{r:\{x:Nat\}.}\ \ \texttt{r.x}$$

$$\texttt{T=\{x:Nat\}} \quad \texttt{S=\{x:Nat,y:Nat\}}$$

Τότε:

$$\dfrac{\dfrac{\dfrac{\dfrac{}{\vdash \texttt{0:Nat}} \quad \dfrac{\cdots}{\vdash \texttt{1:Nat}}}{\vdash \texttt{rec} : \texttt{S}}\,(\text{τ.εγγρ.}) \qquad \dfrac{}{\texttt{S} <: \texttt{T}}\,(\text{εγγ.πλ.})}{\vdash \texttt{rec} : \texttt{T}}\,(\text{υπαγ.}) \qquad \dfrac{\cdots}{\vdash \texttt{f} : \texttt{T} \rightarrow \texttt{Nat}}}{\vdash \texttt{f}\ \ \texttt{rec} : \texttt{Nat}}\,(\text{τ.εφαρμ.})$$

- Όταν έχουμε subtyping, δε χρειαζόμαστε το κομμάτι as στους εναλλακτικούς τύπους.
- Νέα Σύνταξη:

$$
\begin{array}{lll}
t ::= & \ldots & \\
 & <l{=}t>\ \cancel{\mathtt{as}\ T} & \text{κόλλημα ετικέτας} \\
v ::= & \ldots & \\
 & <l{=}v>\ \cancel{\mathtt{as}\ T} &
\end{array}
$$

- Νέος κανόνας τύπων:

$$
\frac{\Gamma \vdash t : T}{<l{=}t> \,:\, <l : T>}\,(\text{τ.ετ.})
$$

- Όταν έχουμε subtyping, δε χρειαζόμαστε το κομμάτι as στους εναλλακτικούς τύπους.
- Νέα Σύνταξη:

$$
\begin{aligned}
t ::= \quad & \ldots \\
& \langle l{=}t\rangle \cancel{\ \mathtt{as}\ T} \qquad \text{κόλλημα ετικέτας} \\
v ::= \quad & \ldots \\
& \langle l{=}v\rangle \cancel{\ \mathtt{as}\ T}
\end{aligned}
$$

- Νέος κανόνας τύπων:

$$\frac{\Gamma \vdash t : T}{\langle l{=}t\rangle : \langle l : T\rangle}\ (\text{τ.ετ.})$$

- Όταν έχουμε subtyping, δε χρειαζόμαστε το κομμάτι as στους εναλλακτικούς τύπους.
- Νέα Σύνταξη:

$$
\begin{array}{lll}
t ::= & \ldots & \\
 & <l{=}t>\cancel{\ \texttt{as}\ T} & \text{κόλλημα ετικέτας} \\
v ::= & \ldots & \\
 & <l{=}v>\cancel{\ \texttt{as}\ T} &
\end{array}
$$

- Νέος κανόνας τύπων:

$$\frac{\Gamma \vdash t : T}{<l{=}t> : <l : T>} \text{(τ.ετ.)}$$

- Μονοτονική σχέση πλάτους:

$$\frac{}{<(l_i : T_i)^{i \in 1..n}> \; <: \; <(l_i : T_i)^{i \in 1..n+k}>} \text{(εναλ.πλ.)}$$

- Μονοτονική σχέση βάθους:

$$\frac{\forall i. \quad S_i <: T_i}{<(l_i : S_i)^{i \in 1..n}> \; <: \; <(l_i : T_i)^{i \in 1..n}>} \text{(εναλ.βάθ.)}$$

- Αντιμετάθεση ετικεττών:

$$\frac{X \text{ αντιμετάθεση του } Y}{<X> \; <: \; <Y>} \text{(εναλ.αντιμετ.)}$$

- Μονοτονική σχέση πλάτους:

$$\frac{}{<(l_i : T_i)^{i \in 1..n}> \;<:\; <(l_i : T_i)^{i \in 1..n+k}>}\,(\text{εναλ.πλ.})$$

- Μονοτονική σχέση βάθους:

$$\frac{\forall i. \quad S_i <: T_i}{<(l_i : S_i)^{i \in 1..n}> \;<:\; <(l_i : T_i)^{i \in 1..n}>}\,(\text{εναλ.βάθ.})$$

- Αντιμετάθεση ετικεττών:

$$\frac{X \text{ αντιμετάθεση του } Y}{<X> \;<:\; <Y>}\,(\text{εναλ.αντιμετ.})$$

- Μονοτονική σχέση πλάτους:

$$\frac{}{<(l_i : T_i)^{i\in 1..n}> \; <: \; <(l_i : T_i)^{i\in 1..n+k}>}(\text{εναλ.πλ.})$$

- Μονοτονική σχέση βάθους:

$$\frac{\forall i. \quad S_i <: T_i}{<(l_i : S_i)^{i\in 1..n}> \; <: \; <(l_i : T_i)^{i\in 1..n}>}(\text{εναλ.βάθ.})$$

- Αντιμετάθεση ετικεττών:

$$\frac{X \text{ αντιμετάθεση του } Y}{<X> \; <: \; <Y>}(\text{εναλ.αντιμετ.})$$

- Αντι-μονοτονικός / μονοτονικός κανόνας συναρτήσεων:

$$\frac{T_1 <: S_1 \qquad S_2 <: T_2}{S_1 \rightarrow S_2 \; <: \; T_1 \rightarrow T_2}(\text{υποτ.συν.})$$

- Παράδειγμα:

```
h=λg:{x:Nat}→Nat.g{x=2}+1
```

με το f του προηγούμενου παραδείγματος:

$$\texttt{h f} \longrightarrow \texttt{f\{x=2\}+1} \longrightarrow^* \texttt{3}$$

με `λr:{}.4`:

$$\texttt{h(}\lambda\texttt{r:\{\}.4)} \longrightarrow \texttt{(}\lambda\texttt{r:\{\}.4)\{x=2\}+1} \longrightarrow^* \texttt{5}$$

- Κορυφαίος τύπος Top: $\dfrac{}{S <: \texttt{Top}}(\text{top})$

- Αντι-μονοτονικός / μονοτονικός κανόνας συναρτήσεων:

$$\frac{T_1 <: S_1 \qquad S_2 <: T_2}{S_1 \to S_2 <: T_1 \to T_2}\text{(υποτ.συν.)}$$

- Παράδειγμα:

$$\texttt{h=}\lambda\texttt{g:\{x:Nat\}}\to\texttt{Nat.g\{x=2\}+1}$$

με το f του προηγούμενου παραδείγματος:

$$\texttt{h f} \longrightarrow \texttt{f\{x=2\}+1} \longrightarrow^* \texttt{3}$$

με λr:{}.4:

$$\texttt{h(}\lambda\texttt{r:\{\}.4)} \longrightarrow \texttt{(}\lambda\texttt{r:\{\}.4)\{x=2\}+1} \longrightarrow^* \texttt{5}$$

- Κορυφαίος τύπος Top: $\dfrac{}{S <: \texttt{Top}}$(top)

- Αντι-μονοτονικός / μονοτονικός κανόνας συναρτήσεων:

$$\frac{T_1 <: S_1 \qquad S_2 <: T_2}{S_1 \rightarrow S_2 \; <: \; T_1 \rightarrow T_2}(\text{υποτ.συν.})$$

- Παράδειγμα:

$$\texttt{h=}\lambda\texttt{g:\{x:Nat\}}\rightarrow\texttt{Nat.g\{x=2\}+1}$$

με το f του προηγούμενου παραδείγματος:

$$\texttt{h f} \longrightarrow \texttt{f\{x=2\}+1} \longrightarrow^* \texttt{3}$$

με λr:{}.4:

$$\texttt{h(}\lambda\texttt{r:\{\}.4)} \longrightarrow \texttt{(}\lambda\texttt{r:\{\}.4)\{x=2\}+1} \longrightarrow^* \texttt{5}$$

- Κορυφαίος τύπος Top: $\dfrac{}{S <: \texttt{Top}}(\text{top})$

- Αντι-μονοτονικός / μονοτονικός κανόνας συναρτήσεων:

$$\frac{T_1 <: S_1 \qquad S_2 <: T_2}{S_1 \rightarrow S_2 <: T_1 \rightarrow T_2}\text{(υποτ.συν.)}$$

- Παράδειγμα:

$$\texttt{h=}\lambda\texttt{g:\{x:Nat\}}\rightarrow\texttt{Nat.g\{x=2\}+1}$$

με το f του προηγούμενου παραδείγματος:

$$\texttt{h f} \longrightarrow \texttt{f\{x=2\}+1} \longrightarrow^* \texttt{3}$$

με λr:{}.4:

$$\texttt{h(}\lambda\texttt{r:\{\}.4)} \longrightarrow \texttt{(}\lambda\texttt{r:\{\}.4)\{x=2\}+1} \longrightarrow^* \texttt{5}$$

- Κορυφαίος τύπος Top: $\frac{}{S <: \texttt{Top}}\text{(top)}$

- Αντι-μονοτονικός / μονοτονικός κανόνας συναρτήσεων:

$$\frac{T_1 <: S_1 \qquad S_2 <: T_2}{S_1 \to S_2 \ <: \ T_1 \to T_2}\text{(υποτ.συν.)}$$

- Παράδειγμα:

$$\texttt{h=}\lambda\texttt{g:\{x:Nat\}}\to\texttt{Nat.g\{x=2\}+1}$$

με το `f` του προηγούμενου παραδείγματος:

$$\texttt{h f} \longrightarrow \texttt{f\{x=2\}+1} \longrightarrow^{*} 3$$

με λ`r:{}.4`:

$$\texttt{h(}\lambda\texttt{r:\{\}.4)} \longrightarrow (\lambda\texttt{r:\{\}.4)\{x=2\}+1} \longrightarrow^{*} 5$$

- Κορυφαίος τύπος `Top`: $\dfrac{}{S <: \texttt{Top}}\text{(top)}$

Επισκόπηση - Ι

- Συστήματα τύπων: στατικές μη πλήρεις αποδείξεις ορθότητας που προσεγγίζουν τον υπολογισμό
    - εντοπίζουν (συντηρητικά) λανθασμένες χρήσεις τιμών
    - χρήσιμα για: εντοπισμό σφαλμάτων, αφαίρεση, τεκμηρίωση, ασφάλεια
    - εκφραστικότητα: πόσες ορθές εκφράσεις επιτρέπονται
- λ-λογισμός χωρίς τύπους + φυσικοί + αληθοτιμές
    - λάθος τύπων $\leftrightarrow$ η αποτίμηση 'κολλάει'
- Επιθυμητές ιδιότητες συστήματος τύπων
    - ασφάλεια (η αποτίμηση διατηρεί την ορθότητα)
    - πρόοδος (η αποτίμηση ορθών εκφράσεων δεν κολλάει)
- Μελέτη τύπων μέσω του ορισμού σχέσης: $\Gamma \vdash t : T$
- Απάλειψη $\longrightarrow$ πότε μία έκφραση χωρίς τύπους μπορεί να πάρει τύπο σε ένα σύστημα

Επισκόπηση - I

- Συστήματα τύπων: στατικές μη πλήρεις αποδείξεις ορθότητας που προσεγγίζουν τον υπολογισμό
  - εντοπίζουν (συντηρητικά) λανθασμένες χρήσεις τιμών
  - χρήσιμα για: εντοπισμό σφαλμάτων, αφαίρεση, τεκμηρίωση, ασφάλεια
  - εκφραστικότητα: πόσες ορθές εκφράσεις επιτρέπονται
- λ-λογισμός χωρίς τύπους + φυσικοί + αληθοτιμές
  - λάθος τύπων $\leftrightarrow$ η αποτίμηση 'κολλάει'
- Επιθυμητές ιδιότητες συστήματος τύπων
  - ασφάλεια (η αποτίμηση διατηρεί την ορθότητα)
  - πρόοδος (η αποτίμηση ορθών εκφράσεων δεν κολλάει)
- Μελέτη τύπων μέσω του ορισμού σχέσης: $\Gamma \vdash t : T$
- Απάλειψη $\longrightarrow$ πότε μία έκφραση χωρίς τύπους μπορεί να πάρει τύπο σε ένα σύστημα

Επισκόπηση - Ι

- Συστήματα τύπων: στατικές μη πλήρεις αποδείξεις ορθότητας που προσεγγίζουν τον υπολογισμό
  - εντοπίζουν (συντηρητικά) λανθασμένες χρήσεις τιμών
  - χρήσιμα για: εντοπισμό σφαλμάτων, αφαίρεση, τεκμηρίωση, ασφάλεια
  - εκφραστικότητα: πόσες ορθές εκφράσεις επιτρέπονται
- λ-λογισμός χωρίς τύπους + φυσικοί + αληθοτιμές
  - λάθος τύπων $\leftrightarrow$ η αποτίμηση 'κολλάει'
- Επιθυμητές ιδιότητες συστήματος τύπων
  - ασφάλεια (η αποτίμηση διατηρεί την ορθότητα)
  - πρόοδος (η αποτίμηση ορθών εκφράσεων δεν κολλάει)
- Μελέτη τύπων μέσω του ορισμού σχέσης: $\Gamma \vdash t : T$
- Απάλειψη $\longrightarrow$ πότε μία έκφραση χωρίς τύπους μπορεί να πάρει τύπο σε ένα σύστημα

Επισκόπηση - I

- Συστήματα τύπων: στατικές μη πλήρεις αποδείξεις ορθότητας που προσεγγίζουν τον υπολογισμό
  - εντοπίζουν (συντηρητικά) λανθασμένες χρήσεις τιμών
  - χρήσιμα για: εντοπισμό σφαλμάτων, αφαίρεση, τεκμηρίωση, ασφάλεια
  - εκφραστικότητα: πόσες <u>ορθές</u> εκφράσεις επιτρέπονται
- λ-λογισμός χωρίς τύπους + φυσικοί + αληθοτιμές
  - λάθος τύπων $\leftrightarrow$ η αποτίμηση ‘κολλάει’
- Επιθυμητές ιδιότητες συστήματος τύπων
  - ασφάλεια (η αποτίμηση διατηρεί την ορθότητα)
  - πρόοδος (η αποτίμηση ορθών εκφράσεων δεν κολλάει)
- Μελέτη τύπων μέσω του ορισμού σχέσης: $\Gamma \vdash t : T$
- Απάλειψη $\longrightarrow$ πότε μία έκφραση χωρίς τύπους μπορεί να πάρει τύπο σε ένα σύστημα

Επισκόπηση - Ι

- Συστήματα τύπων: στατικές μη πλήρεις αποδείξεις ορθότητας που προσεγγίζουν τον υπολογισμό
  - εντοπίζουν (συντηρητικά) λανθασμένες χρήσεις τιμών
  - χρήσιμα για: εντοπισμό σφαλμάτων, αφαίρεση, τεκμηρίωση, ασφάλεια
  - εκφραστικότητα: πόσες <u>ορθές</u> εκφράσεις επιτρέπονται
- λ-λογισμός χωρίς τύπους + φυσικοί + αληθοτιμές
  - λάθος τύπων $\leftrightarrow$ η αποτίμηση 'κολλάει'
- Επιθυμητές ιδιότητες συστήματος τύπων
  - ασφάλεια (η αποτίμηση διατηρεί την ορθότητα)
  - πρόοδος (η αποτίμηση ορθών εκφράσεων δεν κολλάει)
- Μελέτη τύπων μέσω του ορισμού σχέσης: $\Gamma \vdash t : T$
- Απάλειψη $\longrightarrow$ πότε μία έκφραση χωρίς τύπους μπορεί να πάρει τύπο σε ένα σύστημα

Επισκόπηση - Ι

- Συστήματα τύπων: στατικές μη πλήρεις αποδείξεις ορθότητας που προσεγγίζουν τον υπολογισμό
  - εντοπίζουν (συντηρητικά) λανθασμένες χρήσεις τιμών
  - χρήσιμα για: εντοπισμό σφαλμάτων, αφαίρεση, τεκμηρίωση, ασφάλεια
  - εκφραστικότητα: πόσες <u>ορθές</u> εκφράσεις επιτρέπονται
- λ-λογισμός χωρίς τύπους + φυσικοί + αληθοτιμές
  - λάθος τύπων $\leftrightarrow$ η αποτίμηση `κολλάει'
- Επιθυμητές ιδιότητες συστήματος τύπων
  - ασφάλεια (η αποτίμηση διατηρεί την ορθότητα)
  - πρόοδος (η αποτίμηση ορθών εκφράσεων δεν κολλάει)
- Μελέτη τύπων μέσω του ορισμού σχέσης: $\Gamma \vdash t : T$
- Απάλειψη $\longrightarrow$ πότε μία έκφραση χωρίς τύπους μπορεί να πάρει τύπο σε ένα σύστημα

Επισκόπηση - I

- Συστήματα τύπων: στατικές μη πλήρεις αποδείξεις ορθότητας που προσεγγίζουν τον υπολογισμό
  - εντοπίζουν (συντηρητικά) λανθασμένες χρήσεις τιμών
  - χρήσιμα για: εντοπισμό σφαλμάτων, αφαίρεση, τεκμηρίωση, ασφάλεια
  - εκφραστικότητα: πόσες ορθές εκφράσεις επιτρέπονται
- λ-λογισμός χωρίς τύπους + φυσικοί + αληθοτιμές
  - λάθος τύπων $\leftrightarrow$ η αποτίμηση ‘κολλάει’
- Επιθυμητές ιδιότητες συστήματος τύπων
  - ασφάλεια (η αποτίμηση διατηρεί την ορθότητα)
  - πρόοδος (η αποτίμηση ορθών εκφράσεων δεν κολλάει)
- Μελέτη τύπων μέσω του ορισμού σχέσης: $\Gamma \vdash t : T$
- Απάλειψη $\longrightarrow$ πότε μία έκφραση χωρίς τύπους μπορεί να πάρει τύπο σε ένα σύστημα

Επισκόπηση - Ι

- Συστήματα τύπων: στατικές μη πλήρεις αποδείξεις ορθότητας που προσεγγίζουν τον υπολογισμό
    - εντοπίζουν (συντηρητικά) λανθασμένες χρήσεις τιμών
    - χρήσιμα για: εντοπισμό σφαλμάτων, αφαίρεση, τεκμηρίωση, ασφάλεια
    - εκφραστικότητα: πόσες <u>ορθές</u> εκφράσεις επιτρέπονται
- λ-λογισμός χωρίς τύπους + φυσικοί + αληθοτιμές
    - λάθος τύπων $\leftrightarrow$ η αποτίμηση ‘κολλάει’
- Επιθυμητές ιδιότητες συστήματος τύπων
    - ασφάλεια (η αποτίμηση διατηρεί την ορθότητα)
    - πρόοδος (η αποτίμηση ορθών εκφράσεων δεν κολλάει)
- Μελέτη τύπων μέσω του ορισμού σχέσης: $\Gamma \vdash t : T$
- Απάλειψη $\longrightarrow$ πότε μία έκφραση χωρίς τύπους μπορεί να πάρει τύπο σε ένα σύστημα

Επισκόπηση - II

- Ο τελεστής αναδρομής Y δεν έχει τύπο στον λ-λογισμό με απλούς τύπους
- Απλές επεκτάσεις: εγγραφές, εναλλακτικές τιμές και αναδρομή ως πρωταρχικός τελεστής
- Subtyping: $S <: T$ Οι τιμές ενός υποτύπου S είναι τιμές και του υπερτύπου T
  - χαλαρώνει το σύστημα τύπων επιτρέποντας περισσότερες εφαρμογές
  - βασικές ιδιότητες: υπαγωγή, ανακλαστικότητα, μεταβατικότητα
  - αντιμονοτονικές σχέσεις subtyping σε εγγραφές / συναρτήσεις
  - απλοποίηση της σύνταξης για εναλλακτικούς τύπους
  - χρήση κορυφαίου τύπου Top

Επισκόπηση - II

- Ο τελεστής αναδρομής Y δεν έχει τύπο στον λ-λογισμό με απλούς τύπους
- Απλές επεκτάσεις: εγγραφές, εναλλακτικές τιμές και αναδρομή ως πρωταρχικός τελεστής
- Subtyping: $S <: T$ Οι τιμές ενός υποτύπου S είναι τιμές και του υπερτύπου T
  - χαλαρώνει το σύστημα τύπων επιτρέποντας περισσότερες εφαρμογές
  - βασικές ιδιότητες: υπαγωγή, ανακλαστικότητα, μεταβατικότητα
  - αντιμονοτονικές σχέσεις subtyping σε εγγραφές / συναρτήσεις
  - απλοποίηση της σύνταξης για εναλλακτικούς τύπους
  - χρήση κορυφαίου τύπου Top

Επισκόπηση - II

- Ο τελεστής αναδρομής Y δεν έχει τύπο στον λ-λογισμό με απλούς τύπους
- Απλές επεκτάσεις: εγγραφές, εναλλακτικές τιμές και αναδρομή ως πρωταρχικός τελεστής
- Subtyping: $S <: T$ Οι τιμές ενός υποτύπου S είναι τιμές και του υπερτύπου T
  - χαλαρώνει το σύστημα τύπων επιτρέποντας περισσότερες εφαρμογές
  - βασικές ιδιότητες: υπαγωγή, ανακλαστικότητα, μεταβατικότητα
  - αντιμονοτονικές σχέσεις subtyping σε εγγραφές / συναρτήσεις
  - απλοποίηση της σύνταξης για εναλλακτικούς τύπους
  - χρήση κορυφαίου τύπου Top

Επισκόπηση - II

- Ο τελεστής αναδρομής Y δεν έχει τύπο στον λ-λογισμό με απλούς τύπους
- Απλές επεκτάσεις: εγγραφές, εναλλακτικές τιμές και αναδρομή ως πρωταρχικός τελεστής
- Subtyping: $S <: T$ Οι τιμές ενός υποτύπου S είναι τιμές και του υπερτύπου T
  - χαλαρώνει το σύστημα τύπων επιτρέποντας περισσότερες εφαρμογές
  - βασικές ιδιότητες: υπαγωγή, ανακλαστικότητα, μεταβατικότητα
  - αντιμονοτονικές σχέσεις subtyping σε εγγραφές / συναρτήσεις
  - απλοποίηση της σύνταξης για εναλλακτικούς τύπους
  - χρήση κορυφαίου τύπου Top

Επισκόπηση - II

- Ο τελεστής αναδρομής Y δεν έχει τύπο στον λ-λογισμό με απλούς τύπους
- Απλές επεκτάσεις: εγγραφές, εναλλακτικές τιμές και αναδρομή ως πρωταρχικός τελεστής
- Subtyping: $S <: T$ Οι τιμές ενός υποτύπου S είναι τιμές και του υπερτύπου T
  - χαλαρώνει το σύστημα τύπων επιτρέποντας περισσότερες εφαρμογές
  - βασικές ιδιότητες: υπαγωγή, ανακλαστικότητα, μεταβατικότητα
  - αντιμονοτονικές σχέσεις subtyping σε εγγραφές / συναρτήσεις
  - απλοποίηση της σύνταξης για εναλλακτικούς τύπους
  - χρήση κορυφαίου τύπου Top

Επισκόπηση - II

- Ο τελεστής αναδρομής Y δεν έχει τύπο στον λ-λογισμό με απλούς τύπους
- Απλές επεκτάσεις : εγγραφές, εναλλακτικές τιμές και αναδρομή ως πρωταρχικός τελεστής
- Subtyping: $S <: T$ Οι τιμές ενός υποτύπου S είναι τιμές και του υπερτύπου T
    - χαλαρώνει το σύστημα τύπων επιτρέποντας περισσότερες εφαρμογές
    - βασικές ιδιότητες : υπαγωγή, ανακλαστικότητα, μεταβατικότητα
    - αντιμονοτονικές σχέσεις subtyping σε εγγραφές / συναρτήσεις
    - απλοποίηση της σύνταξης για εναλλακτικούς τύπους
    - χρήση κορυφαίου τύπου Top

Επισκόπηση - II

- Ο τελεστής αναδρομής Y δεν έχει τύπο στον λ-λογισμό με απλούς τύπους
- Απλές επεκτάσεις: εγγραφές, εναλλακτικές τιμές και αναδρομή ως πρωταρχικός τελεστής
- Subtyping: $S <: T$ Οι τιμές ενός υποτύπου S είναι τιμές και του υπερτύπου T
  - χαλαρώνει το σύστημα τύπων επιτρέποντας περισσότερες εφαρμογές
  - βασικές ιδιότητες: υπαγωγή, ανακλαστικότητα, μεταβατικότητα
  - αντιμονοτονικές σχέσεις subtyping σε εγγραφές / συναρτήσεις
  - απλοποίηση της σύνταξης για εναλλακτικούς τύπους
  - χρήση κορυφαίου τύπου Top

Επισκόπηση - II

- Ο τελεστής αναδρομής Y δεν έχει τύπο στον λ-λογισμό με απλούς τύπους
- Απλές επεκτάσεις: εγγραφές, εναλλακτικές τιμές και αναδρομή ως πρωταρχικός τελεστής
- Subtyping: $S <: T$ Οι τιμές ενός υποτύπου S είναι τιμές και του υπερτύπου T
  - χαλαρώνει το σύστημα τύπων επιτρέποντας περισσότερες εφαρμογές
  - βασικές ιδιότητες: υπαγωγή, ανακλαστικότητα, μεταβατικότητα
  - αντιμονοτονικές σχέσεις subtyping σε εγγραφές / συναρτήσεις
  - απλοποίηση της σύνταξης για εναλλακτικούς τύπους
  - χρήση κορυφαίου τύπου Top