Optimisation de la vitesse de compilation par la fusion entre inférence de types et analyse syntaxique

> Enogad Le Biavant– Frederic

Présentation

generale

Parsing

D/ 1. .

Formalisation

.

# Optimisation de la vitesse de compilation par la fusion entre inférence de types et analyse syntaxique

Enogad Le Biavant-Frederic

Alain René Lesage MPI

2025

Optimisation de la vitesse de compilation par la fusion entre inférence de types et analyse syntaxique

> Enogad Le Biavant– Frederic

#### Présentation générale

Définitions Parsing

\_ . .

Formalisation

Karm, 2022



Optimisation de la vitesse de compilation par la fusion entre inférence de types et analyse

> Enogad Le Biavant-

Frederic

# Présentation générale

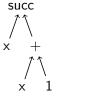
Définitions Parsing

r neone des c

Formalisatio

Conclusion

$$\Gamma = \{ \texttt{+}: int \rightarrow int \rightarrow int \}$$
 let  $succ = \lambda x. (\texttt{+} \ x \ 1)$ 





Optimisation de la vitesse de compilation par la fusion entre inférence de types et analyse syntaxique

> Enogad Le Biavant– Frederic

#### Présentation générale

Définitions

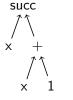
Théorie des typ

Resultat

I OIIIIalisatioi

۸....

$$\Gamma = \{ \texttt{+}: int \rightarrow int \rightarrow int, \ \texttt{x}: \tau \}$$
 let  $succ = \lambda x. (\texttt{+} \ x \ 1)$ 



Optimisation de la vitesse de compilation par la fusion entre inférence de types et analyse syntaxique

> Enogad Le Biavant– Frederic

#### Présentation générale

Définitions

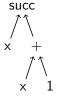
Théorie des type

Résultats

Formalisatio

Conclusion

$$\Gamma = \{ \texttt{+}: int \rightarrow int \rightarrow int, \ \texttt{x}: \tau \}$$
 let  $succ = \lambda x. (\texttt{+} \ x \ 1)$ 



Optimisation de la vitesse de compilation par la fusion entre inférence de types et analyse syntaxique

#### Présentation générale

$$\Gamma = \{ \mathbf{+} : int \rightarrow int \rightarrow int, \ \mathbf{x} : \tau \}$$
 let  $succ = \lambda x. (+ \ x \ 1)$ 

Parsing récursif descendant

succ 
$$\{x:\tau\} / \bigwedge \{\tau = int\}$$
 
$$\times + \{x:\tau\} / \bigwedge$$
 
$$\times 1$$

### Grammaire

```
Optimisation
de la vitesse
de compilation
par la fusion
entre inférence
de types et
analyse
syntaxique
```

Enogad Le Biavant– Frederic

Présentation générale

\_ ... . .

Parsing

Théorie des types

Resultats

Formalisation

۸ -- -- -- -

```
program
                 expr
                 abs
expr
                 app
                 letbinding
            := term [\{ term \}]
app
abs
            := "\" id "." expr
               "let" id "=" expr "in" expr
letbinding
term
                 string
                 int
                 bool
                 id
                 "(" expr ")"
```

## TT - Définitions

Optimisation de la vitesse de compilation par la fusion entre inférence de types et analyse syntaxique

> Enogad Le Biavant– Frederic

Présentation générale

Définitions

Parsing
Théorie des types

\_ . .

T Officialisation

Anneve

Définition de Type : classification de termes (Church). Théorie de travail : Lambda calcul simplement typé (LCST), polymorphique.

let 
$$id = \lambda x.x : \forall \sigma \to \sigma$$

# **Opérations**

Optimisation de la vitesse de compilation par la fusion entre inférence de types et analyse syntaxique

Théorie des types

Déduction naturelle et calcul des séquents (Gentzen).

$$H_1,...,H_n \vdash A_1,...,A_n$$

$$\frac{A \quad B}{C}$$

# Hindley-Milner

Optimisation de la vitesse de compilation par la fusion entre inférence de types et analyse syntaxique

> Enogad Le Biavant– Frederic

Présentation générale

Définitions

Parsing
Théorie des types

Résultat

Conclusion

$$\frac{x:\sigma\in\Gamma}{\Gamma\vdash x:\sigma} \text{ var }$$

$$\frac{\Gamma, x: \tau \vdash e: \tau'}{\Gamma \vdash \lambda x. e: \tau \rightarrow \tau'} \text{ abs }$$

$$\frac{\Gamma \vdash f: \tau \to \tau' \qquad \Gamma \vdash e: \tau}{\Gamma \vdash f \ e: \tau'} \ \mathsf{app}$$

# Hindley-Milner

Optimisation de la vitesse de compilation par la fusion entre inférence de types et analyse syntaxique

> Enogad Le Biavant– Frederic

Présentation générale

D/C-11

Théorie des types

Résultat

Formalisation

Anneve

#### Algorithme W

- Assignation de variables de types aux expressions
- 2 Génération de contraintes
- **3** Substitutions
- **4** Unification
- Instantiation, généralisation

$$\begin{cases} x:\tau \} / \bigwedge \{expr:\beta\} \\ \times & \text{expr} \end{cases}$$
 
$$\{\beta = int, \tau = int \} \uparrow \\ + \{x:\tau \} / \bigwedge \\ \times & 1$$

# Résultats

Optimisation de la vitesse de compilation par la fusion entre inférence de types et analyse syntaxique

> Enogad Le Biavant– Frederic

Présentation générale

Définitions
Parsing
Théorie des type

Résultats

Formalisatio

Jonetusioi

Anneve

 $\mathcal{W}: \tilde{\Gamma} \times Expr \rightarrow Subst \times Type \\ \mathcal{W}^*: \tilde{\Gamma} \times L \rightarrow Subst \times \Gamma \times Expr \times L$ 

Machine: i7 5th gen 3.00Ghz

Fichier de test: 1000 premiers nombres de church

Version non optimisée :  $\approx 26.37s$ 

Version optimisée :  $\approx 3.88s$ 

Fichier de test :

## Formalisation

Optimisation de la vitesse de compilation par la fusion entre inférence de types et analyse syntaxique

> Enogad Le Biavant– Frederic

Présentation générale

générale

Parsing Parsing

D/ 1. .

Formalisation

Anneve

Automates d'arbres :  $\mathcal{A} = (\mathcal{F}, Q, Q_f, \Delta)$ 

lacksquare : Alphabet gradué (fonction d'arité ar)

# Conclusion

Optimisation de la vitesse de compilation par la fusion entre inférence de types et analyse syntaxique

> Enogad Le Biavant– Frederic

Présentation

.

Définitions

Théorie des ty

Résultats

Formalisatio

Conclusion

# Annexe

Optimisation de la vitesse de compilation par la fusion entre inférence de types et analyse syntaxique

> Enogad Le Biavant– Frederic

Présentatio

Définitions

Théorie des ty

Résultats

Formalisation

Conclusion