

λ -Nantes Workshop 2bis

Apprenons à aimer le λ -calcul simplement typé

2025-10-09

Agenda

- ▶ Introduction

Agenda

- ▶ Introduction
- ▶ Action

Agenda

- ▶ Introduction
- ▶ Action
- ▶ Conclusion

Introduction

Rappel de l'épisode précédent

- ▶ On a codé un λ -calcul non typé en Rust, Elixir, Clojure et Java

Rappel de l'épisode précédent

- ▶ On a codé un λ -calcul non typé en Rust, Elixir, Clojure et Java
- ▶ En jouant contre un serveur nous envoyant des expressions à évaluer

Rappel de l'épisode précédent

- ▶ On a codé un λ -calcul non typé en Rust, Elixir, Clojure et Java
- ▶ En jouant contre un serveur nous envoyant des expressions à évaluer
- ▶ On s'est bien amusé

Rappel de l'épisode précédent

- ▶ On a codé un λ -calcul non typé en Rust, Elixir, Clojure et Java
- ▶ En jouant contre un serveur nous envoyant des expressions à évaluer
- ▶ On s'est bien amusé
- ▶ On veut aller plus loin

Rappel de l'épisode précédent

Pour accéder au repository, c'est par là ⇒



Figure 1: Lambda-Nantes code

Aujourd'hui

- ▶ On passe à la vitesse supérieure en ajoutant des types simples au λ -calcul

Aujourd'hui

- ▶ On passe à la vitesse supérieure en ajoutant des types simples au λ -calcul
- ▶ Cela donne le *λ -calcul simplement typé*

Aujourd'hui

$$\begin{array}{ll} \frac{x:\sigma \in \Gamma}{\Gamma \vdash x:\sigma} \quad (1) & \frac{c \text{ is a constant of type } T}{\Gamma \vdash c:T} \quad (2) \\ \frac{\Gamma, x:\sigma \vdash e:\tau}{\Gamma \vdash (\lambda x:\sigma. e):(\sigma \rightarrow \tau)} \quad (3) & \frac{\Gamma \vdash e_1:\sigma \rightarrow \tau \quad \Gamma \vdash e_2:\sigma}{\Gamma \vdash e_1 e_2:\tau} \quad (4) \end{array}$$

Figure 2: λ -calcul simplement typé