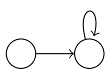
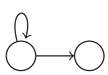
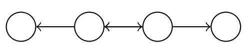
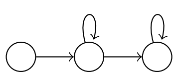
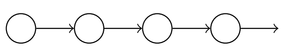
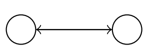
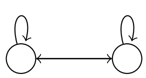
TD2 7 février 2017

λ-calcul et logique informatique

bury@lsv.ens-cachan.fr

Exercice 1 — (Graphes de réduction)
Pour chacun des graphes suivants, trouver un λ-terme qui a pour graphe de réduction ce graphe.

1. 
2. 
3. 
4. 
5. 
6. 

7. 
8. 

Exercice 2 — (Réduction parallèle)
Dans cet exercice, la flèche simple $\rightarrow$ dénote la β-réduction. On définit la réduction parallèle comme suit :

$$\frac{u \Rightarrow u' \quad v \Rightarrow v'}{uv \Rightarrow u'v'}\ \text{app} \qquad \frac{u \Rightarrow u'}{\lambda x.u \Rightarrow \lambda x.u'}\ \text{abs}$$

$$\frac{}{u \Rightarrow u}\ \text{refl} \qquad \frac{u \Rightarrow u' \quad v \Rightarrow v'}{(\lambda x.u)v \Rightarrow u'[x := v']}\ \text{beta}$$

1. Montrer les inclusions suivantes en donnant des exemples illustrant qu'elles sont strictes :
$$\rightarrow \subseteq \Rightarrow \subseteq \rightarrow^*$$
2. Montrer que $\Rightarrow$ est fortement confluente.
3. En déduire que $\rightarrow$ est confluente.

Exercice 3 — Encodages

1. Définir un codage des paires, c'est à dire des termes **pair**, π_1 et π_2 tels que pour tout N_1 et N_2 on a :
$$\pi_i\ (\mathbf{pair}\ N_1\ N_2) \rightarrow^* N_i$$
A-t-on $M =_\beta \mathbf{pair}\ (\pi_1\ M)\ (\pi_2\ M)$ pour tout M ? est-ce un problème ?
2. Pour les entiers, on utilise l'encodage de Church :
$$\overline{n} = \lambda s.\lambda x.\ s^n\ x = \lambda s.\lambda x.\ s\ (s\ (\dots\ x))$$
Rappeler l'encodage de zéro, du successeur, de l'addition et de la multiplication.
3. Définir le test à zéro pour les entiers de Church.
4. Définir le prédecesseur pour les entiers de Church, avec $\mathbf{pred}\ \overline{0} = \overline{0}$.
5. Donner un codage des listes, et définir la liste vide, l'ajout d'un élement en tête d'une liste, le test du vide, la concaténation et le renversement.

Exercice 4 — Stratégie interne faible

La stratégie interne faible, aussi appelée *appel par valeur*, est aussi proche que possible des langages de programmation traditionnels. Elle ne réduit jamais sous les abstractions, et ne réduit un β-redex que lorsque son argument est déja complètement réduit — on dit alors que c'est une *valeur*.

On définit l'ensemble des valeurs V par la grammaire suivante, où $\mathcal{V}$ est l'ensemble des variables et Λ l'ensemble de tous les termes :
$$V := \mathcal{V}V \dots V \mid \lambda x.\Lambda$$
On définit formellement notre stratégie comme suit :
$$\frac{v \in V}{(\lambda x.u)v \rhd u[x := v]} \qquad \frac{u \rhd u'}{uv \rhd u'v} \qquad \frac{u \in V \quad v \rhd v'}{uv \rhd uv'}$$

1. Utiliser le combinateur Y pour définir la fonction factorielle, de sorte que **fact** $\overline{n} \to_\beta^* \overline{n!}$. On rappelle la définition :

$$Y = \lambda f.\ (\lambda x.\ f\ (x\ x))\ (\lambda x.\ f\ (x\ x))$$

2. Quelles sont les réductions de **fact** (sans argument) par $\rhd^*$? Cela pose-t-il problème ? Si oui, proposer une solution.
3. Quelles sont les réductions de **fact** $\overline{0}$ par $\rhd^*$? Cela pose-t-il problème ? Si oui, proposer une solution.

Exercice 5 — Chemins sur Λ^*

On définit l'ensemble des chemins d'un terme $u \in \Lambda$, noté $\mathcal{C}(u)$, comme l'ensemble des objets p tels que $p \prec u$ est dérivable dans le système suivant :

$$\frac{}{x \prec x} \qquad \frac{P \prec M}{\mathbf{L}\ P \prec M\ N} \qquad \frac{P \prec N}{\mathbf{R}\ P \prec M\ N} \qquad \frac{P \prec M}{\lambda x.P \prec \lambda x.M}$$

On travaille modulo α, en renommant implicitement les variables liées pour pouvoir appliquer la règle de passage sous l'abstraction. Par conséquent, deux termes α-équivalents ont les mêmes chemins, et l'ensemble des chemins d'un terme est clos par α-équivalence.

1. Donner les chemins des termes $\lambda x.\ x\ y$, $\lambda x.\ x\ x$, et $\lambda x.\ (\lambda y.\ y)\ x$.
2. Que peut-on dire de deux termes qui ont les mêmes chemins ?
3. On étend notre définition de chemins à Λ^* en rajoutant la règle suivante :

$$\frac{P \prec M \quad Q \prec N}{P[x := Q] \prec (\lambda^* x.\ M)\ N}$$

   Montrer que $M \to N$ implique $\mathcal{C}(M) = \mathcal{C}(N)$, où $\to$ est la réduction de Λ^* qui ne touche qu'aux rédexes annotés d'une étoile.
4. Montrer la réciproque : $\mathcal{C}(M) = \mathcal{C}(N)$ implique $M \leftrightarrow^* N$, c'est à dire la convertibilité dans Λ^*.