

## TD de Programmation Fonctionnelle

### Exercice 1

On fixe un ensemble de variables de type  $V = \{\alpha, \beta, \gamma\}$  et un ensemble de types atomiques  $A = \{\mathbf{int}, \mathbf{bool}\}$ .

Expliquer pourquoi chacun des objets syntaxiques suivants est bien un élément de l'ensemble des types simples  $T$ , et construire son arbre de syntaxe abstraite :

- $\alpha$
- $\mathbf{int}$
- $(\alpha \rightarrow (\alpha \rightarrow \mathbf{bool}))$
- $((\mathbf{int} \rightarrow \mathbf{int}) \rightarrow (\mathbf{int} \rightarrow \mathbf{int}))$
- $(\alpha \rightarrow (\beta \rightarrow \alpha))$
- $((\alpha \rightarrow (\beta \rightarrow \gamma)) \rightarrow ((\alpha \rightarrow \beta) \rightarrow (\alpha \rightarrow \gamma)))$

### Exercice 2

Pour chacun des  $\lambda$ -termes suivants, construire son arbre de syntaxe abstraite, appliquer la fonction « typer », résoudre le système d'équations résultant et en déduire le type du  $\lambda$ -terme.

1)  $\lambda x. \lambda y. x$

2)  $\lambda x. (x \ x)$

3)  $\lambda x. \lambda y. \lambda z. ((x \ z) \ (y \ z))$

### Exercice 3

Typer le  $\lambda$ -terme  $((\lambda x. x) \ (\lambda y. y))$  en construisant un arbre de preuve qui lui associe un type.

(On utilisera seulement la définition des termes simplement typés, et pas la fonction « typer » ou l'algorithme d'unification.)