# Maîtrise d'Informatique
# Traduction de Programmes

26 Mai 1999

*Tous documents autorisés*

1. Je voudrais ajouter à un langage à typage stricte la possibilité de déclarer de nouveaux opérateurs infixes, éventuellement surchargés, et de spécifier leur priorité (une constante entière positive).
   (a) Proposer un syntaxe adéquat à ces déclarations et spécifications.
   (b) Expliquer pourquoi la priorité d'un opérateur surchargé ne devrait pas dépendre des types de ses opérands.
   (c) L'arbre construit par l'analyse syntaxique ne peut plus représenter la structure d'une expression en sous-expressions sémantiques. Pourquoi ?
   (d) Donner un système d'attributs et leurs règles de calcul permettant de déterminer le type d'une expression qui contient des applications de ces nouveaux opérateurs.
   (Votre syntaxe des expressions n'est pas obligé de traiter tableaux, structures ni appels de fonctions.)
2. Quand il est évident au compilateur que chaque exécution finie d'une boucle doit comporter un nombre pair d'itérations, l'optimisateur peut réduire le nombre d'exécutions de tests et de sauts, en *déroulant* la boucle : le nombre d'itérations sera divisé par 2 et chaque itération effectue deux fois une itération de la boucle initiale.
   (a) Décrire une classe assez large de boucles pour lesquelles le compilateur peut décider que le nombre d'itérations sera toujours pair.
   (b) Quels sont les algorithmes à utiliser pour reconnaître les boucles de votre classe ?
   (c) Cette "optimisation" réduira-t-elle le temps d'exécution ?