Devoir sur table – Novembre 2000
MIAS 1ère année – 1er semestre

Aucun document ni machine électronique n'est permis à l'exception de la carte de référence de Scheme.

Répondre sur la feuille même, dans les boîtes appropriées. La taille des boîtes suggère le nombre de lignes de la réponse attendue (utiliser le dos de la feuille précédente si la réponse déborde des boîtes). Le barême (total sur 32) apparaissant dans chaque boîte n'est donné qu'à titre indicatif.

La clarté des réponses et la présentation des programmes seront appréciées. Toutes les fonctions apparaissant dans les réponses doivent être accompagnées de leur spécification. Ne pas désagrafer les feuilles.

Exercice 1

Qu'appelle-t'on une forme spéciale ? Quelles sont les formes spéciales de Scheme que vous connaissez ?

[4/32]

Exercice 2

Écrire une expression valant `((1 2)(3))` en utilisant au moins un `cons` et au moins un `list`.

[4/32]

Section	Groupe	Nom	Prénom

Exercice 3

Écrire une fonction, nommée `somme-carres`, qui prend un entier n et qui calcule la somme des carrés des entiers de l'ensemble $[-n \ldots + n]$? Ainsi :

```
(somme-carres 1) → 2
(somme-carres 2) → 10
```

[6/32]

Section	Groupe	Nom	Prénom

Exercice 4

Soit une liste non vide d'entiers naturels. Écrire une fonction, nommée `extraction`, qui prenant une telle liste renvoie cette liste privée de ses premiers termes ; le nombre de termes enlevés est précisément la valeur du premier d'entre eux. Ainsi

```
(extraction (list 2 3 4 5)) → (4 5)
(extraction (list 0 3 4 5)) → (0 3 4 5)
```

En revanche, `(extraction (list 9 1))` doit provoquer une erreur.

[6/32]

Section	Groupe	Nom	Prénom

Exercice 5

Écrire une fonction, nommée `paires`, prenant deux listes de même longueur et construisant une liste des couples des termes de rang correspondant des deux listes. Ainsi :

```
(paires (list 1 2 3) (list "un" "deux" "trois")) → ((1 "un") (2 "deux") (3 "trois"))
```

[6/32]

Section	Groupe	Nom	Prénom

Exercice 6

Que vaut l'expression `(mystere (list 1 2 3 4 5))` ? Que vaut l'expression `(mystere (list 1 2 3 4 5 6))` ? Écrire une spécification de la fonction `mystere` que voici :

```
;;; Une fonction bien mystérieuse.
(define (mystere x)
  (if (pair? x)
      (if (pair? (cdr x))
          (cons (list (car x) (cadr x))
                (mystere (cddr x)) )
          (list (list (car x) (car x))) )
      x ) )
```

[6/32]