Université de La Réunion — Vendredi 5 juin 1998

DEUG 1^{ere} année – Tronc A – MIAS 1 et MASS 1

Programmation fonctionnelle – Examen terminal **3**

Durée : 2 heures – Aucun document n'est autorisé.

Répondre uniquement dans les cadres prévus à cet effet

Nom :	No dossier :
Prénom(s) :	Signature :
Date de naissance :	

Exercice 3 : Complexité *(5 points)*

Cet exercice a pour but de vous faire programmer une fonction qui teste si un nombre est premier et de calculer sa complexité en temps. Vous utiliserez la méthode suivante : pour savoir si un nombre n est premier, on passe en revue tous les entiers compris entre 2 et $\sqrt{n}$ **(pas besoin d'aller jusqu'à** n**)** ; si l'un d'eux divise n, n n'est pas premier, sinon n est premier.

(0,5 pt) 3.a Écrivez la fonction `carré` qui retourne le carré du nombre qu'on lui donne en argument.

```
(define (carré n)
```

(0,5 pt) 3.b Écrivez la fonction `divise?` qui prend en argument deux entiers a et b et qui renvoie `#t` si a est divisible par b et `#f` sinon. Vous pourrez utiliser la fonction prédéfinie `remainder` qui retourne le reste de la division de deux entiers (Ex : (`remainder` 13 6) = 1).

```
(define (divise? a b)
```

(2 pts) 3.c Écrivez la fonction `trouve-diviseur` qui prend en argument un entier n et qui trouve le plus petit diviseur de n compris entre 2 et $\sqrt{n}$. Vous utiliserez pour cela **les deux** fonctions précédentes.

```
(define (trouve-diviseur)
```

*(0,5 pt)*3.d Écrivez la fonction `premier?` qui prend en argument un entier n et qui répond `#t` si n est premier et `#f` sinon.

```
(define (premier? n)
```

*(1,5 pt)*3.e Quelle est la complexité en temps de la fonction `premier?` (justifiez brièvement) ?