

DEUG 1^{ère} année – Tronc A – MIA5 1 et MASS 1
Programmation fonctionnelle – Examen terminal
Durée : 2 heures – Aucun document n'est autorisé.

Répondre uniquement dans les cadres prévus à cet effet

Nom :	N° table :
Prénom(s) :	N° dossier :
Date de naissance :	Signature :

Exercice 1 : Évaluation d'expressions (5 points)

Donnez le résultat de l'évaluation des expressions suivantes :

```
(car (cdr ( cdr (car '((1 2 3) 4 (5 6))))))  
=>  
(cons 'a (cons ( cons (cdr '(a b)) '()) '()))  
=>  
(map car '((1 2 3)(5)(6 7)))  
=>  
(map + '(1 2 3) '(4 5 6) '(5 3 1))  
=>  
( (lambda (x y) (+ (* x x) (* y y))) 3 4)  
=>
```

Exercice 2 : Graphisme (5 points)

Soit d un rationnel positif. À partir d'un point $M_1(x_1, y_1)$ du plan on définit les trois points $M_2(a, y_1)$ et $M_3(a, b)$ avec : $a = x_1 + d/2$ et $b = y_1 - d/2$

On construit un dessin fractale à l'ordre n , à partir du point M_1 et du rationnel d , de la manière suivantes :

Si $n = 0$, le dessin est réduit au point M_1 .

Sinon :

1. on trace le dessin à l'ordre $n - 1$, à partir du point M_1 et du rationnel $d/2$
2. on trace le segment de droite allant du point final du dessin effectué en 1 jusqu'au point M_2
3. on trace le dessin à l'ordre $n - 1$, à partir du point M_2 et du rationnel $d/2$
4. on trace le segment de droite allant du point final du dessin effectué en 3 jusqu'au point M_3
5. on trace le dessin à l'ordre $n - 1$, à partir du point M_3 et du rationnel $d/2$
6. on trace le segment de droite allant du point final du dessin effectué en 5 jusqu'au point M_1

Définissez une fonction `dessin` à quatre arguments $x1, y1, d$ et n permettant de tracer cette fractale.

```
(define (dessin x1 y1 d n)
```

Exercice 3 : (5 points)

Rappel : le prédicat Scheme `even?` permet de savoir si un nombre est pair ou non

(3 pts) **3.a** Définissez une fonction `suite` permettant de calculer le n^{e} terme de la suite U définie par :

$$\begin{cases} U_0 = 1 \\ U_{2n} = U_n + 3, \text{ pour } n > 0 \\ U_{2n+1} = 2 * U_n - 1, \text{ pour } n > 0 \end{cases}$$

(define (suite n)

(2 pts) **3.b** En remarquant que U_n est calculé à partir de $U_{E(n/2)}$ (où $E(n/2)$ désigne la partie entière de $n/2$), évaluez le nombre d'appel récursifs nécessaires pour calculer `(suite n)` et déduisez en la complexité en temps de la fonction `suite`.

Exercice 4 : (5 points)

(2 pts) **4.a** Définir une fonction `sauf` qui prend comme arguments un atome `a` et une liste `l` et renvoie comme résultat la liste `l` privée de toutes les occurrences de `a` à tous les niveaux de `l`.

(define (sauf a l)

(1 pt) **4.b** En utilisant la fonction `sauf`, définir une fonction `enleve` qui prend comme arguments deux listes `l1` et `l2` et renvoie la liste `l2` privée de toutes les occurrences de tous les éléments de la liste `l1`.

(define (enleve l1 l2))

(2 pts) **4.c** En justifiant brièvement votre réponse, donnez les complexités en temps des fonctions `sauf` et `enleve`.