

Numéro de carte d'étudiant :

Numéro de place :

Rédiger vos réponses comme demandé sur le sujet et glissez-le dans la copie réglementaire sur laquelle vous aurez également indiqué vos noms, prénoms et numéro de carte d'étudiant.

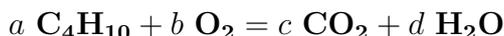
Le barème annoncé sera peut-être modifié. Les notes seront par la suite ramenées vers 20 par un coefficient choisi par les correcteurs.

Rappel : Quand une commande est suivie du délimiteur \$, Maxima n'affiche pas ce qu'il a effectué dans son noyau de calcul relativement à cette commande. Respectez cette règle quand vous rencontrerez ci-dessous ces délimiteurs dollar.

Exercice 1. Ecrire dans les zones blanches, ce que donne Maxima quand on exécute la cellule suivante :

```
(%i1) L: [2,3,1,8,-5]$ L[2];  
  
(%o1)  
  
(%i2) L: append(L, [lmin(L), lmax(L)]);  
  
(%o2)  
  
(%i3) L.member(6,L);  
  
(%o3)  
  
(%i4) taylor(exp(x),x,0,4):  
  
(%o4)  
  
(%i5) eq: 'diff(y,x)-y=0$ sol_gen(ode2,eq,y,x)$ sol: ic1(sol_gen,x=0,y=1);  
  
(%sol)  
  
(%i6) expand((a+%ib)**2)  
  
(%o6)  
  
(%i7) limit((2*x**4+x-2)/(x**3-1),x,-inf)  
  
(%o7)  
  
(%i8) trigrat(sin(x)^2)  
  
(%o8)
```

Exercice 2. On souhaite utiliser Maxima pour équilibrer la formule de combustion du Butane :



Compléter le code suivant pour obtenir des valeurs de a, b, c, d entières positives

```
(%i1) s:solve([
,
], [a,b,c]);

(s) [[a=%r1/5,b=%r1/2,c=4%r1/5,d=%r1]]

(%i2) subst(
=
,
);

(%o2) [a=2,b=5,c=8,d=10]
```

Exercice 3. La commande `primep` de Maxima prend en entrée un entier et retourne `true` si cet entier est premier et `false` sinon.

1. Compléter la fonction suivante prenant en paramètre un entier N et retournant la liste des nombres premiers compris entre 1 et N .

```
(%i1) listePremiers(N):=block([
],

)$

(%i2) listePremiers(20);
(%o2) [2,3,5,7,11,13,17,19]
```

2. Deux nombres premiers $p < q$ sont *jumeaux* si $q = p + 2$. Compléter la fonction suivante prenant en paramètre un entier N et retournant la liste des couples de nombres premiers jumeaux entre 1 et N .

```
(%i1) listePremiersJumeaux(N):=block([
],

)$

(%i2) listePremiersJumeaux(20);
(%o2) [[2,3],[3,5],[5,7],[11,13],[17,19]]
```

Exercice 4. Rappelons que si R est un réel strictement positif, l'ensemble des points du plan de coordonnées (x, y) vérifiant

$$(x - x_A)^2 + (y - y_A)^2 - R^2 = 0$$

est le cercle de centre $A(x_A, y_A)$ et de rayon R . Rappelons aussi que l'ensemble des points de coordonnées (x, y) vérifiant l'équation

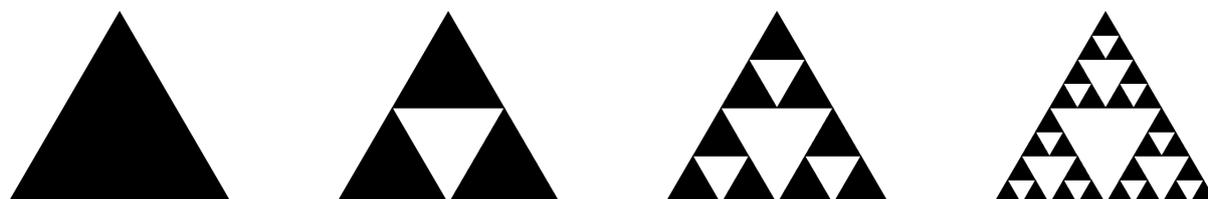
$$(x_B - x_A) \times (y - y_A) - (x - x_A) \times (y_B - y_A) = 0$$

est l'équation de la droite passant par $A(x_A, y_A)$ et $B(x_B, y_B)$.

1. Reproduire sur votre copie, le dessin obtenu avec Maxima avec le programme ci-dessous.
2. Quel est le but de cette construction?

```
(%i9)norme(A,B):=sqrt((A[1]-B[1])^2+(A[2]-B[2])^2)$
polycercle(A,B):=(x-A[1])^2+(y-A[2])^2-norme(A,B)^2$
polydroite(A,B):=(B[1]-A[1])*(y-A[2])-(x-A[1])*(B[2]-A[2])$
cercle(A,B):=block([R],R:norme(A,B),
    parametric(A[1]+R*cos(t),A[2]+R*sin(t),t,0,2*%pi))$
A:[0,0]$B:[2,1]$C:[3,4]$
s1:solve([polycercle(A,B)=0,polydroite(A,C)=0],[x,y])$
B1:subst(s1[2],[x,y])$ /*on admet ici que les coordonnées de B1
    sont positives*/
s2:solve([polycercle(B1,A)=0,polycercle(B,A)=0],[x,y])$
M1:subst(s2[1],[x,y])$ /*on admet ici que le point M1 n'est pas A*/
e:0.25$
wxdraw2d(
    proportional_axes='xy,point_type=circle,color=black,
    points([A,B,B1,C,M1]),
    points_joined=true,color=red,
    points([A,B]),points([A,C]),points([A,M1]),
    color=black,nticks=100,color=black,cercle(A,B1),
    color=blue,cercle(B,A),cercle(B1,A),
    xrange=[-2.5,4.5],yrange=[-2.5,4.5],
    color=black,label(["A",A[1]-e,A[2]-e]),label(["B",B[1]+e,B[2]]),
    label(["B1",B1[1]-e,B1[2]]),label(["C",C[1]+e,C[2]]),
    label(["M1",M1[1]+e,M1[2]+e])
);
```

Exercice 5. Une fractale est un objet géométrique autosimilaire, c'est à dire que l'on peut retrouver la fractale dans son intégralité en zoomant sur une de ses parties. C'est naturellement un objet limite, tous les détails d'une fractale ne peuvent pas être représentés sur une figure. Voici les fractales de Sierpiński S_d avec un degré de détail d valant respectivement 1, 2, 3 et 4.



Cette fractale est obtenue par application successive d'une transformation élémentaire portant sur les triangles : un triangle (dessiné en noir) donne lieu à trois triangles plus petits. La transformation élémentaire correspond exactement au passage de la fractale S_1 à la fractale S_2 . Pour cette partie nous allons représenter les triangles comme une liste de trois points (eux-mêmes représentés comme des listes de deux nombres). Une figure de Sierpiński sera donc naturellement représentée par une liste de triangles. Par exemple, la figure S_1 est donnée par :

`[[[0,0],[0.5,sqrt(3)/2],[1,0]]].`

Il y a bien trois niveaux de crochets ci-dessus.

1. Compléter la fonction suivante qui étant donnés deux points A et B, retourne le milieu du segment [A,B].

```
(%i1) milieu(A,B):=block([
    ],
    )$
(%i2) milieu([1,2],[3,4]);
(%o2) [2,3]
```

2. Compléter la fonction suivante qui étant donné un triangle T, retourne une liste de trois triangles correspondant à la transformation élémentaire de Sierpiński.

```
(%i3) elemSierpinsky(T):=block([
    ],
    )$
```

3. Compléter la fonction suivante Sierpinsky qui étant donné un paramètre entier d, retourne une liste de triangles correspondant à la fractale de Sierpiński S_d .

```
(%i4) Sierpinsky(d):=block([
    Triangles:[0,0],[0.5,sqrt(3)/2],[1,0]
    ],
    )$
```