

```

Donne "L 200 donne "H 200 //dimensions de l'écran//
eff accélère déroule ct
moderadian

```

```

pour f :t
rends (-3*cos :t )- (2*cos (3* :t ))
fin

```

```

pour g :t
rends (-3*sin :t ) + (2*sin (3* :t ))
fin

```

```

pour axes :A :B :C :D
// A, B, C, D seront des bornes données. Si on donne x entre A et B, et y entre C et D alors les coordonnées sur l'écran sont
U = L*(X-A) / (B-A) et V = H*(Y - D) / (C - D). On élargira cette question dans les programmes graphiques suivants //
si (:A * :B) < 0 segment [ entier( (: L *: A) / (:A - :B)) 0] [entier( (:L *:A) / (:A - :B)) :H]
si (:C * :D) < 0 segment [ 0 entier ((:H *: D) / (:D - :C))] [: L entier((:H *: D) / (:D - :C))]
fin

```

```

pour tracer :A :B :C :D :t1 :t2 :h
// Réalise le tracé de la courbe pour A < X < B et C < Y < D et t1 < t < t2 avec t allant de h en H ( le pas de calcul ).//
axes :A :B :C :D donne "t :t1
fixexy entier (((:L * (f :t) ) - :A) / (:B - :A)) entier ((:H * ((g :t) - :D)) / (:C -:D))
tantque :t <= :t2 [
    va [ entier((:L *((f :t) - :A) / (:B -:A)) entier((:H *((g :t) -: D)) / (:C -:D))]
    donne "t :t + :h
]
fin

```

```

// Le point (f(t), g(t)) décrit la courbe quand t va se 0 à 2//

```

```

eff efftxt déroule accélère moderadian
tracer -6 6 ( -6) 6 0 6,28 0,01 // t va de 0 à 2//
modedegré

```