

Transformations du plan

1. Soient \vec{V} un vecteur. Écrire une procédure `tra(V,P)` dont le résultat est l'image de P par la translation de vecteur \vec{V} . (On représentera \vec{V} et P par des listes de deux termes correspondant à leurs coordonnées.)

réponse : `tra(V,P):=[P[1]+V[1],P[2]+V[2]] ;`

2. Soient A un point du plan et $k \in \mathbb{R}$. Écrire une procédure `hom(A,k,P)` dont le résultat est l'image de P par l'homothétie de centre A et de rapport k .

réponse : `hom(A,k,P):=[A[1]+k*(P[1]-A[1]),A[2]+k*(P[2]-A[2])] ;`

3. Soient A un point du plan et $\theta \in \mathbb{R}$. Écrire une procédure `rot(A,theta,P)` dont le résultat est l'image de P par la rotation de centre A et d'angle θ .

réponse : `rot(A,theta,P):=[A[1]+cos(theta)*(P[1]-A[1])-sin(theta)*(P[2]-A[2]),
A[2]+sin(theta)*(P[1]-A[1])+cos(theta)*(P[2]-A[2])] ;`

4. Soient D une droite d'équation $ax + by + c = 0$. Écrire une procédure `proj(D,P)` dont le résultat est la projection orthogonale de P sur D .

(On représentera D par une liste de trois termes `[a,b,c]` correspondant à l'équation $ax + by + c = 0$)

1^{ère} réponse : `proj(D,P):=block([sol],
sol:linsolve([D[1]*x+D[2]*y+D[3]=0,-D[2]*(x-P[1])+D[1]*(y-P[2])=0],[x,y]),
[rhs(sol[1]),rhs(sol[2])]) ;`

MAIS : la version 5.13 de maxima possède un « bogue » dans la fonction `linsolve`. Lorsqu'on impose au logiciel de faire tous les calculs en virgule flottante, les solutions d'un système d'équations sont remplacés par leurs parties entières. Il vaut mieux donc saisir directement la formule suivante :

2^e réponse : `proj(D,P):=[(-D[2]*(D[1]*P[2]-D[2]P[1])-D[1]D[3])/(D[1]^2+D[2]^2),
(D[1]*(D[1]*P[2]-D[2]P[1])-D[2]D[3])/(D[1]^2+D[2]^2)] ;`

5. Soient D une droite d'équation $ax + by + c = 0$ et $k \in \mathbb{R}$. Écrire une procédure `aff(D,k,P)` dont le résultat est le point Q , image de P par l'affinité de base D , de rapport k et de direction orthogonale à D . (Si H est la projection orthogonale de P sur la droite D , alors Q est donné par la relation $\overrightarrow{HQ} = k\overrightarrow{HP}$. On pourra donc utiliser les procédures `proj` et `hom` construites précédemment.)

6. On considère les deux transformations suivantes du plan :

(a) $f = f_1 \circ f_2$ où :

- i. f_1 est l'homothétie de centre $(0, 1)$ et de rapport $k = 0.95$,
- ii. f_2 est l'affinité de base Oy , de rapport -1 et de direction Ox (c'est à dire la symétrie par rapport à Oy) ;

(b) $g = g_1 \circ g_2 \circ g_3 \circ g_4$ où :

- i. g_1 est la translation de vecteur $v = (0, 1 - k) = (0, 0.05)$,
- ii. g_2 est la rotation de centre O et d'angle $\theta = 1.4$,
- iii. g_3 est l'affinité de base Ox , de rapport 0.3 et de direction Oy ,
- iv. g_4 est l'affinité de base Oy , de rapport 0.15 et de direction Ox .

Prendre le point $O = (0.0, 0.0)$ et construire une liste de points (P_0, \dots, P_{100}) telle que $P_0 = O$ et $\forall i : P_{i+1} = h_i(P_i)$ où $h_i = f$ avec une probabilité de 0.9 ou $h_i = g$ avec une probabilité de 0.1 . (On pourra faire un test comme : `if random(1.0)<0.9 then ... else ...`) (Tous les calculs seront faits en virgule flottante. Pour cela saisir la commande : `keepfloat:true;`)

Visualiser le résultat pour s'assurer que la liste obtenue est bien une liste de points en virgule flottante.

Faire un graphique avec une liste de 20000 points construits de la même manière.

Expliquer comment les propriétés de f et de g permettent de comprendre l'aspect du graphique obtenu.

7. Décrire deux applications f et g qui permettent de construire par la méthode de la question précédente le graphique suivant :

