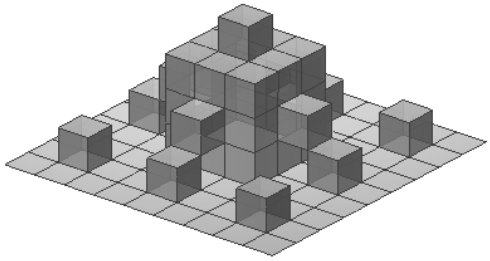
Compte rendu projet de création numérique

Leo Perrat
Abdoulahat Seck
Groupe 485

But du projet:

Réaliser une fractale : la courbe quadratique de Koch, une variante du flocon de Koch avec des angles droits et en 3D

Aperçu :



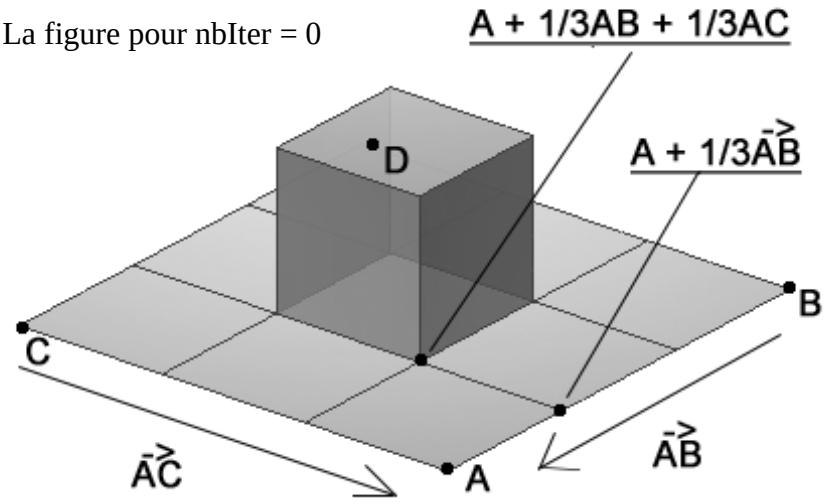
Source : https://commons.wikimedia.org/wiki/File:Quadratic_Koch_3D_(type1_stage2).png
Auteur : Robert Dickau Droits : CC-BY-SA-3.0

Analyse:

C'est une fractale donc il faudra utiliser la récursivité pour dessiner la figure

La variable nbIter sera le nombre d'itérations

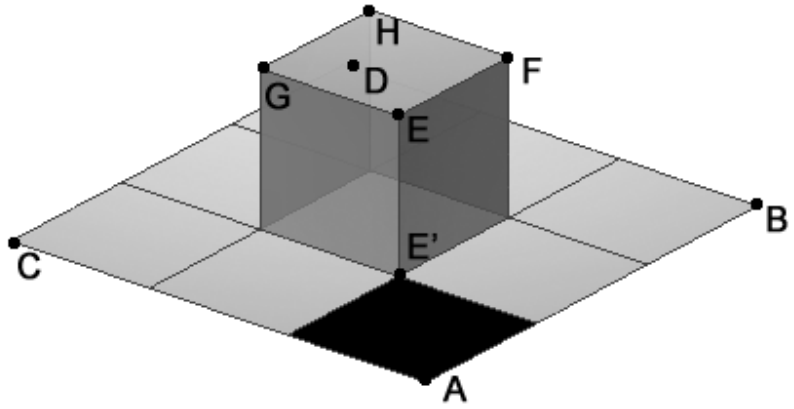
La figure pour nbIter = 0



Original : https://commons.wikimedia.org/wiki/File:Quadratic_Koch_3D_(type1_stage1).png
Auteur : Robert Dickau Droits : CC-BY-SA-3.0

Pour pouvoir dessiner un polygone sur processing il faut connaître les points de chaque sommet. Dans le cas nbIter = 0 on doit dessiner 13 carrés dont on va exprimer les points par des vecteurs, dans le cas récursif on fera un appel récursif pour chaque carré.

Sur l’image ci-dessus A,B,C et D sont les points rentrés dans la fonction tout les autres points seronts composés des vecteurs AB et AC



Par exemple, si on est dans le cas nbIter = 0 pour le carré noir on va dessiner un polygone avec commes points : A, A+1/3AB, A+1/3AB+1/3AC,A+1/3AC
dans le cas nbIter > 0 on fait un appel récursif de la fonction avec A=A, B=A+1/3AB, C=A+1/3AC, D = A+1/3AB+1/3AC

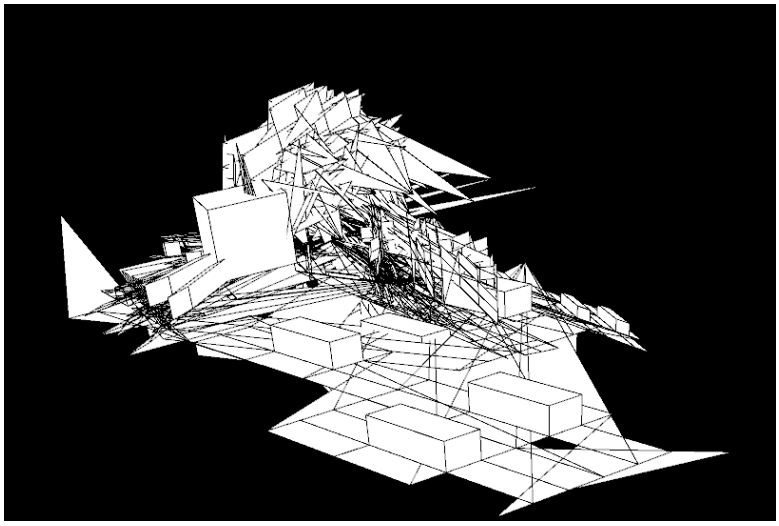
Pour exprimer les points E,F,G et H aux sommets du cube au centre, il faut utiliser le produit vectoriel.
Le produit vectoriel AB^AC donne un vecteur V orthogonal à AB et AC
V n’est pas totalement égal au vecteur E’E car ||V|| = ||AB|| * ||AC|| *|sin(AB,AC)| = ||AB|| * ||AC|| (AB et AC sont orthogonaux donc sin(AB,AC) = 1)

E’E = (V / ||V||) * 1/3||AB||

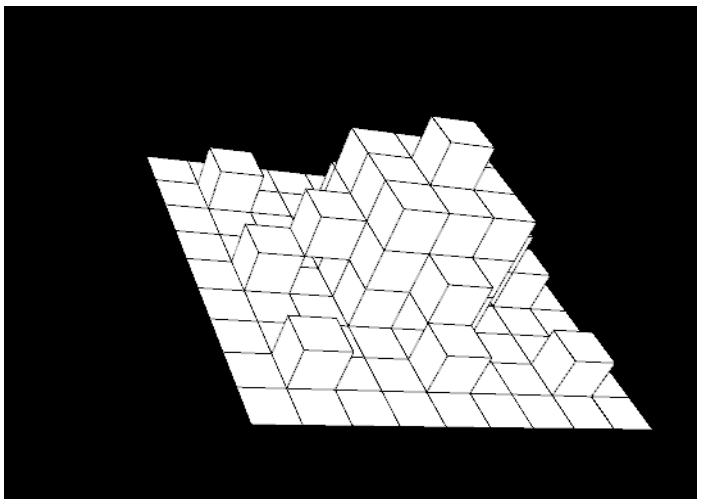
E = A + 1/3AB + 1/3AC + E’E

On peut désormais exprimer tout les points du polygone.
On créera une fonction pour le produit vectoriel

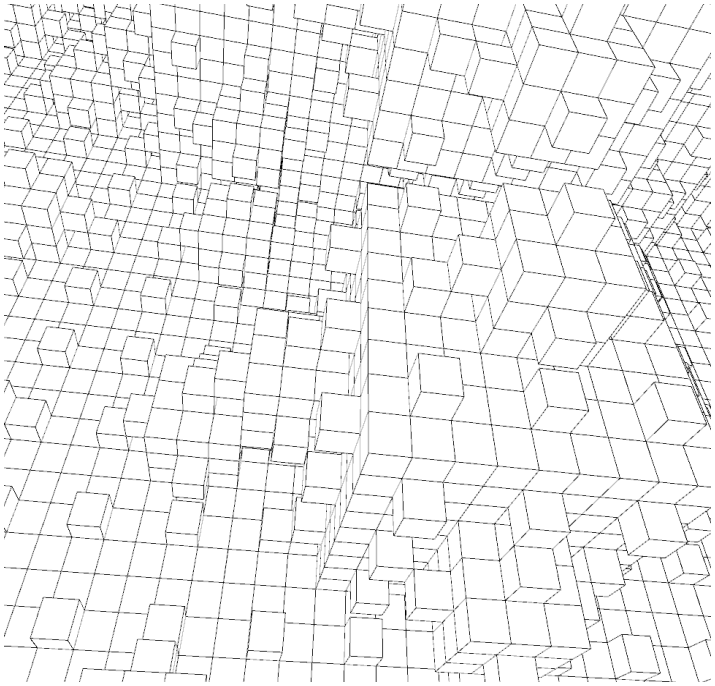
<u>Captures</u> <u>d’ecran:</u>



Premier essai d’execution
Les erreurs sont dues au variables qui étaient passé en référence



Resultat pour kochQuadra(A,B,C,D,1)

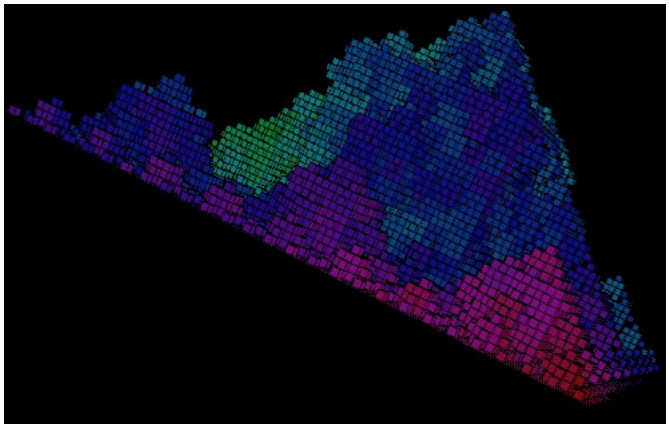


Resultat pour kochQuadra(A,B,C,D,3)

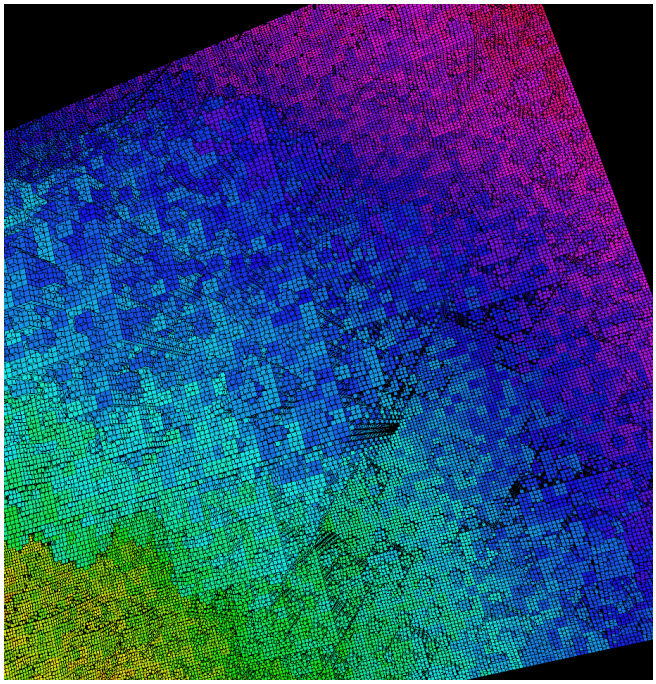
Couleurs:

Ensuite nous avons voulu ajouter des couleurs.
Nous avons voulu faire un dégradé de couleurs du point A au point D, pour cela nous traitons les couleurs en TSL (teinte saturation luminance).
A,B,C et D sont des variables globales dans le programme et polyA,polyB,polyC et polyD les variables de la fonction kochQuadra.
La teinte pour chaque polygone est égale à (||polyAD|| / ||AD||) *100

Pour éviter le clignotement on utilise une graine, qui va produire les memes nombres pseudo aléatoires à chaque execution du draw()



Resultat avec couleurs pour kochQuadra(A,B,C,D,3)



Resultat avec couleurs pour kochQuadra(A,B,C,D,4)

Pour finir nous avons mis des controles claviers pour modifier certains parametres comme le point A (touches Z,Q,S et D), ce qui déforme le polygone, augmenter et diminuer la récursivité (touches R et F) et changer la graine (touche Espace)