



*Recode Project*

# *Sommaire*

---

Avant-propos

1. Présentation de l'artiste

2. Son oeuvre

3. Inspiration & appropriation

Mélusine Vilars  
MI CMM  
2016-2017

# Avant-propos

---

Le «Computer Art» (ou art numérique) est un art dans lequel les ordinateurs jouent un rôle dans la production ou l'affichage de l'oeuvre d'art. L'oeuvre peut être une image, un son, une animation, une vidéo, un jeu vidéo, un site web, un algorithme, une performance ou encore une installation. De nombreuses disciplines traditionnelles intègrent maintenant les technologies numériques ; la séparation entre les oeuvres traditionnelles et les oeuvres numériques est alors de plus en plus floue.

ReCode Project est un projet communautaire visant à préserver le Computer Art en traduisant d'anciens codes en un langage de programmation moderne : le Processing. Ce langage a été créé par deux américains en 2001 dans le but de réaliser des créations graphiques interactives. Cette bibliothèque Java existe également en javascript sous le nom de p5.js, ce qui permet de l'utiliser dans un environnement HTML5.

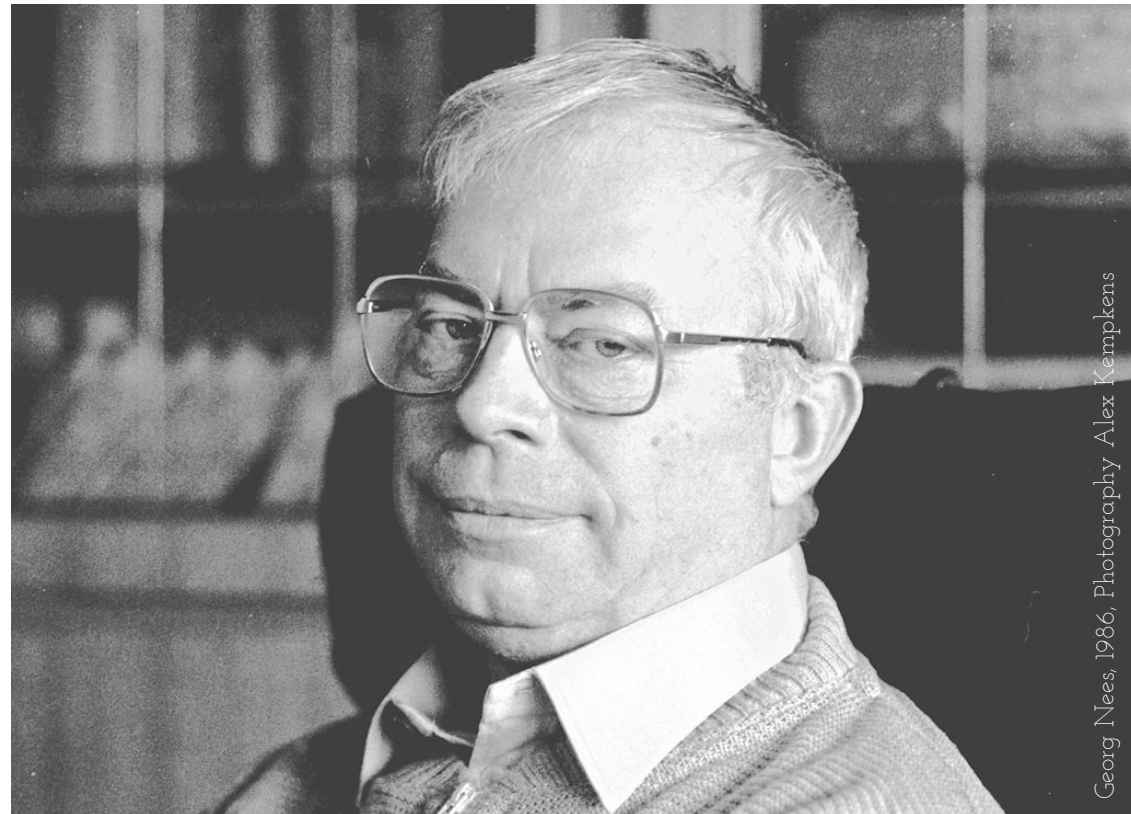
Les codes actualisés sont disponibles au public. Ce projet permet donc de remettre au goût du jour les travaux des pionniers du Computer Art, de créer une communauté active mais aussi d'offrir une ressource d'apprentissage aux praticiens et aux enseignants.

# 1. Georg Nees

---

Georg Nees, né en 1926 et mort en 2016, était un universitaire allemand, pionnier du Computer Art depuis le milieu des années 1960. Il a débuté en tant qu'ingénieur et a pu accéder aux débuts des technologies informatiques en commençant à programmer sur des ordinateurs dès 1959. Il a par la suite commencé à utiliser son passé d'ingénieur pour créer de l'art : ce fut alors une révolution. Son travail est particu-

lièrement remarquable car il a posé les bases d'un nouveau mouvement dans le monde de l'art, initié par des ingénieurs. Des questions se sont alors rapidement posées : est-ce vraiment de l'art, ou est-ce juste un dessin technique, mathématique ? Mais l'intention de Nees était bien de faire de l'art.



Georg Nees, 1986, Photography Alex Kempkens

## 2. Son oeuvre

---

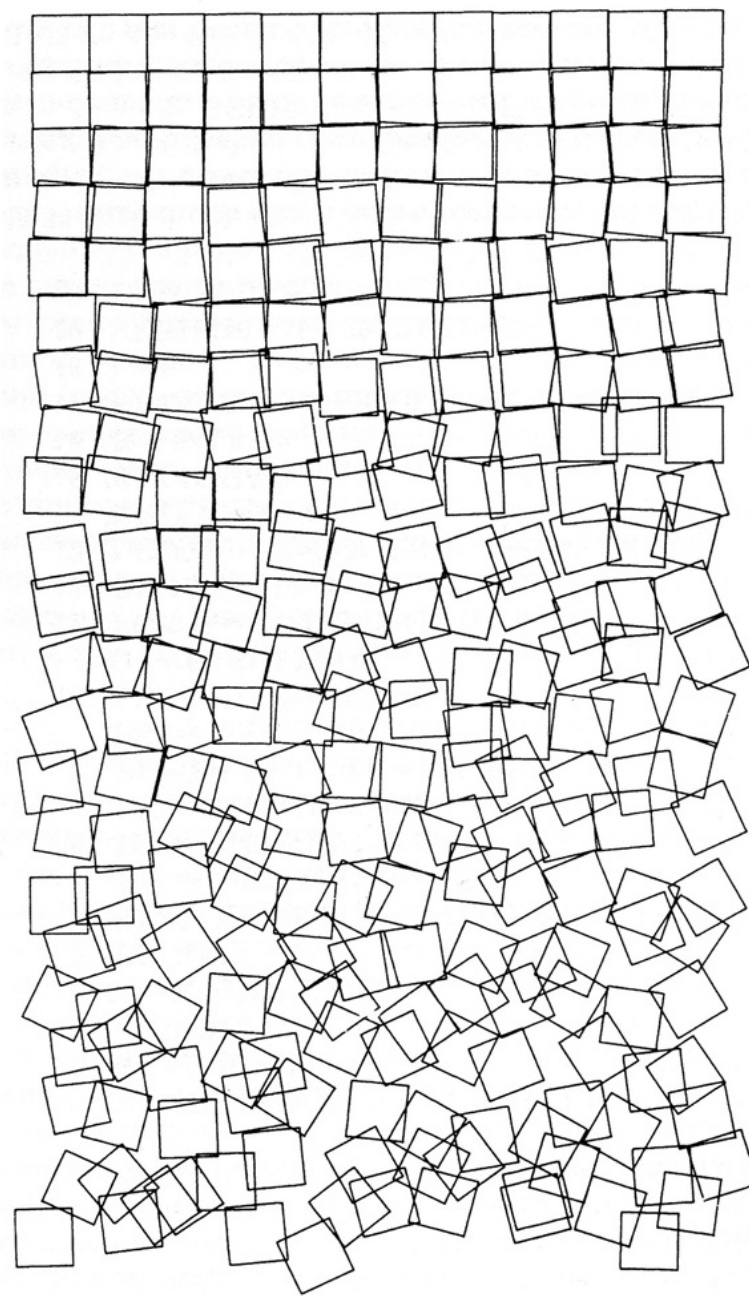
Avec un certain nombre d'autres praticiens travaillant à cette époque, Nees s'intéresse à la relation entre l'ordre et le désordre dans la composition de l'image.

«Schotter» est un graphique informatique réalisé en 1968 à l'aide du Siemens System 2002. Ce graphique montre visuellement la relation entre l'ordre et le désordre : Nees est parvenu à montrer qu'ils peuvent être reproduit grâce à une programmation précise et mathématique.

Malgré la nature chaotique de l'image, le code qui se cache derrière doit être exact et très précis pour arriver à un tel résultat. L'introduction de variables aléatoires dans le programme informatique a amené les carrés ordonnés à descendre dans le chaos. Son programme trace des carrés de côté de longueur constante, mais à des endroits aléatoires et à des angles différents.

Cette oeuvre est l'un des premiers exemples d'art génératif qui montre aussi clairement le lien entre le programme et l'image : le code seul ne peut pas être considéré comme de l'art et l'image elle-même perd beaucoup de valeur sans son code.

Cette oeuvre célèbre peut être observée au Victoria and Albert Museum (V&A) à Londres.





# 3. Inspiration & appropriation

William J. Kolomyjec a été l'un des artistes précurseur dans l'étude des pionniers de l'art informatique. Il a étudié à la Michigan State University à East Lansing Michigan et a été l'un des premiers à obtenir son diplôme dans le domaine du Computer Art.

Pour lui, le Computer Art façonne la société et rompt les barrières entre l'art et la culture d'une part, et la science et la technologie d'autre part. Il prévoyait que l'ordinateur serait l'outil créatif ultime pour l'artiste-scientifique.

Dans les années 1970, il a sélectionné l'oeuvre de Georg Nees, *Schotter*, pour en faire une adaptation : «je pense qu'il convient d'étudier les oeuvres d'autres artistes, en particulier les artistes traditionnels et ceux qui ont établi l'art informatique dans le domaine des beaux-arts».

C'est à partir de cette idée que j'ai choisi de retravailler l'un des codes de Kolomyjec, datant de 1978, *Hex Variation*. Après avoir choisi ce code Processing sur le site de ReCode Project, je l'ai transformé en p5.js pour

pouvoir l'utiliser plus facilement. Ce code se base également sur des formes géométriques qui se placent de manière aléatoire. Mon objectif était donc de moderniser ce code et ses formes et d'y d'ajouter un peu de désordre.

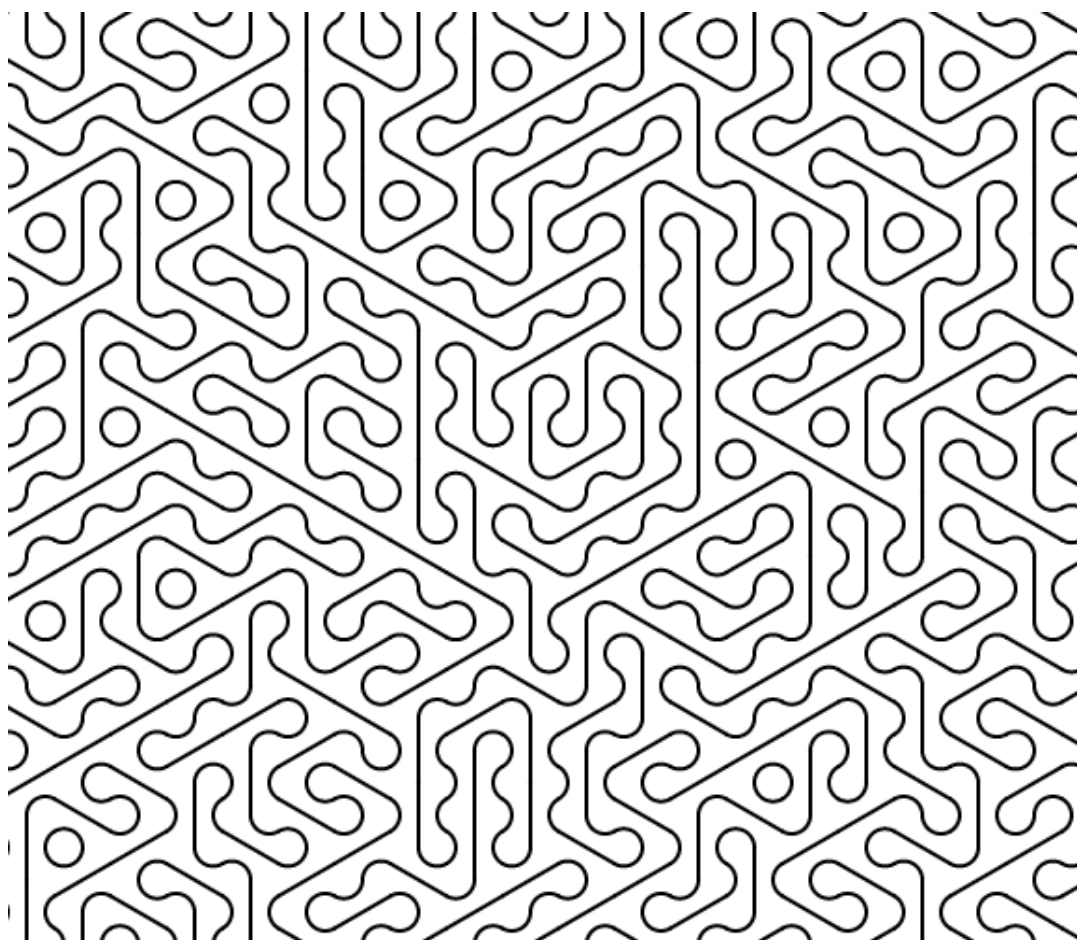
Pour ce faire, j'ai choisi d'y ajouter des couleurs vives, contrastées, et parfois de casser la continuité de ces lignes pour que le résultat soit plus dynamique, moins monotone. En jouant sur l'épaisseur des contours on parvient également à créer un univers très différent.

J'ai choisi de mettre en avant les neufs versions réalisées en les regroupant par trois et en les superposant. Les deux premières images sont imprimées sur du papier transparent et la dernière sur du papier blanc, pour

faire ressortir l'ensemble et le séparer des deux autres groupes. De cette façon, les images forment une composition à part entière. Chaque triptyque représente une ambiance : Tropical, Rêverie et Hiver.

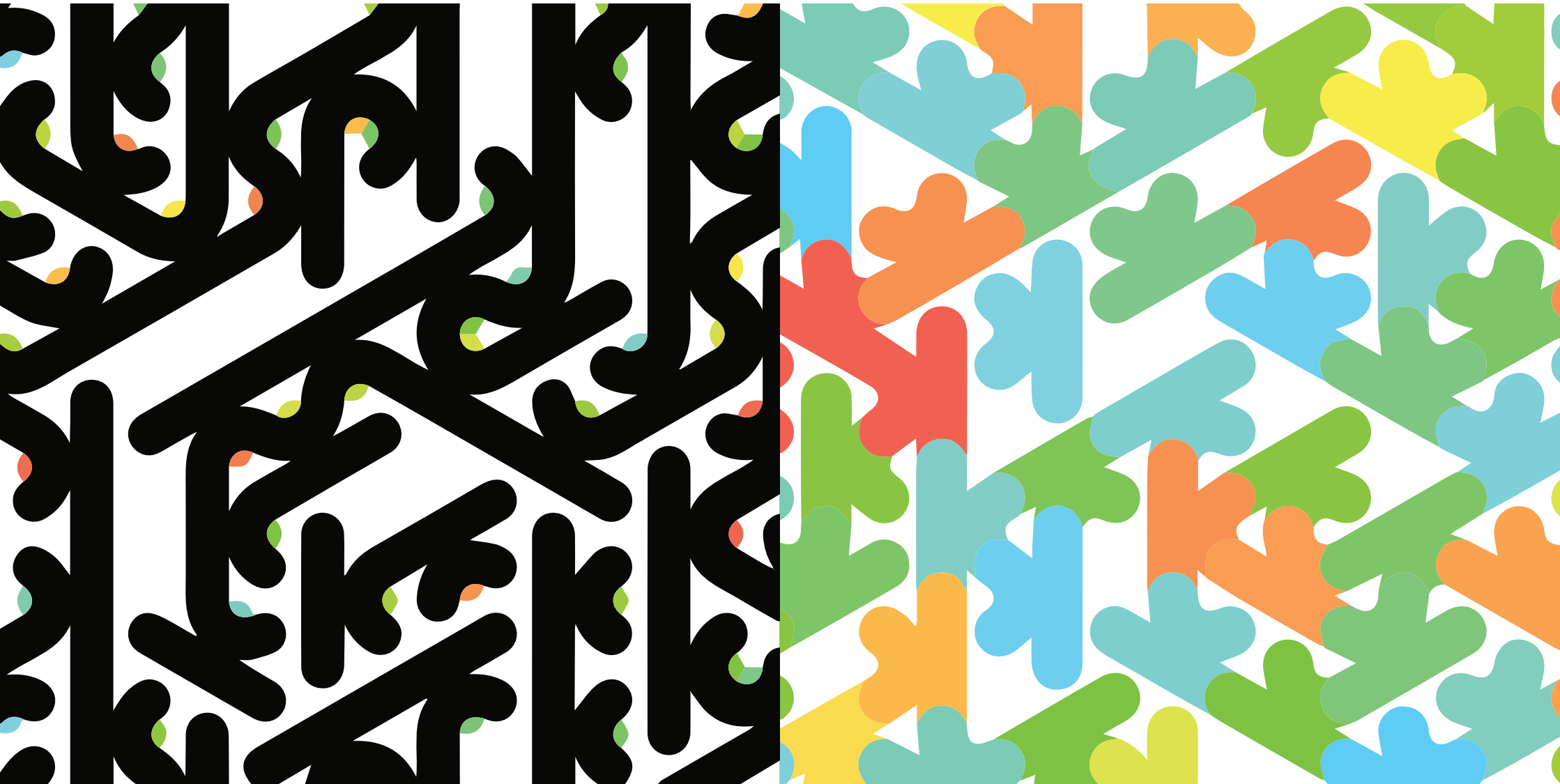
J'ai choisi un format carré et une reliure japonaise avec le titre de l'édition. Le livret s'ouvre vers le haut pour permettre de mieux voir chaque image séparément mais également pour mettre en avant le jeu de lumière à travers le papier transparent.

À partir du code d'origine, j'ai effectué un certain nombre de modifications pour parvenir au résultat escompté. Dans un premier temps, j'ai choisi de rendre le code responsive, pour qu'il s'adapte lorsque l'on réduit la taille de l'écran. J'ai ensuite choisi de modifier la couleur des formes, ainsi que la couleur et la taille des bordures à l'intérieur de la fonction `draw()` pour pouvoir obtenir des couleurs différentes sur chacun des éléments en ajoutant de l'aléatoire. Pour modifier les couleurs facilement j'ai choisi de passer en mode HSB (ou HSL), ce qui permet de modifier indépendamment la teinte, la saturation et la luminosité. Enfin, j'ai modifié par moment la forme des arcs pour obtenir des formes plus irrégulières.



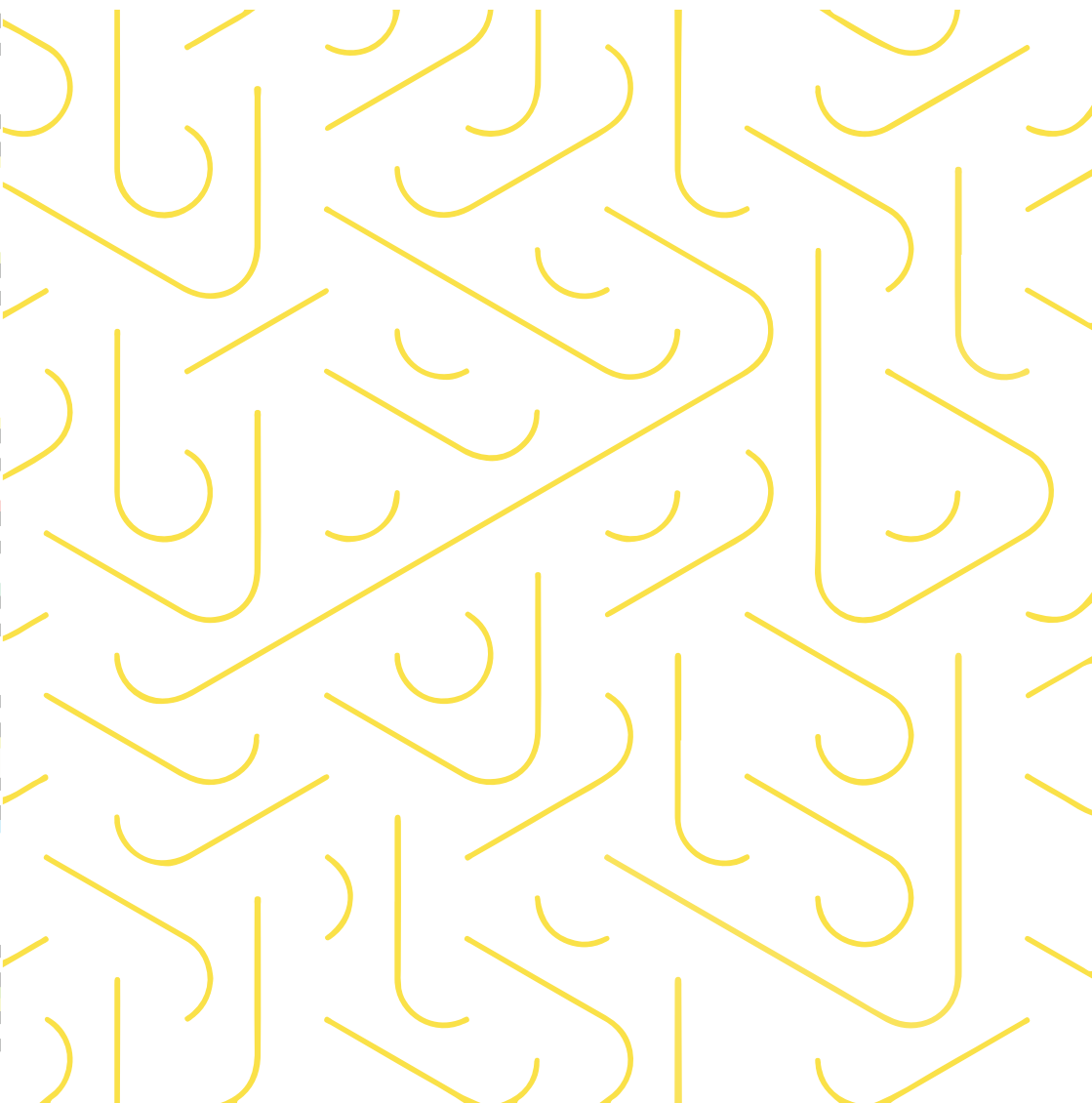
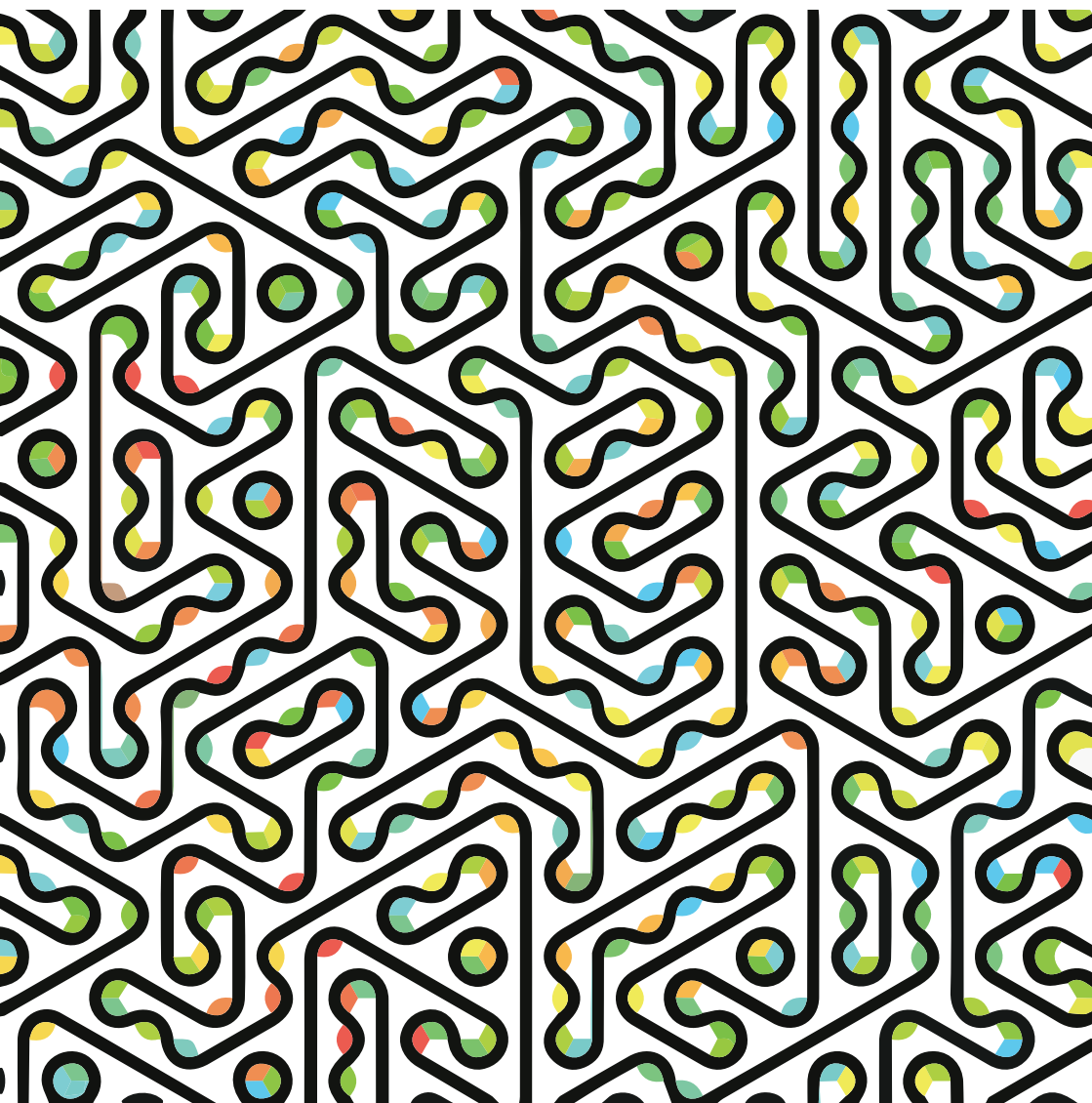
*Tropical : 1*

*Tropical : 2*



*Tropical : 3*

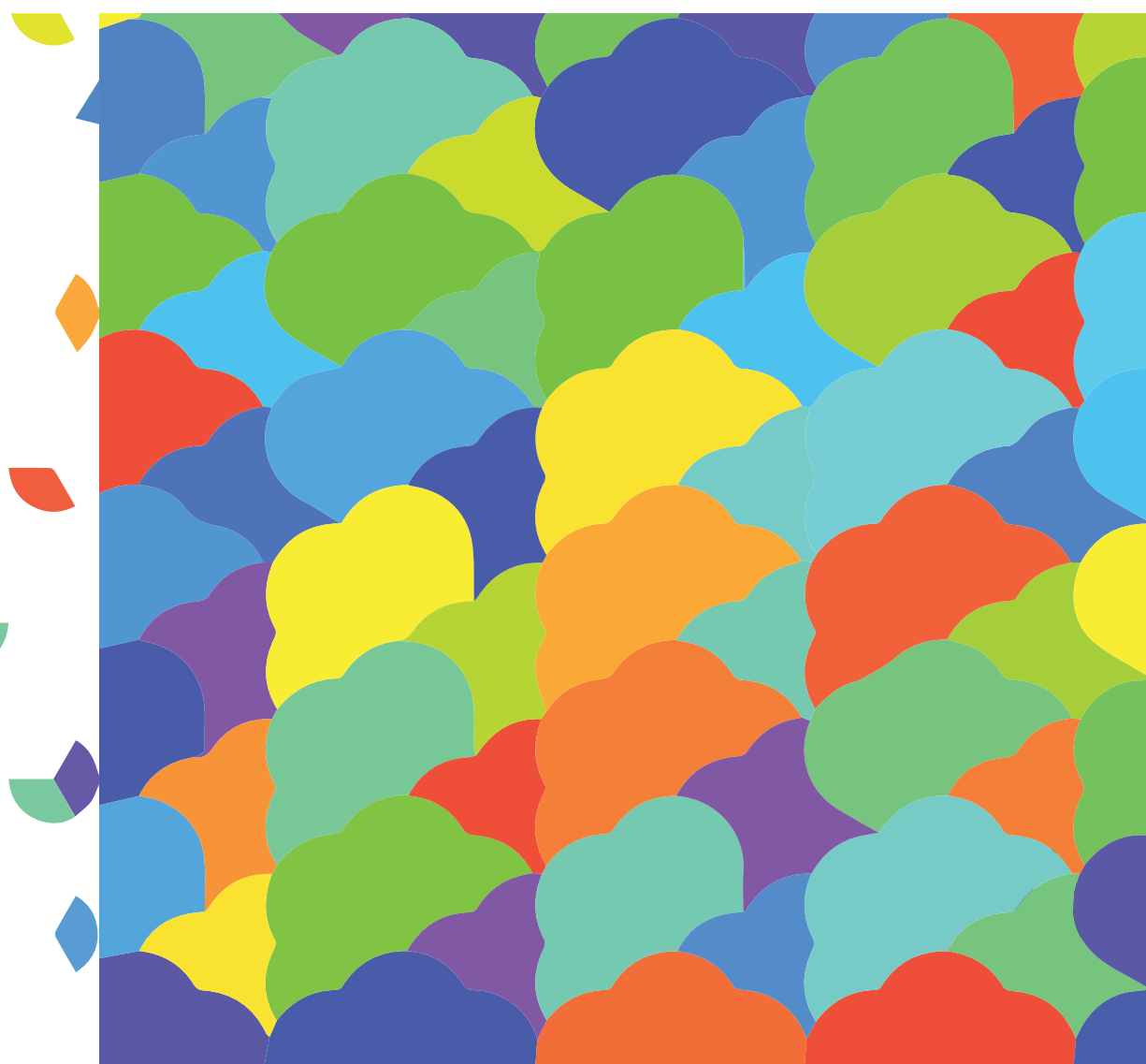
*Rêverie : 1*



*Rêverie : 2*



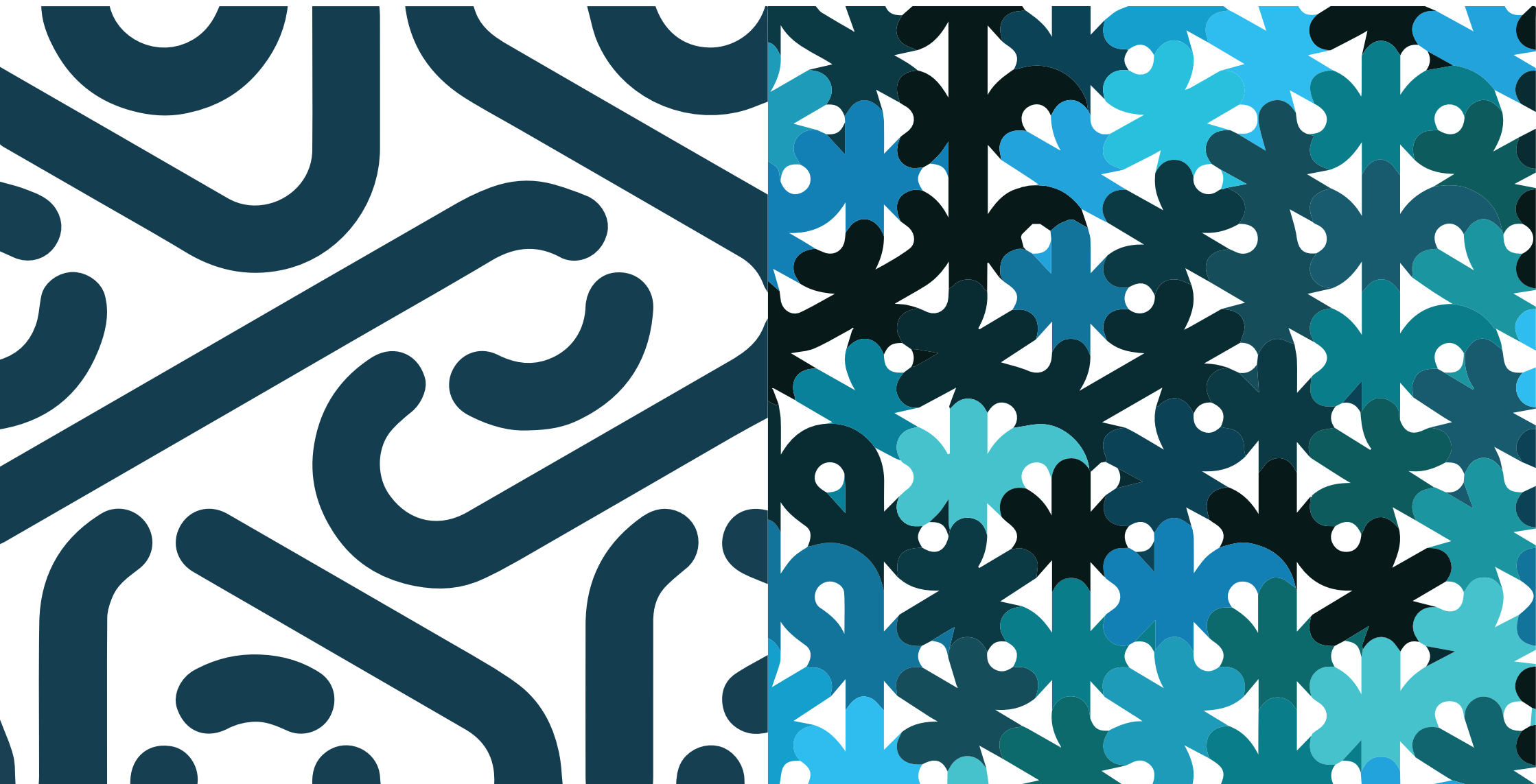
*Rêverie : 3*





*Hiver : 1*

*Hiver : 2*



# *Hiver : 3*

