

LIFAMI  
Contrôle Continu Terminal (Durée : 1h30)  
Vendredi 15 mai 2020

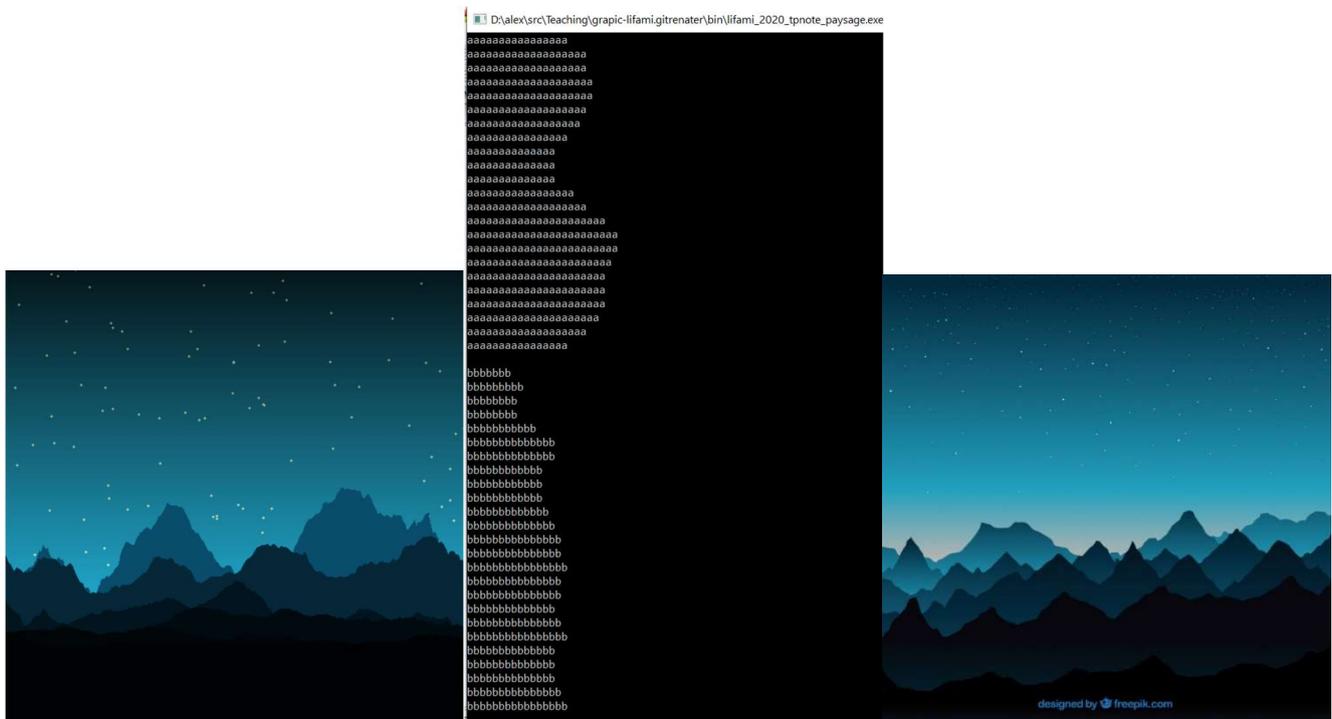


Figure 1. A gauche le résultat obtenu avec Graptic, au milieu en mode texte. A droite, une illustration d'artiste qui a inspiré ce sujet.

L'objectif de ce sujet est de construire et d'afficher un paysage comme ci-dessus, soit avec Graptic (gauche), soit en mode texte (milieu). Ce paysage est composé d'une superposition de plusieurs plans appelés *Montagne* dans ce sujet. En mode texte, il faut imaginer une rotation de  $90^\circ$  sur la gauche de l'image car il est plus commode d'afficher les hauteurs sous forme de lignes. De plus, les différents plans sont affichés l'un en dessous de l'autre, il ne sera pas demandé de les superposer. Une montagne est représentée par une couleur et une carte d'altitude (carte de hauteurs) à une dimension. Une carte d'altitude est une fonction définie sur l'intervalle  $[0,1]$  renvoyant l'altitude. Cette fonction d'altitude est stockée de manière discrète dans un tableau de taille variable. Pour construire une montagne stockée dans sa carte d'altitude, différentes résolutions de fonction d'altitude sont sommées. Chacune de ces fonctions d'altitudes sont construites en remplissant le tableau de valeurs aléatoires (ou bruit). Ce principe de construction par somme de bruit aléatoire est appelé construction par Bruit de Perlin.

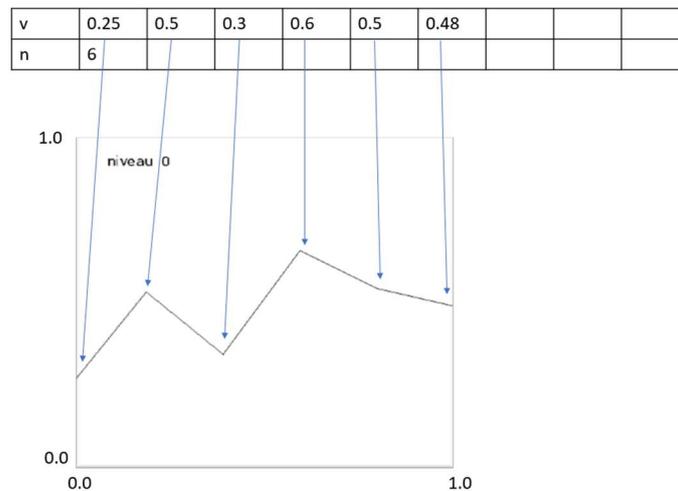
Quatre structures différentes sont manipulées dans ce sujet.

- *Couleur* comportant soit 3 entier r,g,b en mode Graptic; soit un caractère en mode texte.
- *Altitude* comportant un tableau de valeurs réelles, et un entier indiquant combien de valeurs sont utilisées.
- *Montagne* comportant un champ Altitude et une couleur.
- *Monde* comportant un tableau de Montagne, et un entier indiquant combien de Montagne sont utilisées.

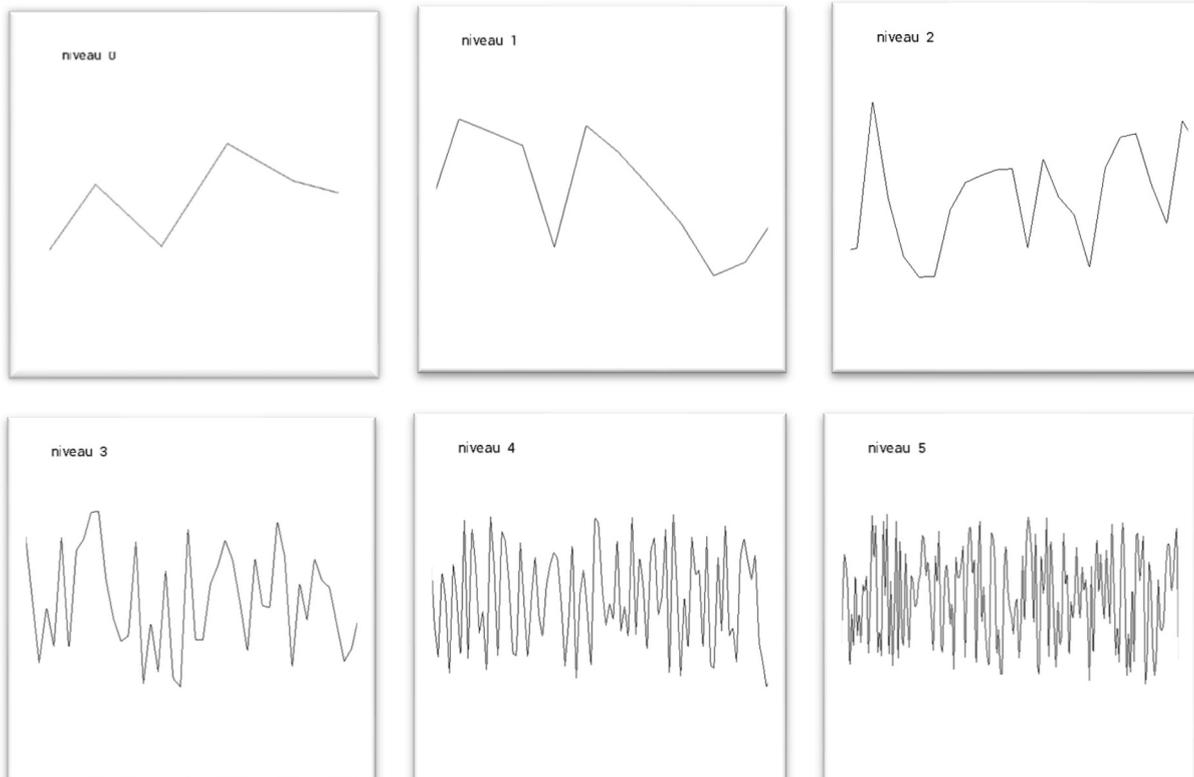
Les questions sont détaillées directement dans le code initial. Ce document présente les concepts nécessaires à la compréhension du sujet.

## Altitude : carte de hauteur

La structure *Altitude* comporte un tableau de valeurs réelles, et un entier indiquant combien de valeurs sont utilisées. La fonction *altitudeEval* interpole entre les cases du tableau pour donner une valeur d'altitude de manière continue entre les abscisses 0 et 1.0, comme illustré sur la figure ci-dessous :



Les images ci-dessous montrent 6 cartes de hauteurs avec différents nombres de valeurs stockées dans le tableau. Pour les 6 niveaux, les nombre de valeurs sont respectivement : 6, 12, 24, 48, 96, 192 valeurs. Ces 6 cartes de hauteurs sont initialisées avec des altitudes(=hauteurs) aléatoires compris entre 0 et 1. Les fonctions *operator+* et *operator\** permettront respectivement d'additionner deux cartes de hauteurs et de multiplier par un réel une carte de hauteur. Les valeurs de hauteurs ne resteront pas forcément entre 0 et 1.



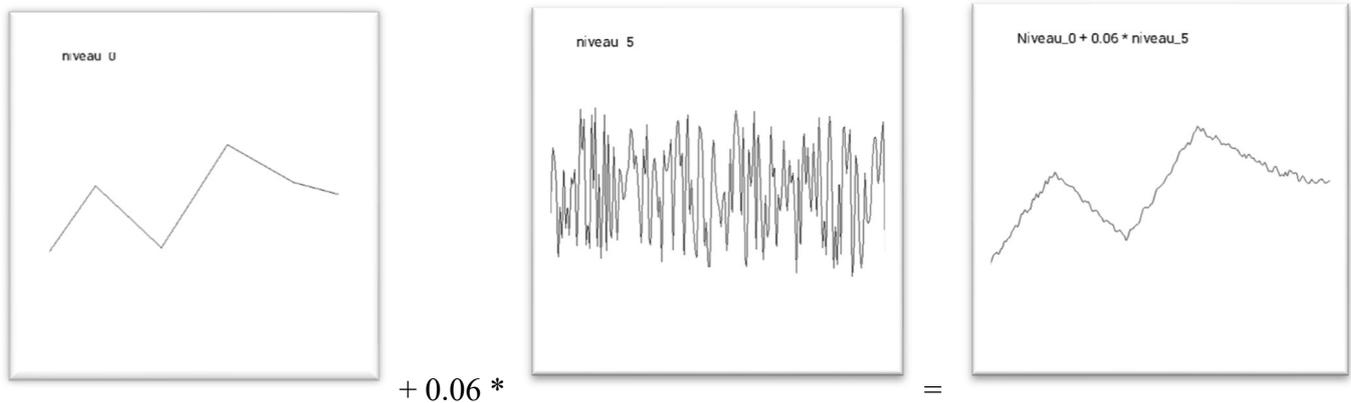
Le niveau 0 avec ses 6 valeurs peut représenter la silhouette principale de la montagne (ordre de grandeur de plusieurs kilomètres). Le niveau 5 avec ses 192 valeurs peut représenter les petites variations à la surface de la montagne (ordre de grandeur du mètre). Les niveaux intermédiaires sont les détails avec différents ordres de grandeurs (5km, 1km, 100m, 10m).

### Explication pour la question 3

Pour passer d'une abscisse entre 0 et 1 à une valeur d'altitude, il faut trouver le numéro de case correspondant et interpoler. La fonction *altitudeEval* part d'un réel  $x$  dans l'intervalle  $[0,1]$ , calcule le numéro de case  $n\_case$  en multipliant  $x$  à la taille du tableau : la partie entière de  $n\_case$  est la case du tableau à interpoler avec la suivante ; le coefficient d'interpolation est la partie décimale de  $n\_case$ .

### Explication pour la question 9

Pour construire une montagne, il faut sommer les 6 niveaux en les multipliant par un coefficient qui correspond à leur ordre de grandeur. Une somme simple ajoutant les détails du niveau 5 au niveau 0 donne ceci :



Chaque image est l'affichage d'une structure *Altitude*. Les opérations sont réalisées avec la fonction *operator+* qui additionne deux structures *Altitude* (question 5) et avec la fonction *operator\** qui multiplie un réel à une structure *Altitude* (question 6).

On peut intégrer la taille de la fenêtre dans le calcul pour faciliter son affichage. Voici le calcul qui intègre tous les niveaux et suppose un affichage dans une fenêtre de 500x500. Ce calcul prend 256 pour le niveau 0 car 256 représente environ la moitié de l'écran et est facilement divisible par 2.

$$\text{altitude} = 256 \times \text{niveau}[0] + 128 \times \text{niveau}[1] + 64 \times \text{niveau}[2] + 32 \times \text{niveau}[3] + 16 \times \text{niveau}[4] + 8 \times \text{niveau}[5]$$

On retrouve la silhouette avec deux bosses du niveau 0, additionner de toutes les autres résolutions.

