

HUNOT-MARTIN
Alaric
M1 Imagine
Université Montpellier
26/09/23

Compte Rendu TP2 HAI714

Modélisation 3D : maillage sphère et plan

Introduction:

Pendant ce TP, nous avons appris à créer des maillages 3D et nous nous sommes familiarisés avec la programmation sous OpenGL en facetisant une sphère et un plan en 3D. Cela impliquait la création de sommets et de triangles pour constituer les modèles 3D, ainsi que la modélisation des normales aux sommets et la colorisation de ces derniers.

Exercice 1 : Setunitsphere

Résolution de l'exercice :

-écrire les angles θ et ϕ pour chaque sommet de la double boucle for:

$\theta = i * 2 * (\pi / nX)$

$\phi = (-\pi / 2) + j * (\pi / (nY - 1))$

-on définit chaque sommet à partir des équations de la forme paramétrique de la sphère et des angles θ et ϕ

-on crée une fonction normalize de normalisation des vecteurs

-on ajoute chaque sommet au mesh (`o_mesh.vertices.push_back(Vec3(x,y,z))`), on ajoute chaque normale au sommet par la fonction normalize (`o_mesh.normals.push_back(normalize(Vec3(x,y,z)))`) et on ajoute une couleur aléatoire par défaut pour chaque sommet(`o_mesh.colors.push_back(...)`)

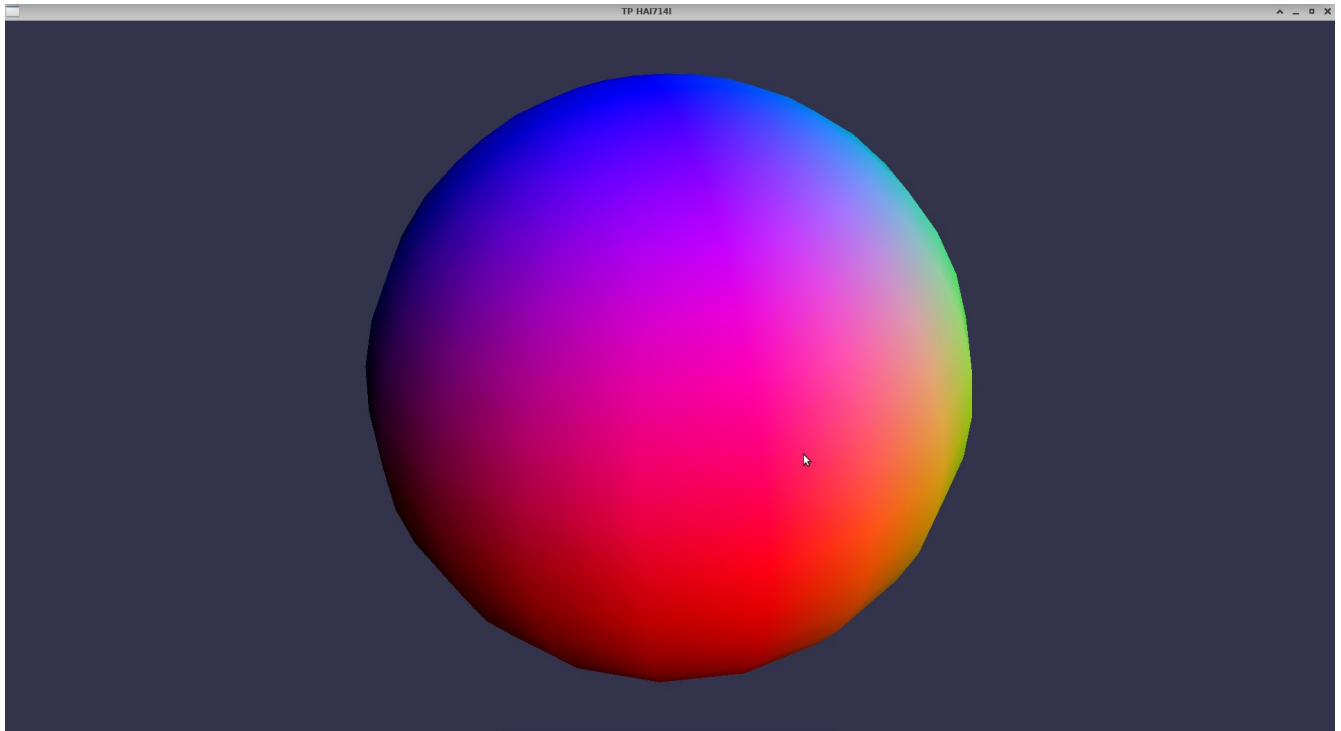
-on crée une fonction idx pour aider à la définition des triangles.

-on ajoute les triangles par la fonction idx (`o_mesh.triangles.(Triangle(idx(...),idx(...),idx(...)))`);

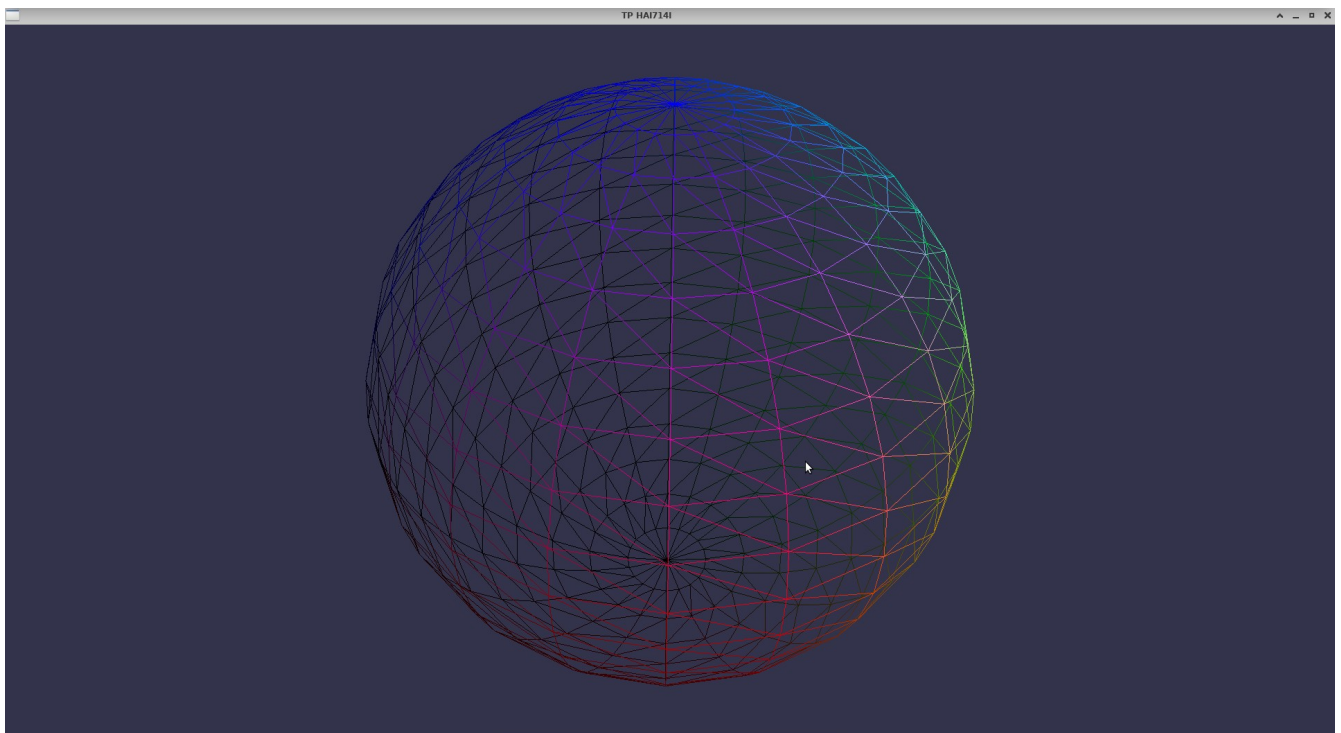
-on crée 2 variables globales `sx` et `sy`

-on ajoute dans la fonction key 2 cases '+' et '-' qui incrémente et décrémente `sx` et `sy` et appelle la fonction `setunitsphere`

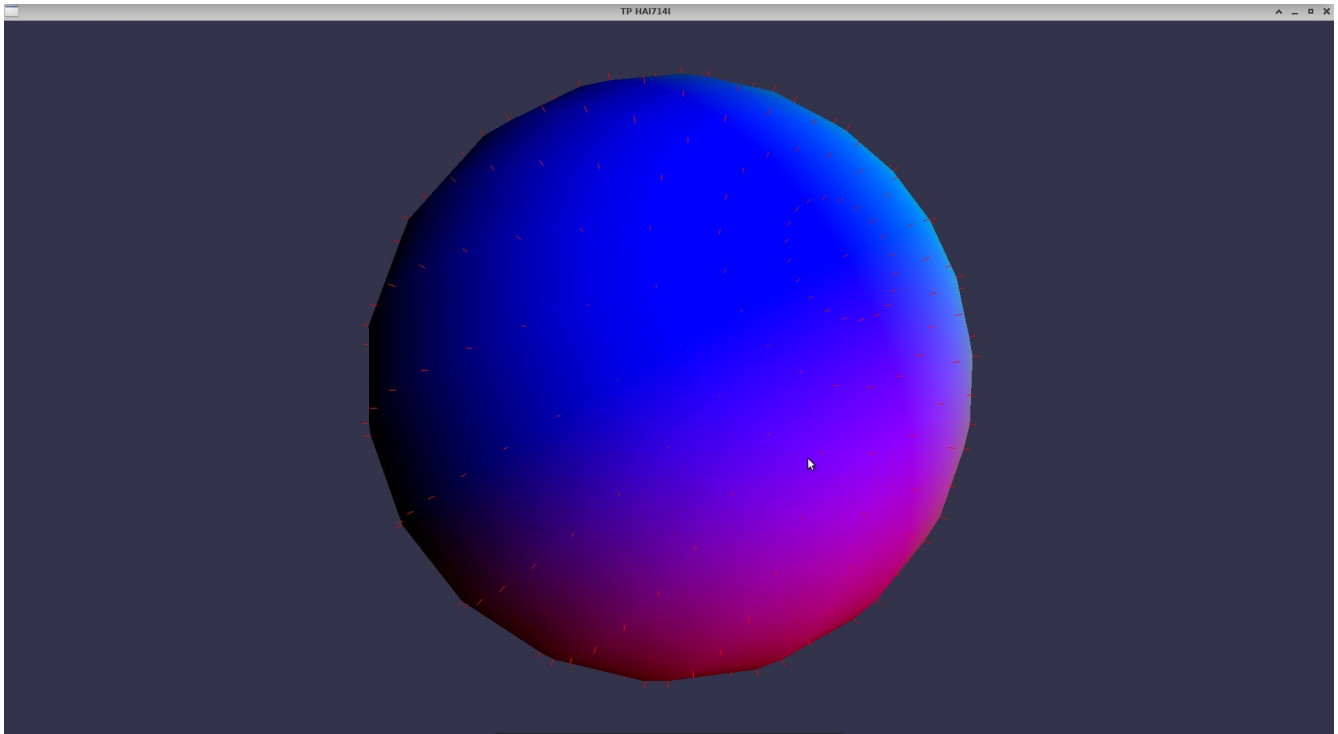
Rendu :



sphère maillage



mesh sphère



sphère+normales

Exercice 2 : SetTesselatedSquare

Résolution de l'exercice :

-on crée notre plan aléatoire de la manière suivante dans une double boucle for:

- * on fixe $x = (i / (nX - 1))$ et $y = (j / (nY - 1))$
- *on crée deux variable r1 et r2 qui stocke 2 nombres réels aléatoires entre -1 et 1
- *on utilise la méthode getnoise avec les bornes $\min(r1, r2)$ et $\max(r1, r2)$
- *on ajoute les sommets, les normales et les triangles au mesh toujours avec l'aide de idx

note: je me suis amusé si on voulait seulement résoudre l'exercice il aurait fallu juste utiliser la méthode getnoise sur l'intervalle $[-1;1]$, on obtient le rendu de la première image.

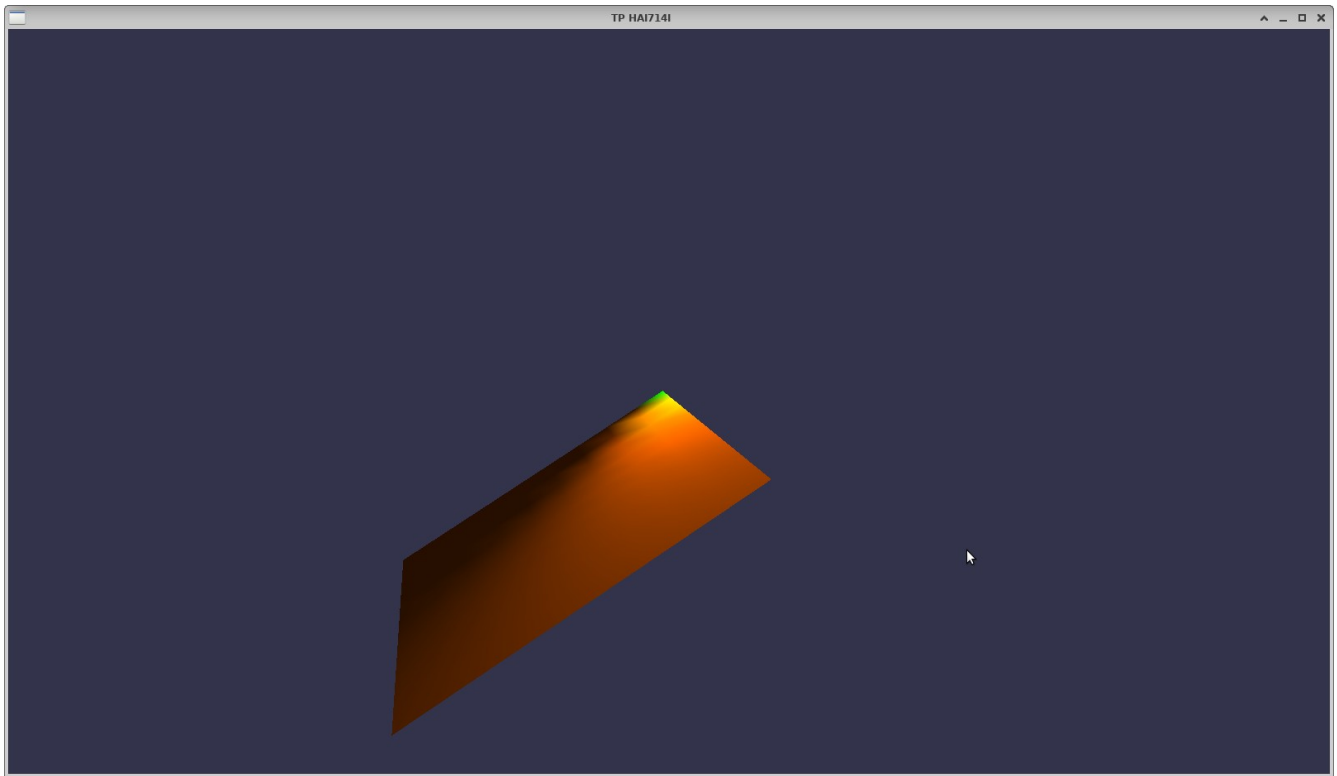
-on ajoute une couleur au sommet en fonction de si la normale est verticale (vert) ou non (brun) ou on crée une variable qui vérifie si le sommet a pour normal(0,0,0) si oui il le colore en vert sinon en brun

-pour gérer le bruit :

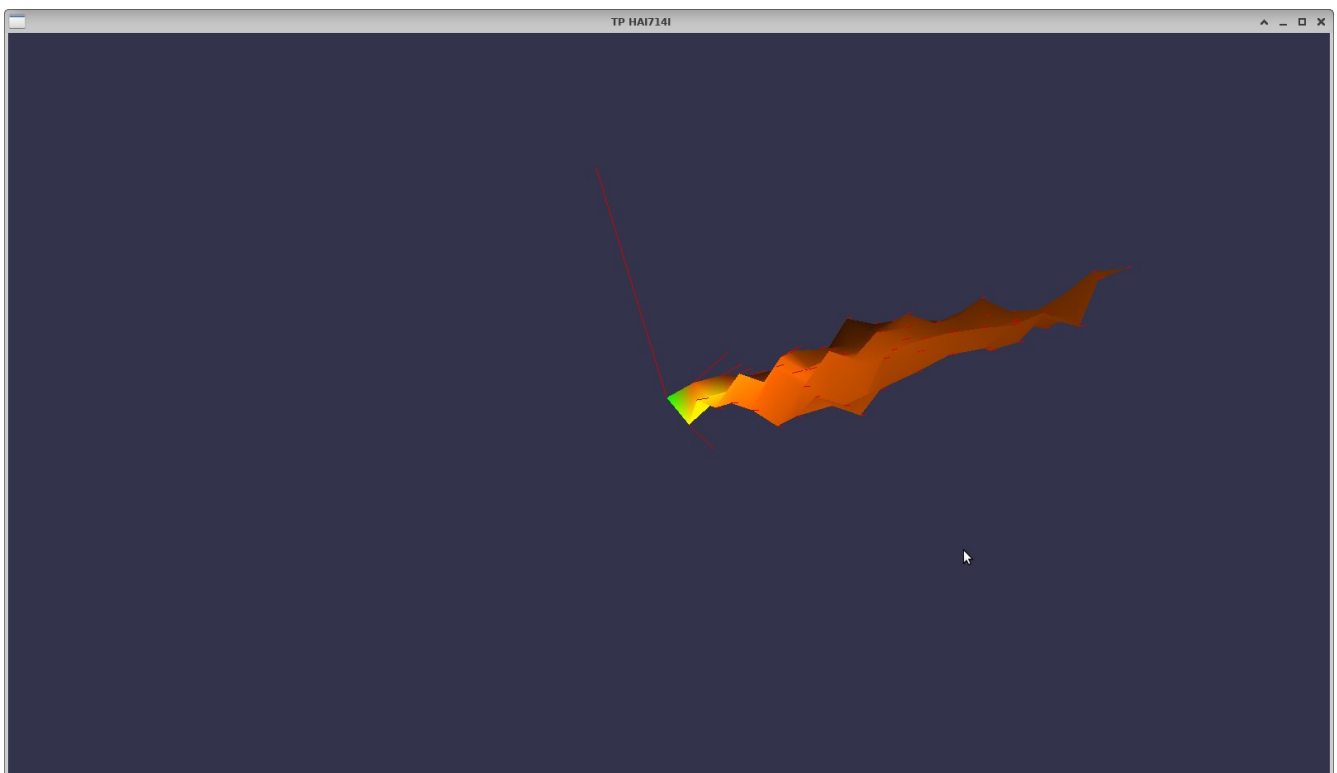
- *on crée 2 variables globales offSetX et offSetY
- *on ajoute offSetX et offSetY à x et y

*on ajoute les cases a d z s qui incrémentent ou décrémentent offsetX ou offSetY

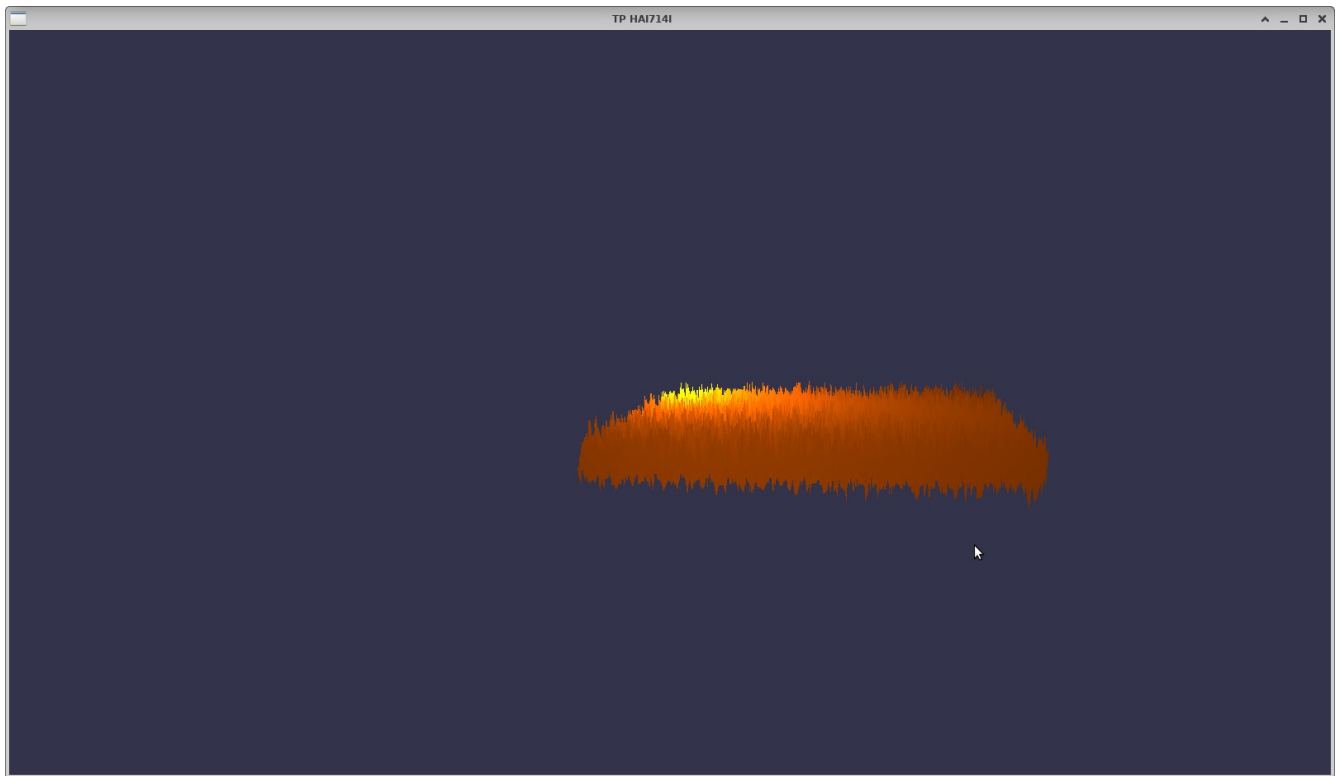
Rendu :



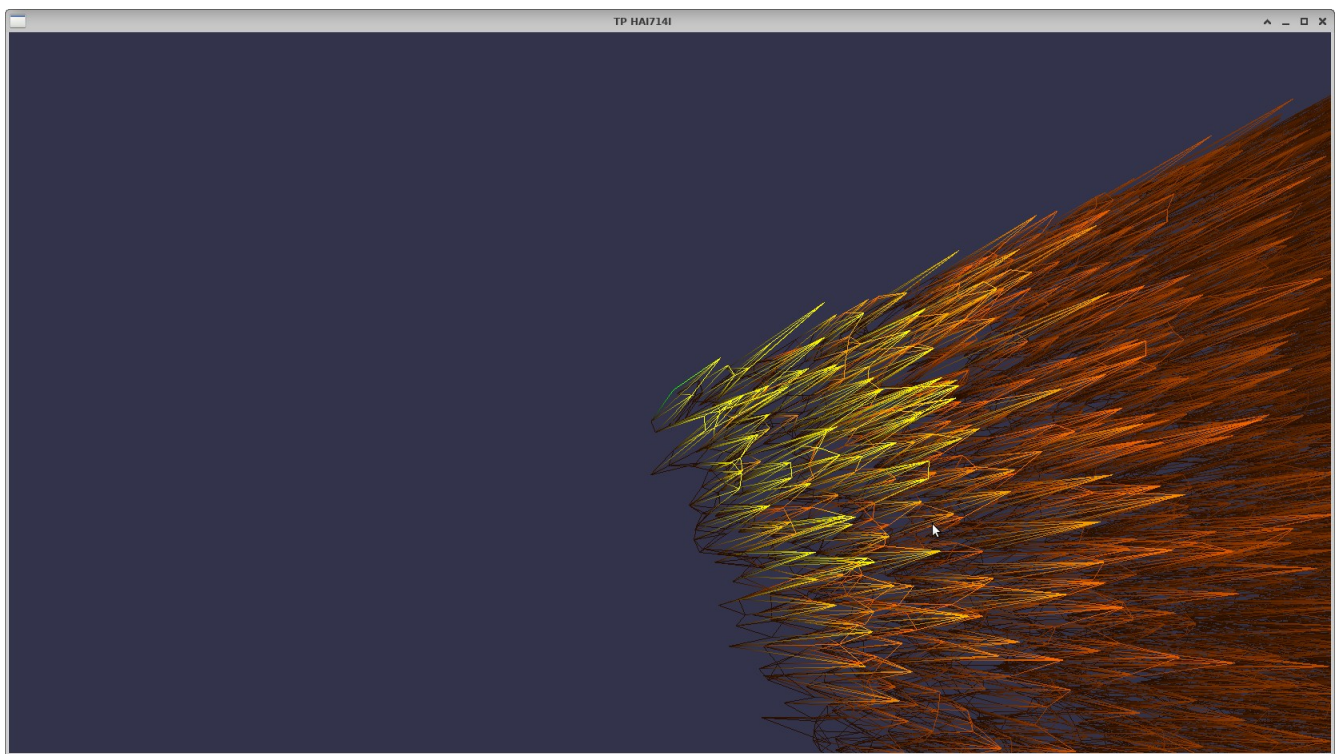
plan sans relief



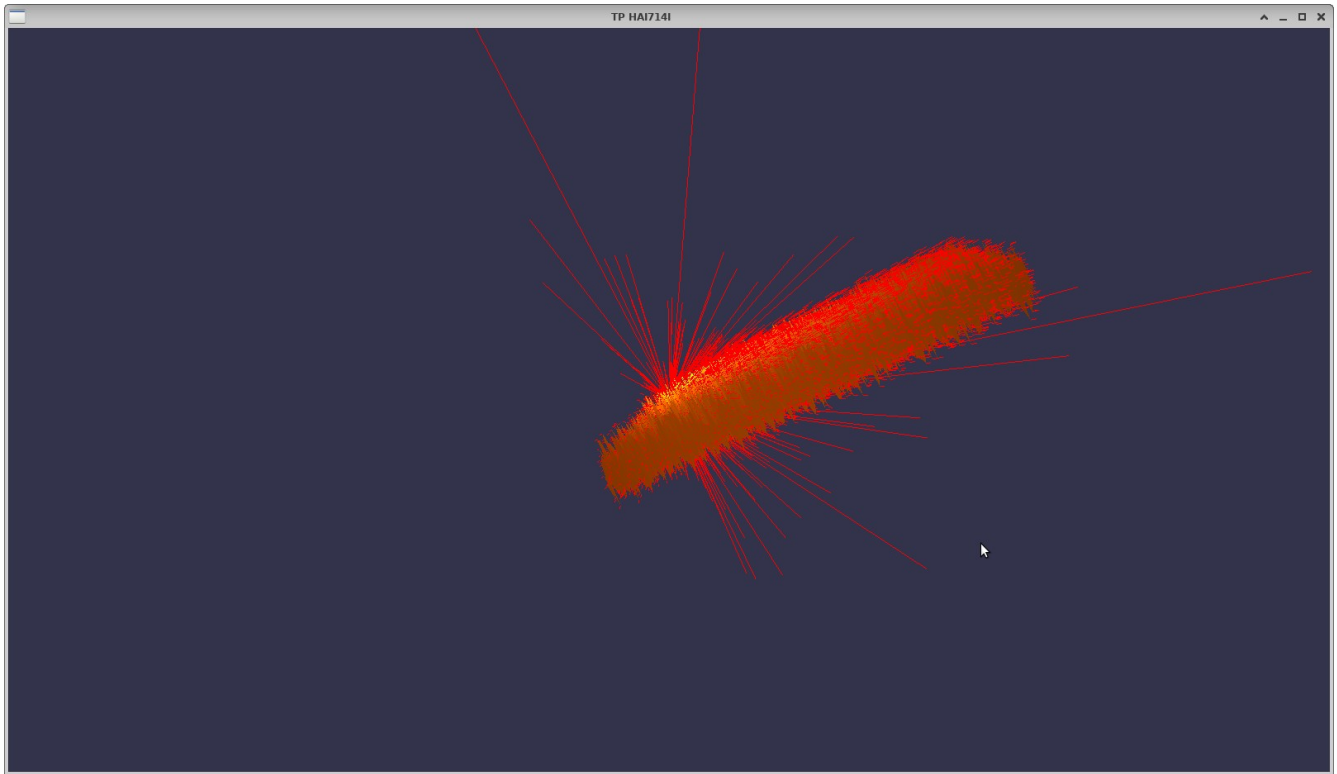
plan pour $nX=nY=10$ +normales



plan pour $nX=nY=200$



zoom maillage pour $nX=nY=200$



normales pour $nX=nY=200$

Conclusion:

Dans le premier exercice, nous avons effectué la création d'une sphère en utilisant des équations paramétriques et en manipulant les angles theta et phi. Nous avons également abordé des concepts essentiels tels que la normalisation des vecteurs et la gestion des couleurs pour chaque sommet.

Le deuxième exercice nous a conduit à générer un plan avec des reliefs aléatoires et en incorporant des facteurs aléatoires pour créer des détails sur la surface.

En somme, ces exercices nous ont dotés des compétences fondamentales nécessaires pour créer des modèles 3D simples, comprendre les notions clés de la modélisation géométrique, et commencer à explorer les vastes possibilités offertes par la programmation 3D avec OpenGL. Ils constituent une base solide pour des projets futurs plus complexes et stimulants dans le domaine de la modélisation et de l'infographie 3D.