

Master Informatique

Programation Graphique Haute-Performance

Documents non autorisés – Durée 1h30

Vos réponses doivent être claires et concises (respect des maxima indiqués). L'examen est long mais noté sur 25, il n'est donc pas indispensable de répondre à toutes les questions.

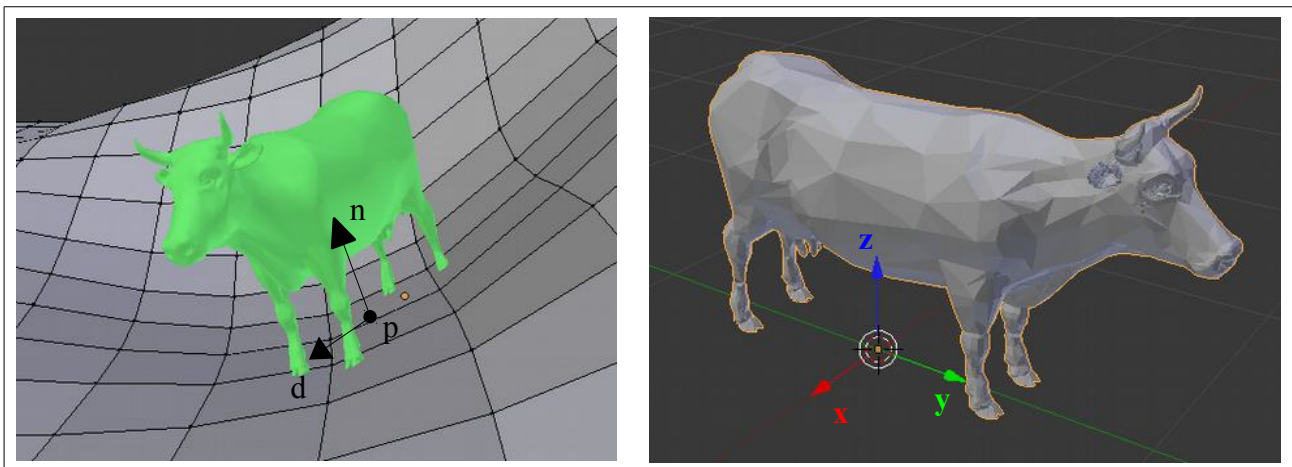
A Transformations de l'espace (4pts)

Dans cet exercice nous supposons que nous avons accès à une fonction $rot(\theta, a)$ qui retourne une matrice de rotation d'angle θ autour de l'axe a , ainsi qu'une fonction $trans(t)$ qui retourne une matrice effectuant une translation d'un vecteur t . Ces deux fonctions retournent des matrices 4x4.

A.1 Indiquer la matrice retournée par $trans([x \ y \ z])$.

A.2 Indiquer la matrice retournée par $rot(-\pi/2, [1 \ 0 \ 0])$.

A.3 On souhaite positionner un objet sur un terrain de façon à ce qu'il semble posé au sol. A partir du point p où l'on souhaite placer l'objet, de la normal n du terrain en ce point et d'un vecteur d indiquant dans quelle direction l'objet doit être orienté, indiquez comment calculer la matrice de transformation permettant de réaliser cela. Nous supposons que le modèle de l'objet est réalisé de façon à ce qu'il soit orienté vers l'axe +y, que le haut soit indiqué par l'axe +z et qu'il soit "posé" sur le plan x-y ($z=0$).



B OpenGL (14 pts)

B.1 Donnez le schéma du pipeline OpenGL en mettant en évidence les principaux étages de traitement.

B.2 Quel est le principe du *displacement mapping*? Quels sont ses principaux avantages et inconvénients par rapport au *normal mapping*? (5 lignes max)

B.3 Citez trois techniques **complémentaires** permettant d'accélérer le rendu d'une scène complexe par rasterisation. Expliquez le principe d'une de ces techniques au choix. (10 lignes max + schéma si nécessaire)

B.4 Quel est le principal avantage et le principal inconvénient des textures 3D par rapport aux textures 2D? (3 lignes max)

B.5 A quoi sert le mipmapping? Quel est sa principale limitation. (3 lignes)

B.6 Expliquez comment rendre une scène comportant des objets semi-transparents. Chaque objet possède un coefficient $\alpha_i \in [0,1]$ où 0 signifie que l'objet est complètement transparent, et 1 que l'objet est opaque. (6 lignes max)

B.7 Expliquez comment simuler un vidéo projecteur projetant une image sur une scène 3D arbitraire (pas uniquement un mur plan). (6 lignes max)

C CUDA (7pts)

C.1 Voici un exemple de code CUDA appliquant la fonction f sur un ensemble de données en fonction d'un masque binaire :

```
__GLOBAL__ void kernel(Data *data, int *mask) {  
    int tid = threadIdx.x + blockIdx.x * blockDim.x;  
    if(mask[tid])  
        f(data[tid]);  
}
```

On suppose que f est une fonction coûteuse, et que $mask$ contient seulement 10 % de 1, et 90 % de 0.

- Pourquoi un tel code n'est pas efficace sur une architecture de type CUDA? (5 lignes max)
- Comment améliorer les performances? (5 lignes max)

C.2 Voici un exemple de code CUDA calculant le centre de masse d'un ensemble de points :

```
1 : __GLOBAL__ void kernel(float *masses, float3 *positions, int n) {
2 :   int tid = threadIdx.x + blockIdx.x * blockDim.x;
3 :   if(tid<n)
4 :     positions[tid] = masses[2*tid]*positions[2*tid]
5 :                   + masses[2*tid+1]*positions[2*tid+1];
6 : }
7 :
8 : float3 center_of_mass(float *masses, float3 *positions, int n) {
9 :   float *d_masses;
10 :  float3 *d_positions;
11 :
12 :  /* Allocations of d_masses and d_positions,
13 :     and copies of masses and positions to d_masses and d_positions */
14 :  [...]
15 :
16 :  dim3 DimBlock(n,1,1);
17 :  dim3 DimGrid(1,1,1);
18 :  for(int size = n/2; size>1; size /= 2)
19 :    kernel<<<DimGrid,DimBlock>>>(d_masses, d_positions, size);
20 :
21 :  return d_position[0];
22 : }
```

- a) Ce code présente deux erreurs faisant que le résultat retourné par la fonction `center_of_mass` n'est pas correct. Quelles sont-elles et comment les résoudre ?
- b) Quelles sont les limitations sur les entrées/arguments ?
- c) Expliquez, sans fournir de code, comment obtenir de meilleures performances ?