

On se propose de réaliser un logiciel permettant de se déplacer dans un espace tri-dimensionnel ainsi que de manipuler cet espace un peu à la manière de *minecraft*.

L'espace est un cube discret de côté  $2^n$ . Chaque cube élémentaire est soit plein soit vide.

Le logiciel doit avoir un mode permettant de créer un cube élémentaire là où il n'y en a pas ou d'en enlever un là où il y en a. D'autre part on doit pouvoir sauvegarder l'état d'un espace ou relire l'état d'un espace. Le codage d'un espace dans un fichier doit respecter le format (texte) suivant :

- la première ligne un identifiant de format : 1 ou 2
- la seconde ligne contient la taille du côté du cube :  $2^n$
- le reste du fichier est constitué de  $(2^n)^3$  0 ou 1 (pour le format 1 noir et blanc) indiquant si le cube élémentaire est plein ou vide ou  $(2^n)^3$  triplet R G B (pour le format 2) indiquant la couleur du cube élémentaire ou -1 -1 -1 si le cube est absent

Par exemple le fichier :

```
1
2
1 1 1 1 1 1 1 1
```

représente le cube plein constitué de 8 cubes élémentaires.

Par exemple le fichier :

```
2
2
255 0 0 255 0 0 255 0 0 255 0 0 255 0 0 255 0 0 255 0 0
```

représente le cube plein constitué de 8 cubes élémentaires rouges. On doit aussi avoir des petits programmes annexes permettant de créer des espaces prédéfinis comme :

- **createCubeInCube**  $n$   $m$  permettant de créer un espace de taille  $2^n$  (au format 1) contenant un sous-cube plein de taille  $2^m$ .
- **createLineInCube**  $n$   $x_0$   $y_0$   $z_0$   $x_1$   $y_1$   $z_1$  permettant de créer un espace de taille  $2^n$  contenant une ligne de cubes entre les points  $(x_0, y_0, z_0)$  et  $(x_1, y_1, z_1)$ . Le tracé dans l'espace de la ligne devra être obtenu par l'algorithme de Bresenham étendu à la 3D.
- **createSphereInCube**  $n$   $x_c$   $y_c$   $z_c$   $r$  R G B permettant de créer un espace (au format 2) de taille  $2^n$  contenant une sphère de rayon  $2^m$  de couleur R G B centrée en  $(x_c, y_c, z_c)$ . Le tracé de cette sphère par l'algorithme DDA étendu à la 3D sera apprécié.

Une fois un espace créé on doit pouvoir s'y déplacer à l'aide mouvements fluides dans une vue en perspective avec des effets permettant de se représenter facilement la profondeur. La perspective doit pouvoir être contrôlée en choisissant par exemple la focale, le champ de vision, etc. La position de la caméra ne doit pas être discrète et les 6 degrés de liberté de l'espace 3D doivent être manipulables (3 translations, 3 rotations)! Dans cette vue il doit être possible de déplacer un «curseur» 3D permettant de représenter en fil de fer un sous-cube quelconque de l'espace (par-dessus la perspective) puis de remplir ou vider ce sous-cube. Note pour mieux se représenter les cubes élémentaires, on prendra soin de distinguer les arêtes de leurs faces...

Dans un second temps on essaiera d'implémenter les octrees afin de représenter efficacement l'espace. On devra aussi étendre le format de lecture et d'écriture sur fichier afin de stocker l'espace sous forme d'octree.

## Suggestions

- effets de lumière avec des lumières placées dans le cube (à la place d'un cube élémentaire, à la place de l'œil)
- suivi de trajectoire, replay