Cours de programmation séquentielle

Écriture de fichiers STL

1 Buts

— Écriture de fichiers textes.
— Visualisation de fichier STL.
— Manipulation de triangles.

2 Énoncé

Un fichier STL (pour stéréolithographie) est un format de fichier permettant d'écrire de représenter des surfaces à l'aide de triangles. Il est souvent utilisé par les outils de conception par ordinateur, pour l'impression 3D, ...

Un fichier STL décrit une surface triangulée. Chaque composant est un triangle, décrit par ses trois sommets (trois points tri-dimensionnels) ainsi que sa normale.

Dans ce travail, vous allez créer une structure `triangle` composée de trois sommets (vertex/vertices en anglais) qui sont des `point_3d` (voir le travail pratique de la semaine passée) et une librairie permettant de manipuler ces triangles. Puis, vous écrirez un petit outil permettant d'écrire des fichiers STL.

2.1 Le format ASCII-STL

Pour des raisons de simplicité, nous allons écrire les fichiers STL sous la forme de fichiers ASCII (texte). Ces fichiers sont composés de trois parties:

1. Une entête, qui est composée d'une seule ligne

   ```
   solid name
   ```

   où `solid` est un mot clé et `name` est une chaîne de caractères donnant le "nom" de votre surface.
2. Une suite de faces triangulaires (il peut y en avoir un nombre arbitraire). Chaque face a la structure suivante

   ```
   facet normal ni nj nk
       outer loop
           vertex v1x v1y v1z
           vertex v2x v2y v2z
           vertex v3x v3y v3z
       endloop
   endfacet
   ```

où `v1x/y/z`, `v2x/y/z`, et `v3x/y/z` sont les trois sommets du triangle et `ni nj nk` sont les trois composantes du vecteur normal du triangle. Les sommets du triangle doivent être ordonnés selon la "règle de la main droite" (de façon cohérente avec la normale). Ces valeurs sont écrite en notation scientifique (utilisez le format `%e` pour les écrire). L'indentation n'est pas obligatoire.

3. Finalement, le fichier se termine par

   ```
   endsolid name
   ```

   où `name` doit être le même qu'au point 1.

3 Travail à réaliser

3.1 Compléter la librairie `point_3d`

Bien que cela ne soit techniquement pas la meilleure solution en général, pour des raisons de simplicité, rajoutez les fonctions suivantes dans votre librairie `point_3d`:

1. Une fonction normalisant un `point_3d` en place:
   ```
   void point_3d_normalize_inplace(point_3d *p);
   ```
   La normalisation d'un vecteur, $\vec{v}$, se fait en divisant le vecteur par sa norme
   $$\frac{\vec{v}}{||\vec{v}||}. \tag{1}$$

2. Une fonction retournant un vecteur de longueur un, à partir d'un vecteur quelconque
   ```
   point_3d point_3d_normalize(point_3d *p);
   ```

3. Une fonction calculant le produit vectoriel de deux vecteurs
   ```
   point_3d point_3d_vector_product(point_3d *p1, point_3d *p2);
   ```
   Le produit vectoriel, $\vec{w}$, de deux vecteurs, $\vec{u}$, $\vec{v}$ se calcule à l'aide de la formule suivante
   $$\vec{w} = \vec{u} \times \vec{v} = (u_y v_z - u_z v_y, -u_x v_z + u_z v_x, u_x v_y - u_y v_x)^{\mathrm{T}}. \tag{2}$$

3.2 Écrire une librairie de manipulation de triangles

La structure de triangles géométriques est la structure `triangle`

```
typedef struct {
    point_3d v1, v2, v3;
} triangle;
```

Un triangle peut se représenter grâce à la géométrie vectorielle. Si les sommets d'un triangle sont repérés par les vecteurs $\vec{v}_1$, $\vec{v}_2$, et $\vec{v}_3$ (voir la fig. 1).

Les arêtes du triangle sont les vecteurs $\vec{v}_{12}$, $\vec{v}_{13}$, et $\vec{v}_{23}$ qui relient respectivement $\vec{v}_1$ et $\vec{v}_2$, $\vec{v}_1$ et $\vec{v}_3$, et $\vec{v}_2$ et $\vec{v}_3$ (voir la fig. 2).

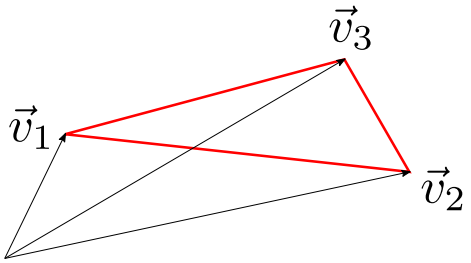

FIGURE 1: Les vecteur $\vec{v}_1$, $\vec{v}_2$, $\vec{v}_3$ représentant un triangle.

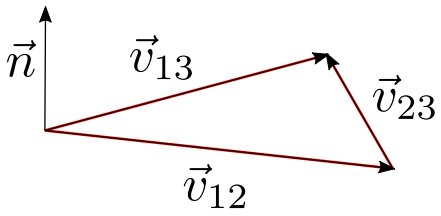

FIGURE 2: Les vecteur $\vec{v}_{12}$, $\vec{v}_{13}$, et $\vec{v}_{23}$ reliant les sommets d'un triangle, ainsi que le vecteur normal $\vec{n}$.

Les vecteurs $\vec{v}_{ij}$ se calculent à l'aide de la formule

$$\vec{v}_{ij} = \vec{v}_j - \vec{v}_i. \tag{3}$$

Le produit vectoriel de deux arêtes permet de calculer le vecteur normal au triangle (voir la fig. 2), ainsi que sa surface

$$\vec{n} = \vec{v}_{12} \times \vec{v}_{13}, \tag{4}$$

$$A = \frac{1}{2}||\vec{n}||. \tag{5}$$

Pour cette partie du travail, il s'agit d'écrire les fonctions suivantes:

1. Une fonction triangle créant un triangle, à partir de trois sommets:
   `triangle triangle_create(point_3d v1, point_3d v2, point_3d v3);`
2. Une fonction calculant le vecteur normal à la surface d'un triangle
   `point_3d triangle_compute_normal(triangle *t);`
3. Une fonction calculant la surface d'un triangle
   `double triangle_compute_area(triangle *t);`

3.3 Librairie d'écriture de fichier STL

La structure fichier STL de triangles géométriques est la structure `stl_file`

```
typedef struct {
    char *name;
    FILE *file;
} stl_file;
```

Il faut écrire les fonctions suivantes:

1. Une fonction qui crée une nouvelle la structure `stl_file`
   ```
   stl_file stl_file_create(char *filename, char *name);
   ```
   qui ouvre le fichier `filename` et initialise `name`.
2. Une fonction qui écrit le pied de page, ferme le fichier, mets `name` à `NULL`
   ```
   void stl_file_close(stl_file *s);
   ```
3. Une fonction qui écrit un triangle dans le fichier STL en argument (la fonction `fprintf()` pourrait vous être utile)
   ```
   void stl_file_write_triangle(stl_file *s, triangle *t);
   ```
4. Une fonction qui écrit une liste de triangles dans le fichier `filename` avec un `name` passés en argument
   ```
   void stl_file_create_and_write(char *filename, char *name,
                                  triangle *t, int num_triangles);
   ```

4 Tests

Afin de tester votre code, essayez d'écrire un cube dont les sommets sont:

`(0,0,0), (0,0,1), (0,1,0), (0,1,1), (1,0,0), (1,0,1), (1,1,0), (1,1,1)`

Il se décompose en 2 triangles par face de cube. Il doit ressembler au fichier `cube.stl` se trouvant sur cyberlearn.

Vous pouvez visualiser votre fichier STL, à l'aide du programme `paraview` qui doit être installé sur toutes les machines. Sinon vous pouvez également utiliser le site https://www.viewstl.com/.