

WebGL TP6

LP DAWIN

2015-2016

Le but de ce TP est de passer de le 2D à la 3D. Pour cela, vous aurez besoin de la bibliothèque glmatrix, que vous trouverez sur <http://glmatrix.net>, ou bien en incluant <https://raw.githubusercontent.com/toji/glmatrix/master/dist/gl-matrix-min.js> Pensez à lire la documentation, le seul type que nous utiliserons est `mat4`.

Pour commencer, vous devez avoir un dessin de triangle en 2D fonctionnel.

1 Dessiner en 3D

Pour passer de la 2D à la 3D, il faut prendre en compte la *profondeur* (axe Z) et réaliser la *projection* du monde 3D sur votre écran en 2D.

1.1 Premier dessin

1. Ajoutez dans votre programme une matrice de projection et initialisez la avec la fonction `perspective`.
2. Ajoutez un `uniform` de type `mat4` dans le vertex shader et utilisez `gl.uniformMatrix4fv` pour y envoyer la matrice.
3. Maintenant que la matrice de projection est dans le shader, vous pouvez changer la position en `z` des sommets du triangle, appliquer la projection et dessiner.
4. Essayer plusieurs valeurs de `z` et observez le résultat.

1.2 Ajouter Z dans les buffers

1. Ajoutez une troisième coordonnée dans le buffer de position et faites toutes les modifications nécessaires.

2 Transformations en 3D

2.1 Rendre le triangle visible

1. Remettre tous les `z` du triangle à 0.
2. Nous allons utiliser une deuxième matrice de transformation pour passer des coordonnées objet à modèle. Faites les modifications nécessaires pour utiliser cette matrice.
3. Utilisez cette matrice pour déplacer le triangle sur l'axe `z`, afin de le rendre visible.

2.2 Rotations et ordre des transformations

Implémentez les **animations** suivantes :

1. Appliquez une rotation autour de l'axe Z.
2. Faites tourner le triangle sur lui-même autour de la verticale.
3. Faites tourner le triangle autour de la caméra.

3 Cube

1. Modifier les buffers pour dessiner un cube (6 faces, 12 triangles, 36 sommets)
2. Ajouter de la couleur aux sommets du cube en fonction de leur position. Pour cela, pas besoin d'ajouter un buffer, vous pouvez directement utiliser la position.
3. Ajouter une animation de rotation du cube. Que constatez-vous ?
4. Y remédier en ajoutant `gl.enable(gl.DEPTH_TEST)` dans la fonction de dessin. Il faut aussi changer le clear en `gl.clear(gl.COLOR_BUFFER_BIT | gl.DEPTH_BUFFER_BIT)`.