

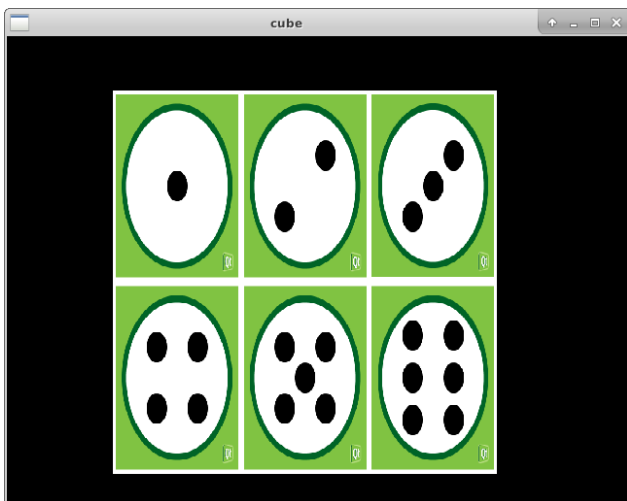
Question 1 :

La classe 'MainWidget' touche à tout ce qui a trait à l'écran, l'affichage et l'interaction avec celui-ci. La classe 'geometryEngine' construit l'objet qui figurera lors de l'affichage. Le fichier 'vshader' manipule les différents vertex de l'objet, les couleurs associées, la position. Tandis que 'fshader' est utilisé pour l'affichage des pixels, et la modification de leur couleur.

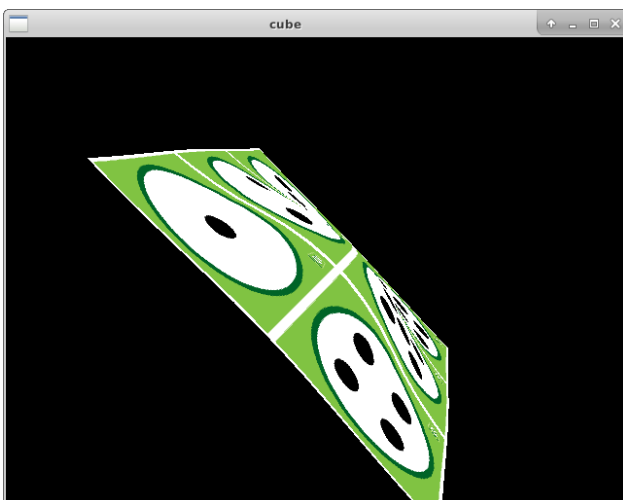
Question2 :

La méthode 'GeometryEngine::initCubeGeometry()' lance la construction d'un cube tandis que 'GeometryEngine::drawCubeGeometry(QOpenGLShaderProgram \*program)' réalise la construction du cube en remplissant 2 tableaux, le tableau des sommets du cube et le tableau des indices des sommets permettant la construction des faces du cube.

Question 3 :



Question 4 :



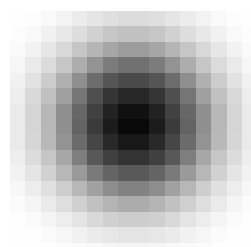
Action clavier :

Flèche haut/bas : déplacer la caméra selon y

Flèche gauche/droite : déplacer la caméra selon x

Molette souris : déplacer caméra selon z

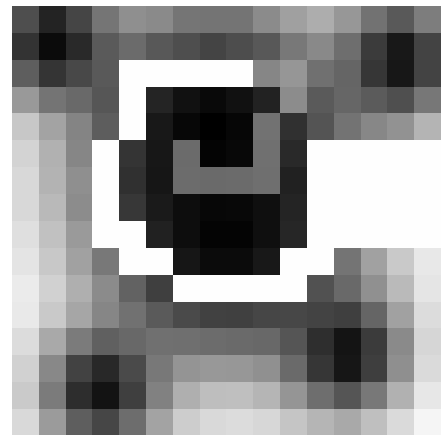
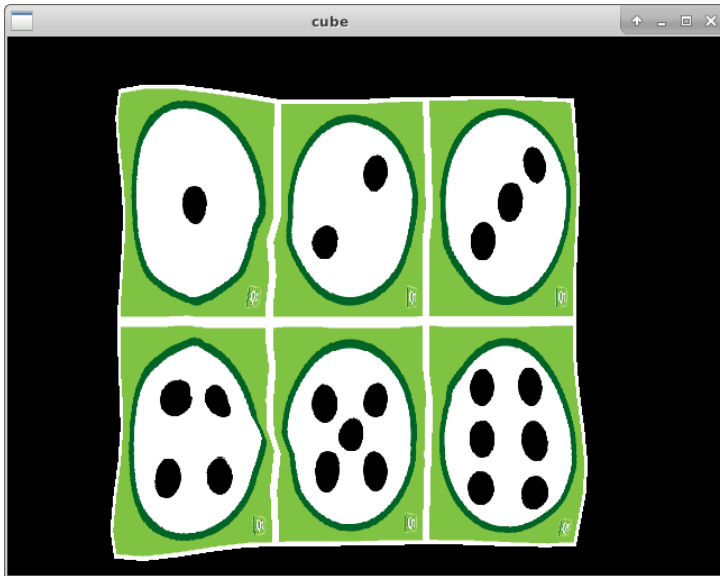
Clic souris : faire pivoter l'objet



heightMap associée

Question 5 :

La heightMap est construite à partir d'une image 16\*16 au format pgm



heightMap associée

#### Question 6 :

Pour observer la rotation du plan, se référer à la fonction `pivoter()` dans la classe `mainwindow()`. Celle-ci indique un axe et une vitesse de rotation à appliquer au modèle.

#### Question 7 :

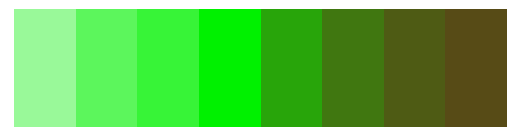
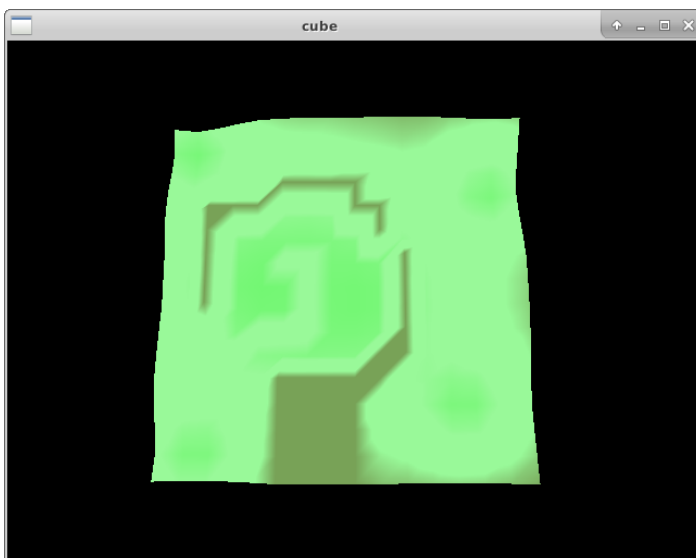
La mise à jour de l'affichage, se fait par un timer de la classe `Qtimer`, celui-ci est initialisé en utilisant la fonction `timer.start()` et une valeur indiquant tous les combien de milliseconde l'on souhaite que l'affichage soit rafraîchi. Pour le modifier, il suffit de changer la valeur d'initialisation. Pour raisonner en fps, il est nécessaire d'entrer une valeur telle que `timer.start(1000/fps)`. Toutefois un affichage à 1000 fps n'est guère être plus rapide qu'un affichage à 60 fps. Cela est dû à la vitesse de rafraîchissement des écrans est elle même limitée. Cependant entre 1 et 60 fps la différence de fluidité est notable !

Pour modifier la vitesse de rotation dans toutes les fenêtres à la fois, celle-ci se fait avec une variable globale et l'utilisation des touches `[+]` et `[-]`. On constate une perte de fluidité pour 1 fps si la rotation est trop rapide.

#### Partie bonus :

Pour les lumières, utiliser les lumières dans `OpenGL`, mais il est nécessaire de calculer un tableau des vecteurs normaux de chaque triangle et de respecter un sens de calcul, pour que les ombres soient toutes cohérentes.

Pour l'affichage de texture dépendante de la couleur, il faudrait utiliser une image palette de dégradé et en fonction de l'altitude se rapprocher ou non d'un des bords de la palette.



Palette couleur d'altitude