

OPENGL 4

Durée

3 jours

Référence Formation

4-LC-OGM

Objectifs

Appréhender et comprendre OpenGL dans ses versions les plus récentes

Participants

Développeur.

Pré-requis

Notions de base du monde de la 3D, langage C

Moyens pédagogiques

Accueil des stagiaires dans une salle dédiée à la formation équipée d'un vidéo projecteur, tableau blanc et paperboard ainsi qu'un ordinateur par participant pour les formations informatiques.

Positionnement préalable oral ou écrit sous forme de tests d'évaluation, feuille de présence signée en demi-journée, évaluation des acquis tout au long de la formation.

En fin de stage : QCM, exercices pratiques ou mises en situation professionnelle, questionnaire de satisfaction, attestation de stage, support de cours remis à chaque participant.

Formateur expert dans son domaine d'intervention

Apports théoriques et exercices pratiques du formateur

Utilisation de cas concrets issus de l'expérience professionnelle des participants

Réflexion de groupe et travail d'échanges avec les participants

Pour les formations à distance : Classe virtuelle organisée principalement avec l'outil ZOOM.

Assistance technique et pédagogique : envoi des coordonnées du formateur par mail avant le début de la formation pour accompagner le bénéficiaire dans le déroulement de son parcours à distance.

PROGRAMME

- Présentation

Versions et historique (1.x à 4.x, ES1, ES2)

Compatibilité GL et GLES

Place d'OpenGL sur le marché actuel de la 3D (WebGL, DirectX, Vulkan)

Principes de fonctionnement d'une carte 3D

Pipeline programmable

Extensions OpenGL

Bindings et langages

- Initialisation et contexte

Création de l'espace de rendu

Les API concernées : GLX, WGL, CGL, EGL, □

Les abstractions possibles : GLUT, SDL, □

Gestion des extensions (GLEW, GLEE, □)

Les bibliothèques "support" (GLU, GLM, matrices et vecteurs, □)

CAP ÉLAN FORMATION

www.capelanformation.fr - Tél : 04.86.01.20.50

Mail : contact@capelanformation.fr

Organisme enregistré sous le N° 76 34 0908834

[version 2023]

- Configuration de la scène

Espace de rendu 3D : frustum, projection,
Tampons Z et stencil
Face culling

- Les géométries

VBO (Vertex Buffer Object)
Buffers indexés et non-indexés
Buffers entrelacés et non entrelacés (interleaved)
VAO (Vertex Array Object)
Organisation des structures de données, stockage d'objets
Transform Feedback

- Les textures

Formats de fichiers basiques
Taille des textures, NPOT
Nombre de dimensions des textures (1D, 2D, 3D)
Textures cubemaps
Chargement, stockage et sampling d'une texture
Notion de MipMaps
Formats spécifiques (DDS, KTX,)
Formats compressés (S3TC, ETC1)
ETC2/EAC (GL 4.3+)
Immutable Storage

- Les Shaders

Les étages de base (vertex, fragment)
Les étages avancés (tessellation, geometry, transform)
Le langage GLSL
Types de données (vec, mat, sampler,) et tailles
Entrées / sorties des shaders
Branchements et itérations
Récupération du programme binaire (GL 4.1+)
Techniques avancées, présentation de raymarching

- Compléments

Gestion des erreurs (GL 4.3+)
Présentation des Compute Shaders (GL 4.3+)