

OPENGL ES

Durée

3 jours

Référence Formation

4-LC-OGES

Objectifs

Comprendre le rôle, les possibilités et les contraintes d'OpenGL ES dans le monde de la 3D temps réel embarquée
Comprendre les différences entre OpenGL et OpenGL ES, ainsi que la différence de vision entre OpenGL ES 1.X et 2.X.

Participants

Développeur.

Pré-requis

Connaissances de base en développement. Les démonstrations seront réalisées à l'aide du langage C.

PROGRAMME

- 1. Présentation d'OpenGL
place d'OpenGL sur le marché actuel de la 3D
rôle d'OpenGL et compléments nécessaires
ce qu'OpenGL n'est pas et ce qu'il ne fait pas
notions : rasterisation, vertex, fragment, pixel, texel, ...
- 2. OpenGL ES
différences et spécificités
OpenGL ES
évolution d'OpenGL ES par rapport à OpenGL
convergence avec OpenGL
gestion de la performance et de la mémoire, optimisations possibles
implémentations d'OpenGL ES
portabilité des applications
correspondances entre les versions d'OpenGL et d'OpenGL ES
- 3. OpenGL ES 1.x : fixed pipeline
espace de rendu 2D, framebuffer, buffering, ...
machine à états
matrices
espace de rendu 3D : frustum
géométries et modèles : meshes
vertex arrays, vertex buffers
éclairage, ombrages et ombres portées
blending, transparences, brouillard, lissage, ...
textures, multitexturing, mipmaps, compression, ...
tampons Z et stencil
skyboxes, systèmes de particules, ...
- 4. OpenGL ES 2.X : shaders
présentation, changement d'orientation
comment retrouver les fonctionnalités du pipeline fixe
gérer la compatibilité entre OpenGL ES 1.X et 2.X
impact sur les performances
portabilité des shaders
OpenGL ES Shading Language (GLSL)
vertex shader, fragment shader
multitexturing, stencil/depth test, per-pixel lighting, image space post-processing, ...

présentation d'autres utilisations avancées des shaders
évolutions probables

- 5. Autour d'OpenGL ES : conception d'applications complètes

intégrer les autres domaines

- entrées utilisateur
- sons et effets
- physique
- gérer les assets au sein du projet
- modélisation 3D, textures (contraintes, règles, outils, ...)
- formats (performance ou standards ?)
- workflow caractéristique de conception (application et contenu)
- étapes du développement, maquettage, itérations
- porter la logique et la structure de la scène
- scène graphs
- bibliothèques et moteurs existants
- moteurs 3D
- moteurs applicatifs dédiés

- 6. Bindings et intégration

- quels langages ?
- OpenGL et le web

intégration de contenu / rendu tiers (bitmap, vectoriel, vidéo, ...)

OpenGL en tant que système de fenêtrage

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.