TP Timers

Rémi Ronfard remi.ronfard@inria.fr  http://team-inria.imagine

# Objectifs

* Etudier et modifer la boucle (game loop) de rendu du TP2.
* Afficher la scène dans 4 fenêtres différentes, avec 4 timers
* Animer la scène à vitesse constante.
* Bonus : étudier et modifier l’organisation logicielle des différentes classes.

# Questions de cours (Cours 1)

Question 1 : Etant donnés les vecteurs a = (2,4,6) et b=(3,5,7) et le scalaire c = 2, calculer le produit scalaire a.b, le produit vectoriel axb et le produit cb.

Question 2 : Quelle est la relation entre les produits scalaires a.b et b.a ? entre les produits vectoriels axb et bxa ?

Question 3 : Connaissant la position P = (x,y,z) du joueur, comment tester rapidement si il est plus proche d’un point A = (xa, ya, za) ou d’un point B (xb, yb, zb) ?

Question 4 : calculer le produit AB des deux matrices

1 2 3

A = 4 5 6

7 8 9

et

2 0 0

B = 0 2 0

0 0 2

Question 5 : quand la transposée d’une matrice est-elle égale à son inverse ? proposer une méthode rapide pour vérifier cette condition.

# Déroulement du TP

Question 6 : Ouvrir, compiler et exécuter le projet TP3. Tester les interactions utilisateurs (touches A,Z,E,Q,S,W,X). Décrire ce que fait chacune.

A quoi servent les paramètres etat, ss, rotX et rotY ? Créer une classe Camera contenant tous ces paramètres, et modifier la classe GameWindow pour qu’elle possède un pointeur vers sa caméra.

Question 7 : Modifier la classe GameWindow pour qu’elle effectue le rendu de la scène à intervalles fixes (et non pas à chaque modification). On utilisera pour cela la classe QTimer qui permet d’associer l’exécution d’une méthode de la classe GameWindow (renderNow) à un signal (timeout) donné par le timer.

timer->connect(timer, SIGNAL(timeout()),this, SLOT(renderNow()));

Modifier le constructeur de la classe GameWindow pour qu’il prenne en paramètre la fréquence de mise à jour (frames per second).

Question 8 : Créer quatre instances de la classe GameWindow, avec des fréquences de mise à jour différentes de 120 fps, 60 fps, 30 fps et 1 fps.

Par souci de simplicité, les quatre instances partageront la meme caméra et recevront les memes événements utilisateurs (touches clavier).

Question 9 : Ajouter une fonction d’animation qui fait tourner la scène à vitesse constante autour de l’axe Y, sans intervention de l’utilisateur. Utiliser pour cela une touche clavier en mode « on/off » : lorsque l’utilisateur tape la touche « C » une première fois la scène se met en mouvement ; lorsque l’utilisateur tape la même touche « C » une seconde fois, la scène s’immobilise.

Question 10 : Utiliser les touches « P » et « M » pour modifier la fréquence d’échantillonnage de chaque instance de votre moteur de jeu. Par exemple, on pourra multiplier la fréquence par 2 à l’aide de la touche « P » et la diviser par 2 à l’aide de la touche « M ». Que se passe-t-il lorsque la fréquence

Bonus : pourquoi la classe GameWindow hérite-t-elle de OpenGLWindow ? Quels sont les avantages et inconvénients de cet héritage ? Comment serait-il possible d’éviter cet héritage ?

# Compte rendu

Répondre aux questions de cours.

Répondre aux questions de TP.

Décrire les classes et méthodes implémentées pour chaque question.

Bonus : répondre aux questions sans modifier votre code.