

Projet PCII 2015-2016

1. Rappel des objectifs

Pour rappel voila les informations officiels de cette unité d'enseignement.
Projet Programmation concurrentielle et interfaces interactives (5 crédits)
Volume Horaire : Cours : 5h , TD : 10h , TP : 25h , Travail personnel : 25h
Compétences : Donner les outils conceptuels et techniques pour
(i) concevoir et développer des programmes concurrents, et
(ii) concevoir et réaliser l'interface graphique d'applications interactives.



frederic.vernier@u-psud.fr

Description :

Le cours présente une introduction aux concepts fondamentaux de la Programmation Concurrentielle et l'Interaction Homme-Machine (IHM). Les travaux dirigés permettent de mettre en œuvre, et les travaux pratiques à combiner les concepts présentés dans le cours dans un projet. Le cours est divisé en 2 parties. La première partie introduit le concept de Programmation Concurrentiel et la notion des "Threads". La deuxième partie du cours introduit quelques concepts fondamentaux de l'Interaction Homme-Machine (IHM), puis des d'outils de conception et de développement d'interfaces graphiques.

Les travaux pratiques combinent les notions introduits dans ce module et les modules de Systèmes d'exploitation et programmation concurrente(Info32B) programmation orientée objet (Info211A) et Génie Logiciel (info317N), Informatique graphique (Info 216 et Info 112) dans un projet de programmation.

Partie I:

Rappel Entrées/Sorties et Notions de Programmation Concurrentiel
Notion des Threads et de la classe Runnable du langage Java
Synchronisation, code critique et accès concurrent
Producteur/Consommateur et Interblocage

Partie II:

Interaction graphique et interfaces à manipulation directe
Modélisation conceptuelle et guides de conception
Outils de construction d'interface et boîte à outils Swing
Critères ergonomiques

Modalités de contrôle :

40 % Contrôle Continu (Présence, démonstrations intermédiaires) et 60 % Soutenance finale + Rapport.

2. Sujet du TP Challenge !

Dans ce mini-projet d'entraînement il vous est demandé de créer un petit jeu très simple pendant une journée. Un anneau rouge au centre de l'écran rebondit quand l'utilisateur appuie sur une touche. Sans appuyer sur une touche, le cercle tombe en accélérant (comme sous l'effet de la gravité). En appuyant à répétition, le cercle monte très vite par a-coup. Le jeu consiste à garder l'anneau **enfilé** dans une ligne brisée qui avance sans fin. La figure de droite montre un exemple ou l'utilisateur a trop appuyé sur la touche, le cercle a finit par monté trop haut et il a touché la courbe noire par le bas(perdu). Un score s'affiche en fonction de la longueur parcourue et un message s'affiche entre 2 parties.



3. Sujet du projet

Dans ce projet, il vous est demandé de créer un jeu de "Incredible Machine" en langage Java. Le jeu de Incredible Machine consiste à déposer par manipulation directe les objets proposés dans une liste sur un "terrain de jeu" qui comporte déjà quelques objets imposés (par exemple un ballon de basket à droite et un panier à gauche). Une fois satisfait de ses placements le joueur actionne le mode "jeu" et chaque objet se met à bouger indépendamment selon les lois simple de la physique (gravité, mécanique, vent, etc.) ET chaque interaction entre objet est gérée (collision, activation) Le but de chaque niveau est de réaliser une action précise (par exemple faire rentrer le ballon de basket dans le panier). Lorsque c'est gagné, un nouveau niveau est débloqué. Si le joueur pense que c'est perdu il peut arrêter la simulation et les objets retournent à leur place d'origine. Le joueur tente alors de les placer différemment et relance le jeu.



4. Formats de fichier et d'objets

Chaque niveau devra être décrit dans un fichier au format ASCII et lu au chargement. Chaque type d'objet fera l'objet d'une classe (Ballon, Ciseaux, Ventilateur, etc.) qui sera instanciée à l'initialisation du niveau pour chaque objet imposé et au moment de dépôt pour les objets de la liste. Chaque classe d'objet aura une fonction qui chargera les graphismes en découpant des parties précises d'une grande image png qui aura été chargée à l'initialisation du jeu. L'objet "Corde" s'affichera sans graphisme à partir d'un dessin "drawLine()"

<http://vernier.frederic.free.fr/Teaching/Info340/IncrMachine.png>

issu de <http://www.spritters-resource.com/resources/sheets/13/13578.png>

Par exemple pour les ciseaux on découperait

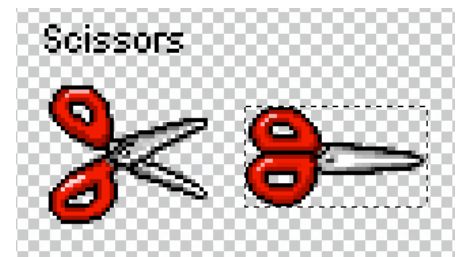
ciseaux ouverts : 748,618 -> 788->652 en y (40x34)

ciseaux fermés : 797,623 -> 841->647 en y (45x25)

Mais pour garder des ciseaux de tailles constantes on dépassera un peu en découpant :

ciseaux ouverts : 748,618 -> 793->652 en y (45x34)

ciseaux fermés : 797,618 -> 841->652 en y (45x34)



De nombreux objets nécessiteront de composer plusieurs graphismes (i.e. le ventilateur) et un automate à état pour gérer dans quel état est chaque objet à tout moment. Enfin et surtout vous devrez gérer COMMENT les actions d'un objets se propagent aux autres objets. Votre objectif sera de gérer un maximum d'objets et leurs interactions. De nombreuses ressources sur Internet ainsi que votre chargé de TD vous aideront à trouver comment marche chaque objet. Il faudra que vous gériez la physique de certains objets (ie vecteurs vitesse et accélération), les collisions (ie pour qu'une balle ferme les ciseaux elle passe sur le manche rouge avec un vecteur vitesse plus vertical que horizontal)

<http://vernier.frederic.free.fr/Teaching/Info340/AnnexeProjet2016.pdf>

5. Organisation du projet

Vous devrez réaliser une application "standalone" Java. Le code devra être découpé en classes de façon efficace et logique. Les fonctions devront être commentées (description des paramètres, du travail réalisé par la fonction, etc.).

Afin de travailler la programmation concurrente certains points devront être respectés :

- 1- Chaque objet sera pilote par son propre Thread.
- 2- Lorsque le jeu est bloqué (Message, Pause, Edition) ces Threads devront être suspendus.
- 3- Un chronomètre global devra afficher le temps écoulé.
- 4- Un Thread indépendant testera si le défi est gagné
- 5- Une glissière permettra d'accélérer/ralentir le temps