

DESSINE MOI UN MOUTON

Programmation en C++11

modes console et graphique

client C++/Java - serveur C++ - protocole CORBA

L'objectif est de créer un programme fonctionnel en C++11 utilisant les concepts de la P.O.O.

L'application devra simuler la réalisation d'une grande fresque murale réalisée à partir de plusieurs dessins d'enfants.

Les enfants (les clients) envoient à leur maîtresse (le serveur) leurs dessins afin de recueillir son avis et ses suggestions d'amélioration. La maîtresse retourne à l'élève son dessin accompagné de ses annotations. La maîtresse peut à tout moment retrouver le dessin d'un enfant.

Les dessins modifiés sont renvoyés à la maîtresse qui, une fois tous les dessins reçus dans leur version définitive, les assemble afin d'en obtenir un plus grand qui constituera la fresque murale.

- Les dessins sont constitués de formes géométriques élémentaires et colorées : lignes, polygones, cercles, ellipses.
- Les calculs du périmètre et de l'aire sont requis. L'aire d'un polygone quelconque peut être calculée par triangulation.
- Les formes géométriques peuvent se transformer par homothétie, se déplacer par translation, par rotation, par symétries centrale et axiale.
- Une image peut également contenir des images de plus petite taille (aire).
- On souhaite pouvoir trier les formes selon plusieurs critères: leur périmètre, leur aire et leur distance à l'origine.
- Aucun doublon de forme géométrique ne peut exister.

Penser à mettre en place une **gestion d'exceptions** lorsque cela s'avère nécessaire.

Dans le cadre du dialogue client-serveur, un **format des images est à définir (fichier plat)**.

I PROJET POUR LES M2 LOGICIEL : C++, JAVA, CORBA

La mise en oeuvre des points suivants est demandée :

- un client en C++
- un client en Java
- un seveur en C++
- un système de publish subscribe (producteur/consommateur) via le service de notification de Corba.

Fonctionnellement on peut choisir le mode de communication push et/ou pull.

La maîtresse est le subscriber/consommateur et les élèves les producteurs.

En mode push, les producteurs poussent les dessins vers le consommateur, ils ont l'initiative et la maîtresse attend de recevoir leurs productions.

En mode pull, c'est la maîtresse qui demande que les dessins soient poussés (produits) par les élèves.

Lorsque la maîtresse renvoie les dessins annotés aux élèves, les rôles sont échangés.

- un service d' annuaire Corba
- plusieurs maîtresses peuvent participer aux échanges

L'affichage des différentes informations est à réaliser en **mode console** :

- interactions élèves-maîtresse.
- visualisation des dessins

II PROJET POUR LES M2 IMAGE : C++

La mise en oeuvre des points suivants est demandée :

- une application client-serveur en C++ utilisant des sockets
- un affichage graphique :
 - palette de formes et couleur, positionnement des points, mode glisser-déposer,

L'utilisation d'une librairie graphique est requise (GTK, Qt ...).

RENDU PROJET

Le projet est à réaliser par **trinôme pour la filière "logiciels"** et par **binôme pour la filière "image"**.

Le rendu est sous la forme **NOM1_NOM2_NOM3.zip**, archive contenant le makefile, les .h et les .cpp et autres fichiers nécessaires, sans oublier le rapport au format pdf.

Les projets sont à déposer sur l'espace de rendu de l'ENT, ceux rendus uniquement par mail seront pénalisés.

Merci de votre sens des responsabilités.

Afin de bien mettre en évidence les relations entre les classes de votre projet, vous pouvez utiliser un logiciel qui vous générera un diagramme de classes à partir de vos sources.

L'idéal est d'utiliser un logiciel libre permettant la rétro-conception.

StarUML : <https://www.projet-plume.org/mots-cles-proposes-par-lauteur/retro-ingenierie>

BOUML : <https://www.projet-plume.org/fiche/bouml>

Une documentation des sources doit être générée automatiquement (utiliser doxygen pour documenter les sources C++ et les annotations adéquates en java afin de fournir une Javadoc).

Votre rapport pourra contenir des graphes d'appels.

Utiliser des outils pour générer des graphes d'appels :

<http://odellconnie.blogspot.fr/2012/07/free-c-software-call-graph-generators.html>

Un rapport doit fournir votre analyse et votre conception.

Les éléments suivants doivent notamment y figurer :

- une introduction exposant clairement les objectifs, limites, choix du projet
- un mode d'emploi de l'application
- des schémas décrivant l'architecture fonctionnelle
- des schémas décrivant les structures de données utilisées
- une explication en français et/ou pseudo-code et/ou langage d'implémentation choisi des principales fonctions (code à fournir et à commenter)
- une conclusion résumant le travail effectué et ouvrant des perspectives
- une bibliographie utilisée
- une table des matières

Les classes conçues doivent être détaillées aussi bien leur interface que leur implémentation.

Une bonne idée est de réaliser le rapport en même temps que le développement.

Les projets feront l'objet d'une soutenance : une démonstration sur les machines de l'Université sera requise.