

EDM4600

Algorithmie de base et interactivité

Session: Hiver 2014

Horaire: vendredi, de 9:30 à 17:00

Enseignant : Jean-Maxime Couillard

Courriel : couillard.jean_maxime@uqam.ca

Descripteur officiel

Atelier axé sur l'appropriation du concept d'interaction dans le contexte des communications médiatiques. Étude et application de techniques algorithmiques de base appliquées à la réalisation d'une production média interactive: les structures conditionnelles, aléatoires et stochastiques, l'itérativité, la récursivité, les structures de données, la gestion des événements, le contrôle des éléments audiovisuels.

Objectifs généraux

- Familiariser les étudiants avec les techniques algorithmiques et mathématiques de l'interactivité;
- les introduire au langage Java à l'aide du logiciel Processing;
- leur fournir les rudiments de la programmation orientée objet;
- les accompagner et les soutenir dans la réalisation d'exercices et de projets.

Activités d'enseignement

Chaque séance sera divisée en deux périodes:

- **avant-midi**: une présentation de la matière avec des périodes de discussion;
- **après-midi**: une période de travail pendant laquelle un auxiliaire d'enseignement sera présent pour répondre aux questions des étudiants.

Un rendez-vous est nécessaire pour toute rencontre à l'extérieur des heures du cours.

Évaluations

- 3 exercices à réaliser en atelier (pour un total de 30%, soit 10% chacun)
- 3 travaux (70%)

Travail		%
#1	Dessiner un monstre/personnage : <ul style="list-style-type: none">• Son aspect visuel doit être créé à partir des fonctions de dessin : <code>line()</code>, <code>point()</code>, <code>rect()</code>, etc.• Les fonctions de dessin doivent utiliser les variables <code>x</code> et <code>y</code> du sketch fournit par l'enseignant. Les valeurs de ces variables doivent correspondre au centre du monstre.• Ne pas dessiner d'arrière plan.• Une boucle <code>while</code> doit être utilisée pour dessiner un ou plusieurs éléments du monstre (cheveux, pattes, tentacules, etc).	15 %
#2	Poursuivre le monstre/personnage et le rendre interactif : <ul style="list-style-type: none">• Implémenter une zone interactive circulaire ainsi qu'une zone interactive rectangulaire (effet de survol et clique pour alterner d'état).• La fonction <code>background()</code> doit être la première ligne suivant la déclaration du <code>draw()</code>.• Utiliser le hasard et une minuterie.• Éviter les changements brusques à l'aide de l'assouplissement.• Ne pas dessiner d'arrière plan.	20 %

Travail		%
#3	Visualisation de données : Recevoir et traiter plusieurs événements OSC envoyés par un programme remis par l'enseignant. <ul style="list-style-type: none"> • Trouver des moyens créatifs de visualiser les données reçues. • Générer dynamiquement (<i>new</i>) des agents (issus d'au moins deux classes différentes) en réactions aux événements OSC. • Les agents doivent disparaître automatiquement après un certain temps. • Envoyer un message OSC au serveur en réaction à un événement local (collision d'agents, clique de souris, etc). Le message doit indiquer la position de l'évènement. 	35 %

- Les travaux #1 et #2 doivent être remis en ligne dans la galerie appropriée du site *openprocessing.org*. L'étudiant doit utiliser son nom (nom_prénom) en tant que nom d'utilisateur. Le champ « description » doit décrire les instructions d'utilisation et toutes les sources d'inspiration. Une image doit présenter le projet dans la galerie. La reproduction d'un exemple fourni par l'enseignant ou la reproduction en entier du code d'un projet trouvé sur Internet ne constitue pas une remise acceptable. Des extraits peuvent être copiés, mais ils doivent être utilisés avec des paramètres différents. Ces extraits et leur source doivent être clairement indiqués dans la description du travail et dans le code source.
- Le travail #3 doit être présenté en classe et remis par la suite sur le disque dur de l'enseignant. Le projet doit porter comme titre le nom de l'étudiant (non_prénom).
- La scène doit avoir une dimension de 400 x 400 pixels pour les travaux #1 et #2, et de 960 x 540 pixels pour le travail #3.
- Afin d'éviter tout plagiat, chaque bloc de code doit être assidument commenté.
- Si le travail ne respecte pas ces directives, il recevra comme note maximale C+.

Barème de notes: <http://www.edm.uqam.ca/default.aspx?pld=188>

Dates importantes

Semaine	Date	
1	10 janvier	Présentation du cours
4	31 janvier	Remise du travail #1
7	21 février	Exercice sur les boucles et les instructions conditionnelles
8	28 février	SEMAINE DE LECTURE
9	7 mars	Remise du travail #2
11	21 mars	Exercice sur les tableaux
12	28 mars	Exercice sur les classes
14	11 avril	Remise du travail #3

Manuel de base

- **Getting Started with Processing**, Casey REAS et Ben FRY, *Make*, 2010

Manuels avancés

- **Learning Processing**, Daniel SHIFFMAN, Morgan Kaufmann, 2008
- **Programming Interactivity: A Designer's Guide to Processing, Arduino, and Openframeworks**, Joshua NOBLE, *O'Reilly Media*, 2009
- **Form+Code in Design, Art, and Architecture**, Casey REAS, Chandler MCWILLIAS, *Princeton Architectural Press*, 2010