

Titre long : Création musicale en langage textuel
Titre court: Création en langage textuel

Sigle : MUS
3326X/6326X

Crédits : 3

Professeur :

Trimestre : Automne

Capacité d'accueil: 21

DESCRIPTION

Cours de formalisation musicale dans un environnement de programmation textuelle. Composition d'œuvres employant la programmation. Élaboration de processus musicaux et d'algorithmes de synthèse sonore.

INSERTION DANS LES PROGRAMMES

- Baccalauréat en musique (1-600-1-1), Bloc 02C Outils-crédation, interprétation – option, V00, A2016
- Baccalauréat en musique - Interprétation jazz (1-605-1-6), Bloc 70L Sciences et technologies de la musique – option, V06, A2012
- Baccalauréat en musique - Interprétation jazz (1-605-1-6), Bloc 80I Sciences et technologies de la musique, V07, A2021
- Baccalauréat en musique - Interprétation chant classique (1-605-1-9), Bloc 70C Sciences et technologies de la musique – option, V05, A2014
- Baccalauréat en musique - Composition instrumentale (1-610-1-3), Bloc 01H Électroacoustique et informatique musicale – option, V05, A2016
- Baccalauréat en musique - Composition mixte (1-610-1-4), Bloc 01B Électroacoustique et informatique musicale – option, V04, A2014
- Majeure en composition et arrangement jazz (1-610-2-6), Bloc 70E Électroacoustique et informatique musicale - option, V00, A2015
- Baccalauréat en musiques numériques (1-615-1-2), Bloc 70A – obligatoire, V01, A2017
- Majeure en musiques numériques (1-615-2-1), Bloc 70A – obligatoire, V02, A2013
- Mineure en musiques numériques (1-615-4-1), Bloc 70C Informatique musicale et médiatique –option, V02, A2013
- Baccalauréat en musique - Écriture (1-620-1-1), Bloc 01K Électroacoustique et informatique musicale – option, V04, A2016

FORMULE PÉDAGOGIQUE

La matière est donnée sous forme cours magistraux hebdomadaires ponctués de nombreux exemples sonores réalisés dans un environnement de programmation.

OBJECTIFS GÉNÉRAUX

- Créer des œuvres musicales en employant la programmation.
- Développer les habiletés nécessaires à la résolution de problèmes musicaux en programmation.
- Développer une pensée logique qui permet d'élaborer des algorithmes musicaux.
- Approfondir sa compréhension du traitement de signal.
- Appliquer les connaissances acquises dans le cadre d'un projet musical.

SUJETS ABORDÉS/ CHRONOLOGIE DE LA MATIÈRE

Cours 1 : Plan de cours. Faire du son dans l'environnement SuperCollider. Variables, fonctions et indentation.

Cours 2 : Langage, serveur, *UGen*, *SynthDef*, fonctions, objets et méthodes.

Cours 3 : Tableaux et itération.

Cours 4 : L'expansion des canaux 1 : stéréophonie et multiphonie.

Cours 5 : Les enveloppes, *Startup file* et l'expansion des canaux (bis).

Cours 6 : Les *patterns* 1: *Pbind*, *Pseq* et les *Events*.

Cours 7 : OSC, MIDI et contrôle externe 1.

Cours 8 : MIDI 2, fichiers sonores, synthèse granulaire 1.

Cours 9 : Synthèse granulaire 2.

Cours 10 : Synthèse granulaire 3 et révision.

Cours 11 : *Bus*, *Nodes*, ordre d'exécution et effets.

Cours 12 : Discussions sur les projets et sujets libres.

Cours 13 : Rencontres et discussions sur les projets.

Cours 14 : Présentation des projets finaux.

TRAVAUX ET ÉVALUATION

	Premier cycle	Cycles supérieurs
Devoir[0] Les <i>UGens</i>	15%	10%
Devoir[1] Les <i>SynthDefs</i>	15%	10%
Devoir[2] Les <i>Patterns</i>	15%	10%
Devoir[3] Aiguillage, OSC et <i>buffers</i>	15%	10%
Description de projet	8%	8%
Projet final (travail à options)	32%	52%
Total	100%	100%

BIBLIOGRAPHIE SOMMAIRE

Harkins, James. "A practical guide to patterns." *document d'aide intégré à SuperCollider 3*, no. 4 (2009).

McCartney, James. "Rethinking the computer music language: SuperCollider." *Computer Music Journal* 26, no. 4 (2002): 61-68.

Roads, Curtis, and John Strawn. *The computer music tutorial*. MIT press, 1996.

Shiffman, Daniel, Shannon Fry, and Zannah Marsh. *The nature of code*. D. Shiffman, 2012.

Wilson, Scott, David Cottle, and Nick Collins. *The SuperCollider Book*. The MIT Press, 2011.