

PHY 1234 – INTRODUCTION À LA PHYSIQUE NUMÉRIQUE – AUTOMNE 2017

Faculté des Arts et des Sciences – Département de Physique

PLAN DE COURS

Nombre de crédits : 3

Professeur : David Lafrenière (Roger-Gaudry, local D-434; david.lafreniere@umontreal.ca)

Horaire du cours : Mercredi 13h30-15h20 au local Z-330 Pav. Claire-McNicoll

Horaire des travaux pratiques en laboratoire :

Section A101/A102 : Jeudi 10h30-13h20 aux locaux M-625/M635, Pav. Roger-Gaudry

Section A103 : Vendredi 8h30-11h20 au local M-605/M-615, Pav. Roger-Gaudry

Démonstrateurs/correcteurs pour séances de travaux pratiques en laboratoire :

Camille Claing (camille.claing@umontreal.ca)

François Labonville (francois.labonville@umontreal.ca)

Chloé Lefebvre (chloe.lefebvre.3@umontreal.ca)

Florence Montpetit (florence.montpetit@umontreal.ca)

Arthur Plante (arthur.plante@umontreal.ca)

Buts et format du cours

Le cours vise trois objectifs complémentaires : (1) vous apprendre les bases de la programmation scientifique, en utilisant le langage *Python*; (2) vous bâtir une « boîte à outils » de méthodes, approches et outils numériques et graphiques permettant de solutionner sur ordinateurs divers problèmes rencontrés en physique (différentiation et intégration numériques, solution d'équations différentielles non linéaires, optimisation, etc.); et (3) vous offrir une introduction à certains domaines de la physique contemporaine où l'approche numérique est centrale (la diffusion, le chaos, la complexité, etc.).

Le cours comporte deux volets : une séance hebdomadaire de 2 heures en classe où seront présentés les différentes méthodes numériques et les concepts physiques, et une séance hebdomadaire de 3 heures de travaux pratiques en laboratoire informatique pour mettre en application les méthodes vues au cours et développer les habiletés de programmation. Les séances de travaux pratiques en laboratoire informatique sont une composante essentielle et obligatoire de ce cours.

Évaluation

L'évaluation du cours sera entièrement basée sur les rapports et codes informatiques à remettre suite aux séances de travaux pratiques en laboratoire. Il y aura environ 8 à 10 rapports à remettre au cours de la session. **Les rapports de laboratoires doivent être rédigés de façon individuelle et indépendante par chaque étudiant.** Chaque rapport est à remettre électroniquement par Studium à une date fixe, bien indiquée sur chaque document de laboratoire, et tout retard sans raison jugé valable entrainera une pénalité de 20% par période de 24 heures.

Notes de cours et ouvrages de référence

Des notes de cours et de laboratoire en format PDF couvrant l'ensemble de la matière vous seront fournies au cours de la session sur le site Studium du cours. Plusieurs ouvrages de référence supplémentaires sur la programmation en Python sont discutés au début du chapitre 2 des notes de cours, il pourrait vous être utile de vous en procurer quelques-uns; les principaux sont les suivants :

- Swinnen, G. *Apprendre à programmer avec Python 3*, version 5, disponible gratuitement à <http://inforef.be/swi/python.htm>
- Langtangen, H.P., *A Primer on Scientific Programming with Python*, 5^e éd., Springer (2016). Note : une version HTML est disponible gratuitement à <http://hplgit.github.io/primer.html/doc/web/>
- Swaroop, C. H., *A byte of Python*, disponible gratuitement à <http://www.swaroopch.com/notes/python/>

Programme du cours

La physique numérique :

Modélisation, algorithmes, programmation, physique numérique

Éléments de programmation en Python

Types d'objets, opérations de base, entrées/sorties, contrôle du flux d'instructions, fonctions, code source, figures et graphiques, etc.

Dérivées, interpolation et intégration :

Représentation discrète d'une fonction, dérivée d'une fonction, interpolation, intégration

Équations différentielles ordinaires :

méthode d'Euler, méthode de Heun, pendule linéaire, pendule non linéaire

Nonlinéarité et chaos :

encore le pendule, espace de phase, pendule amorti, forcé, forcé et amorti, bifurcations, chaos

Monte Carlo :

Nombre aléatoires, fonctions de distribution, distributions non-uniformes, évaluation d'intégrales par Monte Carlo, approche à l'équilibre

Racines et optimisation :

bissection, méthode de Newton, méthode de grimpe, grimpe stochastique, régression linéaire

Marche aléatoire et diffusion :

marche aléatoire, lien avec diffusion, marche aléatoire sur réseau, agrégation, invariance d'échelle, fractales

Complexité :

complexité, modèle Tas-De-Sable, criticalité auto-régulée, modèle Feu-De-Fôret, modèle embouteillage, complexité et émergence