

Graphes et réseaux

Domaine	Ingénierie et Architecture
Filière	Informatique
Orientation	Logiciel (IL)
Mode de formation	Plein temps

Informations générales

Nom:	:	Graphes et réseaux
Identifiant:	:	GRE
Années académiques	:	2019-2020, 2020-2021
Responsable:	:	Jean-François Hêche
Charge de travail:	:	120 heures d'études
Périodes encadrées:	:	80 (= 60 heures)

Semestre	E1	S1	S2	E2	S3	S4	E3	S5	S6
Cours					80				

Connaissances préalables recommandées

L'étudiant doit connaître et savoir utiliser les notions suivantes :

- ensembles, relations et fonctions, calcul de séries élémentaires et manipulation du symbole somme, calcul matriciel ;
- pseudocode, comportement asymptotique des fonctions (notation de Landau), complexité d'un algorithme ;
- propriétés des structures de données linéaires et arborescentes simples (tableaux, listes, tas).

Les unités d'enseignement MAD (mathématiques discrètes), MBT (mathématiques de base pour TIC) et ASD1 (algorithmes et structures de données 1) permettent d'acquérir ces connaissances.

Objectifs

A l'issue de cette unité d'enseignement, l'étudiant-e sera capable de :

- énoncer/discuter les notions de théorie des graphes du cours (graphe « orienté/non orienté », « arête/arc », « chaîne/chemin », « cycle/circuit », « biparti », « complet », « simple », « eulérien », etc) et les appliquer à la caractérisation d'un graphe donné ;
- passer d'une représentation d'un graphe à une autre (graphique, matricielle, par listes) et discuter les avantages et inconvénients des représentations informatiques classiques ;
- décrire les problèmes classiques de la théorie des graphes et énoncer, appliquer et discuter le fonctionnement des algorithmes de résolution étudiés :
 - parcours de graphes (en largeur ou en profondeur et leurs applications : tri topologique, composantes connexes/fortement connexes),
 - arbres et arborescences optimaux (arbres recouvrants de poids minimal, algorithme de Kruskal et de Prim, arborescences recouvrantes de poids minimal, algorithme de Chu-Liu, chaînes et chemins de section optimale),
 - plus courts chemins (algorithmes de Dijkstra, de Bellman-Ford, de Johnson, de Floyd-

- Warshall ou de Dantzig, ...),
- ordonnancement (tri topologique, graphe potentiel-tâche, chemin critique, applications et extensions),
- flots dans les réseaux (flot max, coupe min, algorithme de Ford-Fulkerson, flot à coût min, algorithme de Busacker et Gowen, variantes et applications à des problèmes particuliers : couplages, postier chinois, ... ;
- lire des algorithmes sous forme d'un pseudocode et discuter leur complexité ;
- modéliser une situation donnée à l'aide d'un graphe, identifier le problème de graphes à résoudre et, le cas échéant, appliquer un algorithme permettant de le résoudre (ce point constitue l'un des objectifs majeurs du cours).

Contenu et formes d'enseignement

Répartition des périodes indiquée à titre informatif.

Cours: 80 périodes

- Définitions et notions de base	4
- Représentations des graphes : matrices et listes	2
- Parcours en largeur et en profondeur et applications à la recherche de composantes	6
- Arbres et arborescences : arbres recouvrants optimaux, chaînes et chemins de section optimale, arborescence recouvrante optimale, algorithmes et applications	10
- Plus courts chemins : poids positifs ou nuls, poids quelconques et circuits absorbants, technique de repondération, algorithmes matriciels	18
- Graphes sans circuits : caractéristiques, tri topologique, application à l'ordonnancement (graphe potentiel-tâche, chemin critique, ...)	8
- Graphes eulériens : caractérisations, recherche de parcours eulériens, problème du postier chinois	8
- Flots dans les réseaux : flot de valeur max, coupe de capacité min, flot à coût min, algorithmes et applications (couplages, transbordement, ...)	24

Bibliographie

- Ravindra K. Ahuja, Thomas L. Magnanti, James B. Orlin, Network Flows: Theory, Algorithms, and Applications, 1993, Prentice Hall.
- Thomas Cormen, Charles Leiserson, Ronald Rivest, Clifford Stein, Introduction à l'algorithmique, 2002, Sciences sup, Dunod, Paris.
- Reinhard Diestel, Graph Theory, 2012, Springer-Verlag, Heidelberg.
- Michel Gondran, Michel Minoux, Graphes et algorithmes, 2009, Lavoisier, Cachan.
- Jon Kleinberg, Éva Tardos, Algorithm Design, 2005, Pearson.
- Robert Sedgwick, Kevin Wayne, Algorithms, 2011, Addison-Wesley.

Contrôle de connaissances

Cours:

l'acquisition des matières de cet enseignement sera contrôlée au fur et à mesure par des tests et des travaux personnels tout au long de son déroulement. Il y aura au moins 3 tests d'une durée totale d'au moins 5 périodes.

Examen:

L'atteinte de l'ensemble des objectifs de formation sera vérifiée lors d'un contrôle final commun écrit d'une durée de 60 minutes.

Matériel autorisé:

- Information communiquée directement par l'enseignant.

Calcul de la note finale

Note finale = moyenne cours x 0.5 + moyenne examen x 0.5

Fiche validée le 17.09.2019 par Donini Pier