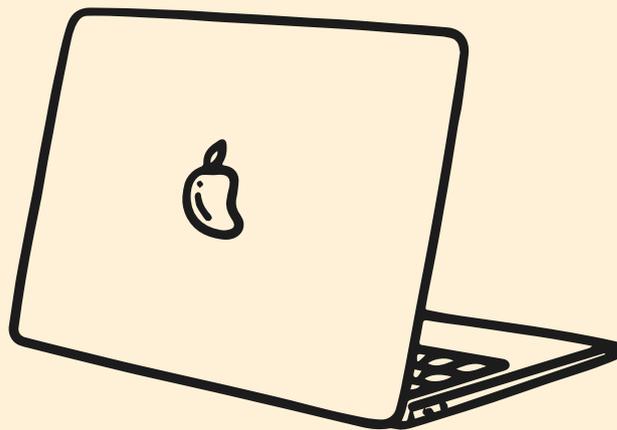


Licence 2

Sciences, Technologies, Santé

2025-2026

informatique



L2 info

Ua¹ **FACULTÉ
DES SCIENCES**
UNIVERSITÉ D'ANGERS



SOMMAIRE

CONTENUS

03

Contacts de la formation

04

Calendrier

05

Volumes horaires et évaluations

06

Contenu des enseignements

10

Informatique

12

Transversaux

Sommaire interactif
pour revenir au sommaire
cliquer sur 



CONTACTS

Sandrine TRAVIER : Directrice Adjointe à la Pédagogie

sandrine.travier@univ-angers.fr

David GENEST : Directeur des études du portail MI

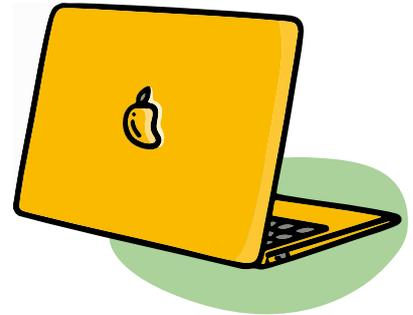
david.genest@univ-angers.fr

Sylvain LAMPRIER : Responsable pédagogique

sylvain.lamprier@univ-angers.fr

Gestion de la scolarité et des examens

l2info.sciences@contact.univ-angers.fr



Scolarité - Examens

Bâtiment A, Rez-de-chaussée

Horaires

8h30 > 12h30

13h30 > 16h30

Du lundi au vendredi

Fermé le mercredi après-midi



Calendrier L2 Maths, L2 Info DL ME, DL MI, PPPE 2025-2026

Rentrée Lundi 08 septembre 2025

P6

du 08 sept. au 24 oct.

EXAM! CC
20-24 octobre

★ Jeudi 18 sept.
Campus Day

★★ Vacances
du 24 oct. au soir
au dimanche 02 nov.

P7

du 03 nov. au 19 déc.

EXAM! CC
15-19 décembre

★★ Vacances
du 19 déc. au soir
au dimanche 04 jan.

P8

du 05 jan. au 13 fév.

EXAM! CC
09-13 février

 22-23 janvier
Inscription 2^{de} chance
P6-P7

2^{de} chance P6-P7
16-20 février

★★ Vacances
du 20 fév. au soir
au dimanche 01 mars

P9

du 02 mars au 17 avril

EXAM! CC
13-17 avril

★★ Vacances
du 17 avril au soir
au dimanche 26 avril

P10

du 27 avril au 03 juin

EXAM! CC
01-03 juin

 15-16 juin
inscription 2^{de} chance
P8-P9-P10

2^{de} chance P8-P9-P10
19 juin, 22-26 juin

JURY


08
JUIL.

sous réserve de modifications



VOLUMES HORAIRES - ÉVALUATIONS

Période	Intitulés	Volumes horaires				ECTS & Coef.	Chance	
		CM	TD	TP	Total		Chance 1 DA idem	Chance 2 DA idem
Algèbre linéaire						7		
P6	Algèbre linéaire	12,0	16,0		28,0	7	CC 33% - 2h	CT 100% - 2h30
P7	Algèbre linéaire	12,0	20,0		32,0		CC 67% - 2h30	
Algorithmique et programmation						16	Note plancher 6	
Algorithmique 3								
P6	Algorithmique 3	6,7	9,3	10,7	26,7	8	CC 50% - 1h30	CT 100% - 1h30
P7	Algorithmique 3	6,7	9,3	12,0	28,0		CC 50% - 1h30	
Programmation orientée objet 1								
P8	Programmation orientée objet	16,0	8,0		24,0	8	CC 20% - 1h	CT 60% - 1h TP P10 40% 1
P9	Programmation orientée objet	9,3	10,7	4,0	24,0		CC 40% - 1h	
P10	Programmation orientée objet		4,0	16,0	20,0		TP 40% - 2h	
Fondements et théorie de l'informatique						14	Note plancher 6	
Fondements de l'informatique 2								
P6	Fondements de l'informatique 2	12,0	12,0		24,0	6	CC 30% - 1h30	CT 100% - 1h30
P7	Fondements de l'informatique 2	12,0	12,0		24,0		CC 70% - 1h30	
Théorie des langages 1								
P8	Théorie des langages 1	9,3	14,7		24,0	6	CC 50% - 1h30	CT 100% - 1h30
P9	Théorie des langages 1	8,0	12,0		20,0		CC 50% - 1h30	
Fondements de l'informatique 3								
P10	Fondements de l'informatique 3	8,0	8,0		16,0	2	CC 100% - 0h45	CT 100% - 0h45
Technologie de l'informatique						16	Note plancher 6	
Bases de données 2								
P6	Bases de données 2	6,7	9,3		16,0	4	CC 50% - 1h	CT 100% - 1h30
P7	Bases de données 2	4,0		12,0	16,0		CC 50% - 1h	
Développement web 2								
P8	Développement web 2	6,7		9,3	16,0	6	CC 30% - 1h30	CT 100% - 2h
P9	Développement web 2	6,7		9,3	16,0		CC 30% - 1h30	
P10	Développement web 2	6,7		9,3	16,0		CC 40% - 1h30	
Systèmes GNU/Linux et Bash								
P8	Systèmes GNU/Linux et Bash	8,0		16,0	24,0	3	CC 100% - 2h	CT 100% - 2h
Systèmes								
P9	Systèmes	12,0	12,0		24,0	3	CC 100% - 1h30	CT 100% - 1h
Transversaux						7		
Anglais								
P6	Anglais 1			8,0	8,0	2	CC 100% - 1h20	CT 100% - 1h
P7	Anglais 1			8,0	8,0			
P8	Anglais 2			8,0	8,0	2	CC 100% - 1h20	
P9	Anglais 2			8,0	8,0			
Projet personnel et professionnel								
P6	3PE		8,0		8,0	3	Assiduité 10%	Pas de seconde chance 2
P7	3PE	8,0			8,0		CC QCM 30%	
P8	3PE	2,7	5,3		8,0		Rapport 30%	
P9	3PE			4,0	4,0		Rapport 30%	
TOTAL		173,5	170,6	134,6	478,7	60		

1 La note de TP de P10 est reportée en seconde chance.

2 L'évaluation est faite par la présence (1), un QCM qui sera neutralisé en cas d'ABJ (2), un rapport qui peut être remis plus tard en cas d'ABJ (3 et 4).



Conditions de validation de l'année

Moyenne générale supérieure ou égale à 10 ET Moyenne de bloc supérieure ou égale à la note plancher pour les blocs ayant une note plancher.
Acquisition des ECTS par UE (note à l'UE ≥ 10) ou par bloc (si note de bloc ≥ 10 , les unités composant le bloc sont acquises)

CM> Cours magistraux

TD> Travaux Dirigés

CM/TD> Cours magistraux et Travaux dirigés intégrés

TP> Travaux Pratiques

CC> Contrôle continu

CT> Contrôle terminal



CONTENU DES ENSEIGNEMENTS

Période 6

	Page
Algorithmique 3	07
Fondements de l'informatique 2	07
Bases de données 2	09
Anglais 3	11
Projet personnel et professionnel	11
Algèbre linéaire	11

Période 7

	Page
Algorithmique 3	07
Fondements de l'informatique 2	07
Bases de données 2	09
Anglais 3	11
Projet personnel et professionnel	11
Algèbre linéaire	11

Période 8

	Page
Programmation orientée objet	07
Théorie des langages	08
Développement Web 2	09
Système GNU/Linux	09
Anglais 4	11
Projet personnel et professionnel	11

Période 9

	Page
Programmation orientée objet	07
Théorie des langages	08
Développement Web 2	08
Systèmes	09
Anglais 4	11
Projet personnel et professionnel	11

Période 10

	Page
Programmation orientée objet	07
Fondements de l'informatique 3	08
Développement Web 2	08

Index interactif
pour revenir utiliser
sur les pages 



P6

P7

ALGORITHMIQUE 3

Responsable [Adrien Goëffon](#)

PROGRAMME

Types abstraits et structures de données avancées : listes chaînées, piles, files, arbres binaires, arbres binaires de recherche, arbres équilibrés, tas, tables de hachage, graphes.

Implémentation d'algorithmes dans le langage C++.

COMPÉTENCES

Maîtriser les principales notions de de l'algorithmique impérative. Définir des algorithmes efficaces en organisant les données au moyen de structures appropriées.

P8

P9

P10

PROGRAMMATION ORIENTÉE OBJET

Responsable [Stéphane Loiseau](#)

PROGRAMME

Il s'agit de comprendre ce qu'est l'approche objet, principalement la programmation objet, et de savoir la mettre en œuvre dans le cadre de programmes simples.

Une partie de l'enseignement portera sur les langages de programmation. Nous nous interrogerons sur ce qu'est un langage de programmation, nous proposerons une classification des langages de programmation, nous montrerons les intérêts des principaux langages et syntaxes.

Le modèle objet sera abordé au travers de la nécessité d'organiser des concepts pour modéliser et de son opérationnalisation pour programmer efficacement.

La programmation objet sera présentée selon deux axes principaux. Le premier concerne les principes : nous présenterons les deux

principes à la base de la programmation objet, à savoir le principe de la classe et celui de la généralisation/spécialisation. Le second axe concerne la qualité : nous présenterons les mécanismes de base de la programmation objet de qualité (constructeur, modificateur, classe abstraite, interface, exception, Thread etc.).

Le langage de programmation JAVA sera étudié : les tp seront en JAVA, le cours sera en pseudo SmallTalk (LOLO) et JAVA.

La notation UML sera abordée.

COMPÉTENCES

Être en mesure de représenter un nombre entier ou réel en binaire. Être en mesure d'exprimer une fonction booléenne à partir de sa table de vérité et de la simplifier algébriquement.

P6

P7

FONDEMENTS DE L'INFORMATIQUE 2

Responsable [Igor Stephan](#)

PROGRAMME

Logique propositionnelle : morphologie ; sémantique (interprétation, satisfiabilité, insatisfiabilité, tautologie, table de vérité), relation d'équivalence et formes normales, complétude fonctionnelle, conséquence sémantique (modélisation de problèmes) et méthodes sémantiques (arbre sémantique et propagation) ; méthodes syntaxiques (analytique : la méthode des tableaux, axiomatique : système de Hilbert, synthétique : le calcul des séquents), propriétés de correction et complétude (vis-à-vis de la sémantique) et terminaison d'une méthode syntaxique.

Logique du premier ordre : morphologie ; sémantique (interprétation dans une structure, vérité versus validité), relation d'équivalence et formes normales, conséquence sémantique (modélisation de problèmes), forme normale de Skolem, théorème de



Skolem–Herbrand–Gödel ; méthodes syntaxiques : l'unification et la Résolution de Robinson.

P8

P9

THÉORIE DES LANGAGES

Responsable [Claire Lefèvre](#)

PROGRAMME

Langages formels, opérations sur les langages, automates à états finis déterministes, non déterministes, déterminisation, minimisation, expressions régulières, algorithmes de recherche de motifs, grammaires non contextuelles

COMPÉTENCES

- Comprendre la notion de langage formel et savoir réaliser des opérations sur ceux-ci.
- Connaître les automates à états finis déterministes et non déterministes.
- Savoir déterminer des automates non déterministes.
- Manipuler et concevoir des expressions régulières.
- Maîtriser les algorithmes de recherche de motifs.
- Comprendre les grammaires non contextuelles.

P10

FONDEMENTS DE L'INFORMATIQUE 3

Responsable [Igor Stephan](#)

PROGRAMME

Cette UE est une introduction à la théorie de la calculabilité et de la complexité. Elle a pour but de présenter les limites du calcul automatique : différence entre problème et programme, classes de langage, machines de Turing, décidabilité, indécidabilité, problème de l'arrêt.

COMPÉTENCES

Comprendre les limites de l'informatique et des différents langages de programmation.

P8

P9

P10

DÉVELOPPEMENT WEB 2

Responsable [David Lesaint](#)

PROGRAMME

Apprentissage de PHP, fonctions de base, manipulation des tableaux classiques et associatifs, des chaînes de caractères et des expressions régulières, programmation côté serveur, accès aux bases de données.

Apprentissage de Javascript, aspects fonctionnel et objet, manipulation du DOM, requêtes asynchrones AJAX.

COMPÉTENCES

- Comprendre le concept de programmation côté serveur.
- Connaître le langage de programmation PHP (gestion des valeurs issues de formulaires, gestion de tableaux traditionnels et de tableaux associatifs, de chaînes de caractères et d'expressions régulières, interface avec une base de données).
- Connaître le langage de programmation Javascript (aspects fonctionnel et objet).
- Comprendre l'arbre DOM d'un document HTML. Savoir manipuler cet arbre avec Javascript et réaliser la validation des entrées de formulaires.
- Savoir réaliser des requêtes asynchrones (AJAX).
- Savoir utiliser conjointement ces technologies afin de réaliser des pages dynamiques.



BASES DE DONNÉES 2

Responsable [Laurent Garcia](#)

PROGRAMME

Cette UE concerne la conception de bases de données relationnelles.

Outils de conception d'un système d'information : dépendances fonctionnelles, graphe des dépendances fonctionnelles, formes normales, schéma Entité/ Association. Méthode MERISE de conception des systèmes d'information et application sur des études de cas complètes. Syntaxe des requêtes SQL de création et suppression de tables et de vues. Syntaxe SQL des types et des contraintes (clés, valeurs, ...). Syntaxe SQL des requêtes de manipulation des données : insertion, mise-à-jour et suppression de n-uplets. Mise en œuvre pratique de la création et de la manipulation de bases de données par l'écriture de requêtes en langage SQL en utilisant le Systèmes de Gestion de Bases de Données (SGBD) PostgreSQL.

COMPÉTENCES

Maîtriser la méthode MERISE, méthode de conception des systèmes d'information, et les outils utilisés (dépendances fonctionnelles, graphe des dépendances fonctionnelles, formes normales, décomposition de relations, schéma Entité/ Association). Savoir mettre en œuvre cette méthode sur des cas pratiques complexes. Connaître la syntaxe du langage SQL et savoir mettre en œuvre en pratique des requêtes de description et de manipulation des données dans le langage SQL : savoir créer et modifier une base de données relationnelle en définissant les relations et leurs attributs, leurs clés et leurs contraintes d'intégrité ainsi que savoir gérer les informations d'une base de données (ajouter, mettre à jour et supprimer des données).

SYSTÈMES GNU/LINUX ET BASH

Responsable [Benoît Da Mota](#)

PROGRAMME

Théorie, fonctionnement, utilisations avancées et administration des systèmes GNU/Linux. Philosophie et histoire des systèmes GNU/Linux, principes généraux du chiffrement. Utilisation avancée en ligne de commandes, scripts, outils du développeur, administration.

COMPÉTENCES

- Comprendre l'histoire et la philosophie des systèmes GNU/Linux et exploiter leur potentiel en ligne de commandes (fichiers, filtres, flux, redirections, etc.). Maîtriser les commandes usuelles du Shell Linux et être autonome pour trouver et exploiter la documentation en ligne pour les commandes moins usuelles.
- Savoir réaliser des scripts BASH.
- Connaître les bases de l'administration d'un système (installation, configuration, services, maintenance, intégration réseau, virtualisation et conteneurs logiciels).
- Être capable d'identifier le sens et le lien entre les termes : exécutable, bibliothèque, compilation, édition de liens, etc.
- Comprendre les principes généraux et les utilisations possibles des techniques de chiffrement (illustration via HTTPS et SSH).

SYSTÈMES

Responsable [Jin Kao-Hao](#)

PROGRAMME

Systèmes d'exploitation : théorie, fonctionnement des systèmes. Les fonctions d'un OS, structures matérielles, gestion de la mémoire centrale, gestion des entrées et sorties, gestion de processus, exclusion mutuelle, processus concurrents, gestion de fichiers.

COMPÉTENCES

- Connaître les fonctions d'un système

d'exploitation.

– Connaître l'origine des notions comme le traitement par lot, la multiprogrammation et leur évolution.

– Connaître la structure matérielle d'un ordinateur, et les ordres de grandeur des temps d'accès et des capacités des différents types de mémoire.

– Connaître la gestion des processus sous Linux/Unix (états et transitions, envoi de signaux, etc.).

– Connaître et savoir appliquer les différentes stratégies d'ordonnancement des processus (round robin, priorités).

– Connaître le système de fichiers, les méthodes d'accès à la mémoire secondaire, les modes de représentation de l'espace libre.

– Connaître les méthodes d'accès à la mémoire centrale, son organisation (pagination, segmentation, pagination à la demande).

– Connaître et savoir éviter les situations d'interblocage de processus, savoir appliquer l'algorithme du banquier.

– Connaître les problématiques de l'exclusion mutuelle, de la communication inter-processus et des processus concurrents.

TRANSVERSAUX

P6

P7

P8

P9

ANGLAIS

Responsable **Philippe Torres**

PROGRAMME

Objectifs du cours d'anglais :

- Permettre aux étudiants de continuer à travailler les cinq compétences en langue (Compréhension écrite et orale, expression écrite et orale, et interaction orale) à travers des supports authentiques (articles, documents, documents audio et vidéo d'internet, graphiques...) et des activités variées (exercices de compréhension, d'expression écrite, jeux de rôle, débats, présentations orales...).
- Étoffer les connaissances lexicales.
- Améliorer la prononciation (bases de phonologie).
- Revoir et comprendre des points de langue (les temps par exemple).

COMPÉTENCES

En fin de licence, on vise le niveau de compétence B2 du Cadre Européen Commun de Référence pour les Langues (CECRL) qui est résumé comme suit : « Peut comprendre le contenu essentiel de sujets concrets ou abstraits dans un texte complexe, y compris une discussion technique dans sa spécialité. Peut communiquer avec un degré de spontanéité et d'aisance tel qu'une conversation avec un locuteur natif ne comporte de tension ni pour l'un ni pour l'autre. Peut s'exprimer de façon claire et détaillée sur une grande gamme de sujets, émettre un avis sur un sujet d'actualité et exposer les avantages et les inconvénients de différentes possibilités».

P6

P7

P8

P9

PROJET PERSONNEL ET PROFESSIONNEL

P6

P7

ALGÈBRE LINÉAIRE

Responsables **Jean-Philippe Monnier, Daniel Naie**

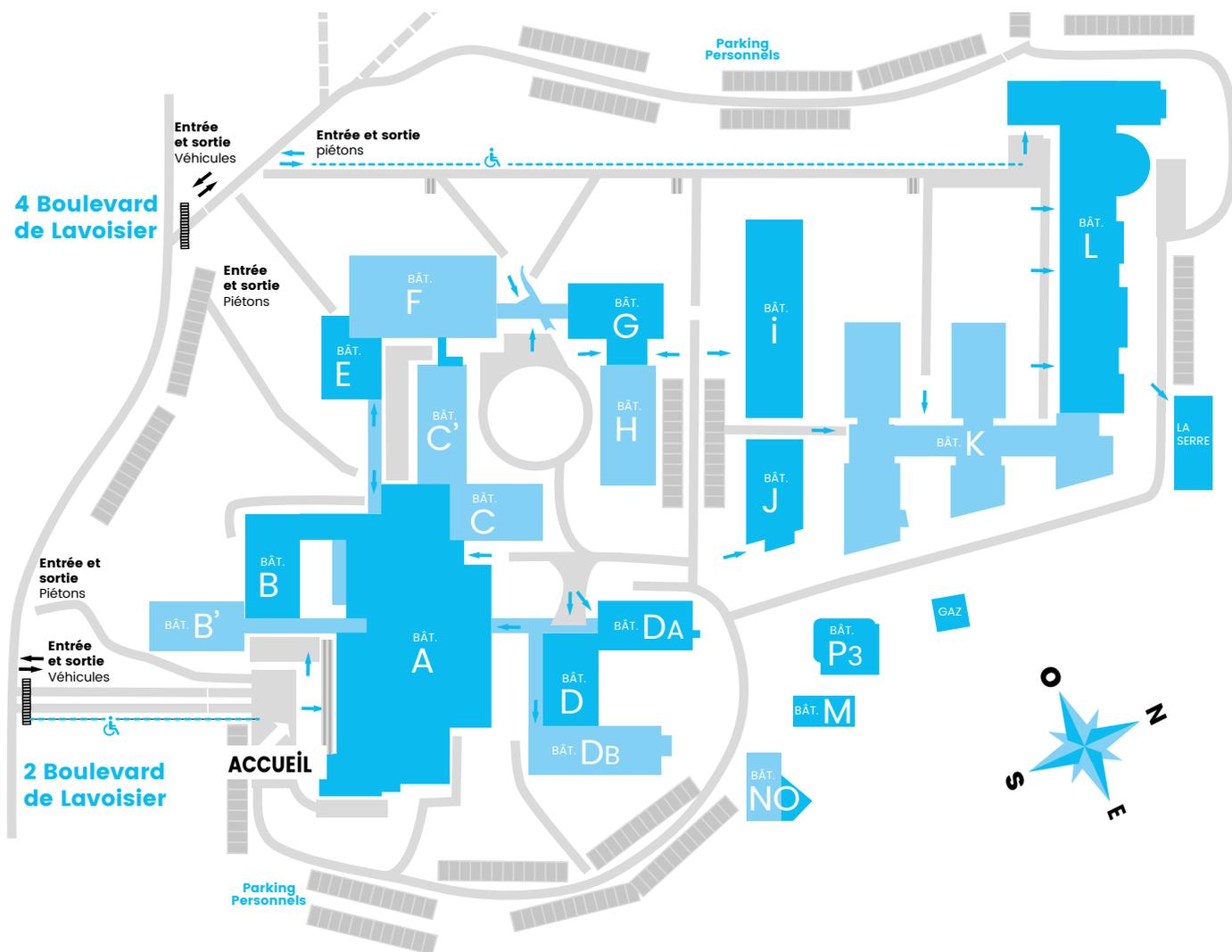
PROGRAMME

- Espaces vectoriels. Combinaisons linéaires, bases et dimension, rang.
- Sous-espaces vectoriels. Coordonnées, équations d'un sous-espace. Théorème de la base incomplète.
- Applications linéaires. Rang, noyau, image. Changement de base. Théorème du rang.
- Représentation matricielle.
- Utilisation de la méthode du pivot pour les calculs explicite revues en L1.

COMPÉTENCES

- Savoir prouver qu'un ensemble est un espace vectoriel.
- Savoir déterminer le rang d'un système de vecteurs.
- Comprendre la notion d'indépendance linéaire.
- Savoir reconnaître une application linéaire, déterminer son noyau et son image.
- Savoir écrire la matrice d'une application linéaire relativement à des bases données, et déterminer son rang.





- A** Scolarité | Accueil | Enseignement (Amphi A à E) | Administration
- B** Enseignement biologie
- B'** Enseignement biologie
- C** Enseignement chimie
- C'** Recherche
- D** Enseignement physique
- Da** Enseignement physique
- Db** Recherche
- E** Enseignement biologie
- F** Enseignement biologie | Recherche
- G** Enseignement géologie | informatique
- H** Enseignement informatique | Recherche
- I** Enseignement mathématiques | Recherche
- J** Enseignement chimie
- K** Recherche
- L** Enseignement transversaux | Enseignement (Amphi L001 à L006)

Impression Service Reprographie UA

