

D E U G

**D I P L Ô M E
D ' É T U D E S**

U N I V E R S I T A I R E S G É N É R A L E S

**P R O G R A M M E
D E S É T U D E S**

SOMMAIRE

PRÉSENTATION GÉNÉRALE	1
PROGRAMME DES ÉTUDES	1
Organisation des études.....	1
Enseignements d'informatique proposés.....	1
Organisation par mention.....	2
<i>Mentions MASS et MIAS</i>	2
<i>Mention SM</i>	3
<i>Mention STPI</i>	3
<i>Mention SV</i>	3
Enseignements disciplinaires de base	3
<i>AF1+AF2 - Algorithmique et programmation : approche fonctionnelle</i>	3
<i>API - Méthodes et outils de l'informatique : approche impérative</i>	4
<i>IS1-IS2 - Programmation d'un calculateur scientifique : approche fonctionnelle</i>	4
<i>IS1+PFI - Programmation d'un calculateur scientifique : approches fonctionnelle et impérative</i>	4
Enseignements optionnels.....	5
<i>GEN - Génie logiciel : buts, méthodes et outils</i>	5
<i>CSR - Conception de systèmes réactifs : buts, méthodes et outils</i>	5
Enseignements méthodologiques.....	5
<i>TIB - Traitement des informations biologiques</i>	5
ADMISSION	6
MATIÈRES ENSEIGNÉES	6
<i>DEUG1 / AF1 : Algorithmique et programmation : approche fonctionnelle (1)</i>	7
<i>DEUG1 / AF2 : Algorithmique et programmation : approche fonctionnelle (2)</i>	8
<i>DEUG1 / IS1 : Programmation d'un calculateur scientifique : approche fonctionnelle (1)</i>	9
<i>DEUG1 / TIB : Traitement des informations biologiques</i>	10
<i>DEUG2 / API : Algorithmique et programmation : approche impérative</i>	11
<i>DEUG2 / GEN : Génie logiciel : buts, méthodes et outils</i>	12
<i>DEUG2 / IS2 : Programmation d'un calculateur scientifique : approche fonctionnelle (2)</i>	13
<i>DEUG2 / PFI : Programmation fonctionnelle et impérative</i>	14
<i>DEUG2 / CSR : Conception de systèmes réactifs : but, méthodes et outils</i>	15

PRÉSENTATION GÉNÉRALE

Le Diplôme d'Études Universitaires Générales (DEUG) est un diplôme national que l'on peut obtenir à l'issue des deux premières années d'études à l'université. Ce diplôme a pour but de donner une formation fondamentale permettant de poursuivre des études de deuxième cycle.

À l'heure actuelle, l'Université de Rennes 1 met en oeuvre cinq des six mentions nationales du DEUG Sciences et Technologies ; il s'agit des mentions :

- Mathématiques Appliquées aux Sciences Sociales (Mention MASS)
- Mathématiques, Informatique et Applications aux Sciences (Mention MIAS)
- Sciences de la Matière (Mention SM)
- Sciences et Techniques pour l'ingénieur (Mention STPI)
- Sciences de la Vie (Mention SV).

Ce DEUG est organisée en unités d'enseignement (UE) capitalisables. La partie informatique de ces unités est sous la responsabilité de l'IFSIC.

PROGRAMME DES ÉTUDES

Organisation des études

Le DEUG Sciences et Technologies est organisé en unités d'enseignement capitalisables (12 à 14 UE réparties sur 2 années). Une UE est dispensée sur un semestre. Chaque unité d'enseignement constitue un regroupement d'enseignements et d'activités.

Au sein d'une unité, un enseignement d'informatique comporte, en proportions sensiblement équilibrées, des cours magistraux (CM), des travaux dirigés (TD) et des travaux pratiques (TP) .

Enseignements d'informatique proposés

Nous proposons une batterie de 9 enseignements d'informatique:

- AF1 - Algorithmique et programmation : approche fonctionnelle (1)
- AF2 - Algorithmique et programmation : approche fonctionnelle (2)
- API - Algorithmique et programmation : approche impérative

- GEN - Génie logiciel : buts, méthodes et outils
- IS1 - Programmation d'un calculateur scientifique: approche fonctionnelle (1)
- IS2 - Programmation d'un calculateur scientifique: approche fonctionnelle (2)
- PFI - Programmation d'un calculateur scientifique: approche fonctionnelle et impérative
- CSR - Conception de systèmes réactifs : buts, méthodes et outils
- TIB - Traitement des informations biologiques

Les enseignements de base sont AF1-AF2-API d'une part, IS1-IS2 ou IS1-PFI d'autre part ; ils visent à donner les bases minimum en informatique, nécessaires à tout scientifique. A l'issue de ces enseignements, un étudiant doit:

- savoir analyser un problème et être capable d'exprimer et mettre en oeuvre un algorithme pour le résoudre ;
- posséder des notions de base sur les machines et les systèmes informatiques ;
- avoir une idée claire de la discipline informatique : histoire, domaines d'application, place dans la société, perspectives d'évolution, ...

Si les objectifs poursuivis sont les mêmes, la mise en oeuvre, compte-tenu du public visé et des volumes horaires, est très différente selon la mention. Par ailleurs, des enseignements optionnels permettent aux étudiants, soit de compléter leur formation, soit de préparer une poursuite d'études en informatique.

La réforme du DEUG Sciences et Technologies prévoit des enseignements dits « méthodologiques » : ainsi, les étudiants de la mention SV suivent-ils l'enseignement intitulé TIB, qui entre dans cette catégorie.

Organisation par mention

Mentions MASS et MIAS

Ces deux mentions donnent accès de droit à la licence d'informatique. Les étudiants suivent donc obligatoirement trois enseignements disciplinaires :

- AF1 et AF2 - Algorithmique et programmation : approche fonctionnelle
- API - Méthodes et outils de l'informatique : approche impérative. En complément, les étudiants peuvent suivre l'option GEN, recommandée pour

- une poursuite d'études en informatique.
- GEN - Génie logiciel : buts, méthodes et outils

Mention SM

Les étudiants suivent obligatoirement deux enseignements d'informatique dispensés au premier semestre de chacune des deux années:

- IS1 - Programmation d'un calculateur scientifique: approche fonctionnelle (1)
- IS2 - Programmation d'un calculateur scientifique: approche fonctionnelle (2)

Mention STPI

Cette mention donnant également accès de droit à la licence d'informatique, les étudiants suivent obligatoirement deux enseignements disciplinaires :

- IS1 - Programmation d'un calculateur scientifique: approche fonctionnelle (1)
- PFI - Programmation d'un calculateur scientifique: approche fonctionnelle et impérative

Un enseignement optionnel leur est proposé en seconde année, recommandé pour une poursuite d'études en informatique:

- CSR - Conception de systèmes réactifs : buts, méthodes et outils

Mention SV

Les étudiants suivent obligatoirement en première année:

- TIB - Traitement des informations biologiques

Enseignements disciplinaires de base

AF1+AF2 - Algorithmique et programmation : approche fonctionnelle

Il s'agit d'un enseignement de première initiation à l'algorithmique et à la programmation. Les objectifs visés sont les suivants :

- Comprendre ce qu'est un algorithme et la démarche de conception/ expression/mise en oeuvre.
- Donner à l'étudiant de bons « réflexes » : abstraction (abstraction

fonctionnelle et abstraction de données), conception de « récurrences ». Compte tenu de ces objectifs, un certain nombre de grands choix (positifs mais aussi négatifs) sont faits :

- Utiliser un langage fonctionnel comme support. Il s'agit de Scheme auquel on ajoute, sur papier, un système de typage.
- Ne pas utiliser de variables (donc se passer de l'itération) et se focaliser sur la conception d'algorithmes récursifs.
- Faire l'impasse sur les notions de systèmes, d'E/S.

API - Méthodes et outils de l'informatique : approche impérative

API fait suite aux deux enseignements précédents ; il s'agit de :

- Approfondir la notion d'abstraction en élargissant la gamme des structures de données disponibles.
- Introduire les notions de variable et d'itération.
- Introduire les E/S.
- Présenter un langage impératif typé (en l'occurrence Pascal).
- Introduire la notion d'objet.

La notion de type abstrait (couches d'abstraction) est centrale.

IS1+IS2 - Programmation d'un calculateur scientifique : approche fonctionnelle

Le calcul semble nécessaire à tout scientifique et nous pensons que l'introduction d'un outil de calcul formel en DEUG est utile. Un langage comme Mathematica offre des possibilités intéressantes. Cet enseignement est proposé aux étudiants de la mention SM. Il s'agit de :

- Montrer les possibilités d'un outil moderne de calcul (calculs exacts, intégrales, résolution d'équations, manipulation de formules).
- Évoquer les structures de données informatiques aptes à modéliser les objets du calcul formel.
- Initier l'étudiant à l'algorithmique et à la programmation en utilisant le langage du système formel.

IS1+PFI - Programmation d'un calculateur scientifique : approches fonctionnelle et impérative

Compte-tenu de la spécificité du public STPI, ces deux enseignements ont un contenu similaire à IS1 + IS2 en ce qui concerne l'approche fonctionnelle, mais abordent également la programmation impérative, incontournable pour une poursuite d'étude en informatique.

Enseignements optionnels

GEN - Génie logiciel : buts, méthodes et outils

L'option GEN est vivement recommandée à ceux qui envisagent une poursuite d'études en informatique. Les buts visés sont les suivants :

- Comprendre qu'un « gros » logiciel demande des méthodes de développement spécifiques.
- Comprendre que la programmation fait appel à des connaissances, en particulier mathématiques, adaptées. Trois grands axes ont été retenus et reliés entre eux d'une manière qui se veut originale.
- L'étudiant réalise 2 TP (pas plus !) mais ces derniers sont des modules d'un logiciel assez gros (2 000 lignes de Scheme) qui représente un projet. Au début de l'année, il récupère une documentation du projet (spécifications détaillées) et le code objet de tous les modules. Chaque TP, qui a une interface spécifiée et qui utilise d'autres modules, passe donc par les phases « test unitaire » et « intégration ».
- Théorie des langages (généralités et langages rationnels). Cet axe est indispensable pour concevoir et réaliser le TP2.
- Logique (généralités sur les systèmes formels et calcul des propositions). Cet axe permet de comprendre les tenants et aboutissants du projet ainsi que ses limitations (le projet est un moteur d'inférence). Il permet également, mais a posteriori, d'éclairer les choix du TP1.

CSR - Conception de systèmes réactifs : buts, méthodes et outils

Dans cet enseignement, l'étudiant réintègre les acquis de IS1 + PFI tant sur le plan approche fonctionnelle pour la description des systèmes que sur le plan impératif pour la partie calcul et traitement des données. Pour ce faire, l'étudiant réalise un projet de taille conséquente autour du pilotage automatique d'un robot mobile ; dans ce projet, la notion de programmation par automates est centrale.

Enseignements méthodologiques

TIB - Traitement des informations biologiques

Cet enseignement obligatoire s'adresse aux étudiants de la mention SV, à qui il permet d'utiliser l'informatique dans le cadre de leur première compétence, la biologie. Le logiciel Mathematica est utilisé pour représenter, traiter et visualiser des données biologiques, sans qu'il soit

nécessaire de programmer. Cet enseignement vise à donner aux étudiants des connaissances de base sur l'utilisation d'un ordinateur et à leur montrer l'intérêt du traitement automatique des données.

ADMISSION

Le DEUG Sciences et Technologies est accessible aux bacheliers scientifiques. L'orientation définitive dans l'une des mentions est faite à l'issue du premier semestre, des passerelles étant prévues entre les différentes mentions pour une réorientation éventuelle. On peut noter que les mentions SM et STPI ont en commun leur premier semestre.

MATIÈRES ENSEIGNÉES

Les différentes matières sont décrites dans les pages suivantes : ne page par matière qui comprend une présentation, un plan, le sigle de la matière et le nombre d'heures d'enseignement.

DEUG Sciences 1er niveau

Algorithmique et programmation : approche fonctionnelle (1)

La première initiation à l'informatique comporte deux modules (AF1 et AF2) étroitement liés. Les principaux buts poursuivis sont (1) découvrir les algorithmes et les machines informatiques, (2) adopter une approche méthodologique de résolution de problème informatique en quatre temps : spécification, conception d'un algorithme, codage, mise au point, (3) utiliser une approche descendante pour concevoir des algorithmes, (4) apprendre une démarche rigoureuse pour concevoir des traitements répétitifs, (5) étudier quelques structures de données et algorithmes fondamentaux.

Pour ce faire, on adopte une approche fondée sur l'utilisation d'un langage fonctionnel et l'effort porte donc sur la notion de fonction, sur le principe de composition et sur l'usage de la récursivité. Une attention particulière est apportée à la démarche de conception descendante et à la notion d'abstraction (abstraction procédurale et abstraction de données). La notion de variable (au sens informatique) est volontairement exclue du programme de l'année, et les structures de données rencontrées sont donc des structures récursives.

Le langage support du cours est Scheme, un dialecte de Lisp.

PLAN

1. Informatique, machines et algorithmes
2. Fonctions et algorithmes
 - notions de base : fonctions et types ;
 - désignation et conditionnelles ;
 - premiers algorithmes.
3. Méthodologie de conception
 - spécification d'un problème et type d'une fonction ;
 - méthodologie générale de conception de programmes : démarche descendante.
4. Traitements répétitifs : la récursivité
 - méthodologie de conception de fonctions récursives et applications.

DEUG1 / AF1 : Cours : 12 heures ; TD : 12 heures ; TP : 12 heures.

MATIÈRES
ENSEIGNÉES

DEUG Sciences 1er niveau

Algorithmique et programmation : approche fonctionnelle (2)

Ce cours est la deuxième partie de l'initiation à l'algorithmique et à la programmation par le biais de la programmation fonctionnelle. Si la première partie était orientée sur la notion d'abstraction procédurale, celle-ci est dédiée à l'étude de l'abstraction de données.

Après avoir montré le besoin de création de types propres à l'utilisateur, ainsi que le caractère impératif de la séparation entre la spécification d'un type et sa mise en oeuvre, le cours présente les paradigmes de structures de données de la programmation fonctionnelle que sont les listes et les arbres. Pour chacun d'entre eux, on aborde des algorithmes classiques tels que les tris de listes, l'insertion dans un arbre etc.

PLAN

1. Structures de données et types abstraits
 - abstraction de données et mise en oeuvre ;
 - hiérarchie de machines abstraites ;
 - le type doublet comme outil de construction primitif.
2. Listes
 - liste et récursivité à droite ;
 - illustration sur des algorithmes de tri.
3. Arbres
 - terminologie ;
 - parcours préfixe, infixe et suffixe ;
 - un outil de modélisation fort.
4. Compléments sur les fonctions
 - fonctions d'ordre supérieur ;
 - complexité.

DEUG Sciences 1er niveau

Programmation d'un calculateur scientifique : approche fonctionnelle (1)

Ce cours est une initiation à la programmation d'un ordinateur en vue de son utilisation dans des filières scientifiques et techniques. Cette initiation est faite en suivant une approche fonctionnelle, à partir du logiciel Mathematica.

PLAN

1. Premier contact avec un ordinateur
 - structure physique, système d'exploitation, logiciels d'application ;
 - TP 1 : premier contact avec le PC sous Windows95, et avec le logiciel Mathematica.
2. Objets de base d'un langage de programmation
 - valeurs, types, expressions ;
 - TP (2 et 3) : évaluation d'expressions Mathematica.
3. Fonctions
 - définition, appel, paramètres formels et effectifs, signature ;
 - TP (4 et 5) : définition de fonctions et leur utilisation. Application à la trajectoire d'un projectile.
4. Expressions booléennes
 - prédicats, expressions booléennes, instructions conditionnelles ;
 - TP 6 : définition de fonctions faisant intervenir des conditions.
5. Affectation
 - environnement, liaison d'un symbole, modularité ;
 - TP final, noté.

DEUG Sciences 1er niveau

Traitement des informations biologiques

Cet enseignement permet une prise de contact avec l'ordinateur ainsi qu'une première approche du traitement informatique de données biologiques. Il comprend une brève introduction au vocabulaire de l'informatique et à l'utilisation d'un système d'exploitation. On traitera en TD et TP des exemples empruntés à la biologie, et plus particulièrement la biologie moléculaire. Le support des travaux pratiques est le logiciel Mathematica.

PLAN

1. Description et principes de fonctionnement d'un ordinateur
2. Aperçu sur les aspects logiciels
3. Bases sur l'utilisation du logiciel Mathematica (objets, expressions, fonctions, listes)
4. Mise en forme et traitement de données biologiques sous forme d'un bloc-notes Mathematica

DEUG Sciences 2ème niveau

Algorithmique et programmation : approche impérative

Il s'agit de poursuivre l'initiation faite au premier niveau en introduisant un nouveau modèle de calcul, basé sur la notion de mémoire et de séquentialité, modèle utilisé par les ordinateurs.

Ainsi, dans un premier temps, on aborde la programmation impérative en insistant sur les différences avec le modèle fonctionnel ; les variables et les principales structures de contrôle sont introduites.

La suite du cours reprend la méthode de conception vue au premier niveau, basée sur l'abstraction de données, en l'adaptant au contexte impératif. On y voit à la fois le principe de la méthode (distinction entre la spécification d'une abstraction et sa mise en oeuvre) et des éléments de langage de programmation et de système d'exploitation permettant de supporter cette méthode (type, objet, compilation séparée, édition de liens). On applique ces notions sur une structure de données classique : la liste.

Ce module permet également de donner une culture générale en architecture (notions de mémoire et de processeur) et systèmes d'exploitation (système de gestion de fichiers, interpréteur de commandes, compilation et édition de liens).

PLAN

1. Partie I

- Introduction : processeur et mémoire, programmation impérative ;
- Constructions de base du langage : variable, conditionnelle, itération. Méthode de conception d'itérations ;
- Les procédures ;
- Construction de type : type énuméré, intervalle, structure, tableau ;
- Abstraction de données : spécification et mise en oeuvre d'un type ;
- Compléments sur le langage de programmation (objets, compilation séparée) ;
- Les listes : spécification, utilisation et mise en oeuvre ;
- Les arbres : spécification, utilisation et mise en oeuvre.

2. Partie II

- Structure et fonctionnement d'un ordinateur (codage de l'information, machine simulord).

DEUG2 / API : Cours : 24 heures ; TD : 24 heures ; TP : 24 heures.

MATIÈRES
ENSEIGNÉES

DEUG Sciences 2ème niveau

Génie logiciel : buts, méthodes et outils

À l'issue des trois modules d'informatique obligatoires pour les mentions Mass et Mias, on propose aux étudiants qui ont envie d'approfondir leurs connaissances en informatique d'élargir leur vision de la discipline dans deux directions complémentaires : d'une part, les problèmes posés par la rationalisation du processus de production du logiciel et, d'autre part, certains outils mathématiques incontournables pour l'informatique. L'idée retenue pour ce module consiste à aborder, de manière fédératrice, le génie logiciel, la théorie des langages, et la logique. Les étudiants sont amenés à participer à la réalisation d'un logiciel de taille conséquente. Au cours de cette réalisation, ils ont l'opportunité d'appliquer des concepts de théorie des langages et de logique qui sont vus en cours et d'utiliser des outils de production de logiciel reposants sur de tels concepts.

PLAN

1. Introduction au génie logiciel
 - spécificités et problèmes ;
 - cycle de développement et cycle de vie.
2. Le logiciel Félin
 - cahier des charges, spécifications et découpage fonctionnel.
3. Langages et grammaires
 - alphabets, mots et langages ;
 - grammaires ; classification de Chomsky.
4. Langages de type 3 et automates à nombre fini d'états
 - langages de type 3, langages rationnels, automates à nombre fini d'états ;
 - programmation par automates ; utilisation d'un générateur d'analyseurs.
5. Systèmes formels
 - introduction, syntaxe et sémantique ; validité et complétude.
6. Calcul des propositions
 - logique mathématique ; langage du calcul des propositions ;
 - sémantique ; syntaxe (Robinson).

DEUG Sciences 2ème niveau

Programmation d'un ordinateur scientifique : approche fonctionnelle (2)

Ce module est la poursuite de l'enseignement d'informatique de première année commun aux mentions SM et STPI. Son objectif est une initiation à la programmation d'un ordinateur en vue de son utilisation dans des filières scientifiques et techniques. Cette initiation est faite suivant une approche fonctionnelle, à partir du logiciel Mathematica. Les TP visent à développer la maîtrise de la programmation pour modéliser des situations couramment rencontrées dans le domaine scientifique.

PLAN

1. Les listes, et leur utilisation pour la modélisation de séquences de valeurs. Fonctions d'accès, de construction, et de calcul sur les listes
2. Les ensembles et leur manipulation par des fonctions d'ordre supérieur
3. Les structures. Définition, fonctions d'accès. Utilisation pour la représentation d'objets hétérogènes
4. Approche du calcul itératif par la notion de point fixe

DEUG Sciences 2ème niveau

Programmation fonctionnelle et impérative

Ce cours fait suite à l'enseignement d'informatique de première année commun aux mentions SM et STPI. Il aborde des notions complémentaires de programmation fonctionnelle, à travers l'utilisation du logiciel Mathematica. Il aborde ensuite la programmation impérative, à travers le langage Pascal.

PLAN

1. Programmation fonctionnelle

- types complexes : listes, structures, ensembles ;
- motifs et filtrage.

2. Programmation impérative

- types, expressions, variables, affectation, lecture/écriture ;
- conditionnelles ;
- itérations ;
- procédures et portée des identificateurs ;
- fichiers ;
- tableaux ;
- type énuméré, type structure ;
- types abstraits (exemple : la pile) ;
- représentation des nombres, décidabilité, complexité.

DEUG Sciences 2ème niveau

Conception de systèmes réactifs : but, méthodes et outils

Ce projet vise à étudier les systèmes informatiques dits « réactifs », comme exemples de systèmes complexes. Pour ce faire, l'étudiant réalise un projet informatique de taille conséquente autour du pilotage automatique d'un ou plusieurs robots mobiles.

PLAN

1. Programmation orientée objet
 - Concepts de base ;
 - Héritage ;
 - Généricité.
2. Systèmes réactifs
 - Notion de comportement, d'agent réactifs ;
 - Le modèle d'automate : composition, raffinement ;
 - Le langage oRis ;
 - Les différents modes de simulation en oRis.
3. Cours liés au projet