

Initiation à la programmation

Module M1011 - DUT Informatique

Introduction

yann.secq@univ-lille1.fr

Abdelghani ATAMENIA, Géry CASIER, Iovka BONEVA, Antoine NONGAILLARD

Menu

- Informations générales
- Programmer est-ce philosopher ?
- Tout ce que vous avez toujours voulu savoir sur la programmation sans jamais avoir osé le demander
- Concrètement ?

Informations générales

- Vos enseignants
- Votre formation
- Programmation en DUT
- Organisation des UE MI I02/I I03

Vos enseignants

- M. Atamenia, Mme Boneva, M. Casiez, M. Nongaillard et M. Secq
- yann.secq@univ-lille1.fr
- Enseignant-Chercheur en Informatique à Lille I
- Enseignement: programmation impérative, OO, fonctionnelle, démarche agile...
- Recherche: conception de systèmes multi-agents

Votre formation

- DUT Informatique: 50% info / 50% autres
- Programmation/conception: 50% des UE info !
- Premier semestre crucial: bases et fondements
- **Soyez acteur de votre apprentissage**

Programmation en DUT

- Programmation impérative (S1)
- Programmation orientée objets (S2/S3)
- Conception orientée objets (S2/S3)
- Programmation web (S3/S4)
- Programmation mobile (S4)
- Programmation fonctionnelle (S4)

Organisation UE M I I 02/03

- Présence obligatoire: cours / TD / TP
- Contrôle continu (IE+CTP+DS+Projet)
- 5 mois pour assurer votre DUT
- **Sans pratique, point de salut !**
- Interagissez avec vos enseignants
- *Install party* Linux (important!)

Programmer = philosopher ?

- Philosophie: compréhension du monde et des hommes
- Monde des idées et des concepts
- Objects abstraits décrivant/généralisant des éléments matériels ou immatériels
- Quel rapport avec la programmation ?

programmer = décrire

- **Programmer:** décrire un traitement automatisable par un calculateur
- Description reposant sur un très petit nombre de mots
- **Informatique = Information + Automatique**
- Traitement automatisé de l'information par un calculateur
- Ordinateur: calculateur universel

Programmer = débattre

- Avant de décrire, il faut comprendre
- Importance des échanges et débats pour appréhender le problème
- Plusieurs descriptions peuvent convenir à une même situation
- Un programme s'adresse avant tout aux humains !

Programmer = organiser la complexité

- **Programmer est difficile**
 - Lutte perpétuelle contre la complexité
 - Structuration différente de la complexité: impérative, OO, fonctionnelle, logique, concurrente ...
- Mais les champs d'application sont infinis
 - Guerres, gestion, communications, divertissements ...
 - *Warcraft* = réseau + base de données + temps-réel + robustesse ...
- Programmer c'est simplifier (KISS!)

Programmer = philosophie appliquée

- Informatique = mathématiques appliquées + ingénierie
- Programmer = forme de philosophie appliquée (créer des « théories » exécutables !)
- Programmation = outil générique de création d'outils
- Humilité: si bug il y a, de votre responsabilité cela sera :)
- *Last but not least*: pensez à votre responsabilité (PRISM)

Woody Allen



“Tout ce que vous avez
toujours voulu savoir
sur ~~le sexe~~ *

*sans jamais oser le demander”



la programmation !

Tout ce que vous ...

- Objectifs des UE MI I 02/03
- Notions à maîtriser
 - Algorithme/Programme
 - Assertions
 - Types et variables
 - Alternatives et boucles
 - Fonctions

Objectifs du semestre

- Maîtriser les briques de bases de la programmation
- Déterminer les éléments pertinents d'un problème
- Savoir analyser et décomposer un problème
- Maîtriser un langage impératif (java simplifié)
- Réaliser un projet significatif (**projet de semestre**)
- **There no speed limit, but ...**

Notions à maîtriser

- Notions à maîtriser
 - Algorithme et programme
 - Assertion et test
 - Type et variable
 - Alternative et boucle
 - Fonction
 - Tableau
 - Décomposition

Algorithme et programme

- un algorithme est une description abstraite de la résolution d'un problème
- la description repose sur un petit nombre d'éléments de base pouvant être composés
- un programme est un algorithme exprimé dans un langage informatique
- un même algorithme peut être implémenté en différents programmes (C, Java, LISP ...)

Assertion

- Assertion = affirmation devant être vraie
- "Tous les hommes naissent libres et égaux"
- "3 additionné à 5 vaut 8"
- Assertion = expression devant être vraie
sinon ... BOUM!
- TP = ensemble d'assertions que vous devez rendre valides

Types et variables

- Informatique = Information + Automatique
- Qu'est-ce qu'une information ?
 - Information = élément représentable / manipulable
- Quel type d'information ?
 - Au début: nombre, texte, booléen

Alternative et boucle

- Structures de contrôle: alternative et boucle
- Alternative: exécute soit A, soit B
 - **SI** (condition) **ALORS** (...) **SINON** (...) / **if** () **else**
- Boucle: répéter plusieurs fois A
 - **POUR** i **ALLANT DE** ... **A** ... **FAIRE** / **for**
 - **TANT QUE** (condition) **FAIRE** / **while**

Fonction

- Fonction = traitement réutilisable
- Une fonction dispose d'information en entrée et produit une sortie
- Informations en entrée: paramètres
- Information en sortie: valeur de retour (résultat)
- Signature de fonction = valeur de retour + nom de la fonction + paramètres

Tableau

- Nécessaire pour stocker et accéder à plusieurs informations
- Représente un ensemble d'information de même type accessibles via un (ou plusieurs) indice(s)
- Tableau à une ou plusieurs dimensions
- Principale structure de donnée que nous utiliserons

Concrètement ?

- Après les concepts, la pratique !
- Même langage en cours / TD / TP
- *ijava* = *Java* mais sans les objets :)
- Outils: emacs, le JDK et ap.jar
- *Let's code !*
- PS: avoir votre propre ordinateur est important

Synthèse

- Vue d'ensemble du premier semestre
- Chacun peut avancer à son rythme ... MAIS
 - Respect des contraintes données dans les énoncés !
 - Projet: expression de votre créativité et terrain pour vous exercer
- *There is no speed limit* (mais pas d'objet !)

?

