IPO

Initiation à la
Programmation
Objet

IGI-1202

A3P

Apprentissage
Par
Problème
de la **P**rogrammation

IGI-3007

Responsable : Denis BUREAU

Précédente présentation

- Pédagogie ≠ => apprentissage ≠
- Inversion TP -> TD -> cours
- Modalités d'évaluation
- Filières, cursus informatique E1
- Implication personnelle
- **Poser des questions**
- Trouver soi-même grâce à une « question »
- Ne pas attendre ni regarder la solution

Rappel :

Nécessité d'une attitude pro-active
dans cette unité.
(signaler dès que qqch ne va pas,
dès qu'il manque une information,
dès que vous avez une question, …)

Se contenter d'assister «passivement»
à chaque séance de TP, TD, cours, et Résa
est une condition nécessaire
mais pas suffisante !

Les séquences

- Elles sont toutes numérotées et détaillées sur la page de l'unité.
- Typiquement : TP*, TD, Résa, Cours
  (* *marqué TDm sur l'emploi du temps*)
- **Chaque séance est « cliquable »** et mène à un article de forum auquel on peut répondre pour poser des questions.
- Lire les échanges fait partie de l'apprentissage.
  Ce n'est pas du spam !

Les TP (TDm)

- TP « *tout court* » (9, énoncé) :
  découverte de nouvelles notions java par la pratique ; chercher à comprendre,
  pas à aller le plus vite possible
  *(pas de rapport à rendre, **un par poste**)*
  mais ne pas perdre de temps :
  profiter au maximum des intervenants !
  Il peut être demandé une lecture préalable.

- ~~TP « de projet » (2, sans énoncé) :~~
  ~~point d'avancement du projet~~
  ~~avec une aide technique pour avancer.~~

Les TD

- TD (3) :

  - écrire un programme différent du TP, mais utilisant les mêmes notions => compréhension

- - écrire du java sur papier => mémorisation, intégration, et **préparation aux contrôles**

- - appeler l'intervenant dès qu'on ne comprend pas qqch ou lorsqu'on a terminé un exercice

Les cours

- Cours « spéciaux » (2 x 1h) :
  - introduction à la pédagogie et à l'unité
  (puis lecture : concepts « objet »)

- Cours « normaux » (7 x 1h) :
  cours courts !
  - restructuration de ce qui a été vu en TP/TD,
    voire en projet ==>
  Vous comprenez ce qui se dit ! Écouter 100% ?
  - basé sur un résumé de cours et des Q/R
  (**indispensable** à la véritable compréhension)

Les séances Résa (ex-Pers)

- Résa « dédiées » (3 x 1 ou 2h) :
  pour accomplir une tâche, comme rendre une version du projet ou remplir un formulaire
  (avec assistance par Teams)
- Résa « simples » (15 x 1 ou 2h) :
  salles réservées pour avancer le projet
  ou pour terminer un TP ou un TD
  (avec assistance par Teams)
- Tout à fait déraisonnable et non rentable
  de ne pas utiliser ces créneaux !

Travail personnel

- Travail non planifié dans ADE :
- Terminer les TP
- Terminer les TD (et les passer sur machine)
- Avancer le projet
- Relire les « apports de connaissance »
- Formuler et **poser des questions**

- ==> environnement de développement
  sur son ordinateur personnel
  (ou rester tard/week-end à l'ESIEE)

Le projet = un jeu d'aventure

- Le but n'est **pas** de réaliser un jeu vidéo !
- Le but est d'apprendre plus de java, et d'apprendre à utiliser une bonne conception objet, en réalisant un jeu d'aventure dont vous choisissez le scénario *(impossible à apprendre autrement, nécessité de refactoring)*.
- Un but secondaire est d'être fier d'avoir réalisé un jeu tout seul, si possible d'être sélectionné pour le Jour des Projets, et pourquoi pas, de gagner un prix ?

En savoir plus sur le projet 1/2

- Jeu d'aventure, tour par tour
- Projet décrit dans un livre (anglais/français)
- Liste officielle des exercices (oblig./optionnels)
- Créativité, contraintes, créativité
- Versions successives :
  - programmation améliorée
  - fonctionnalités ajoutées
- Explications données dans le livre, et exemples de programmation donnés au fur et à mesure, mais résolution de problèmes !
- **Commencez à réfléchir** à votre histoire …

En savoir plus sur le projet 2/2

- Pris par la main au début
  (ne pas traîner)
- De plus en plus en autonomie
  (ne pas se décourager)
- Découverte de notions nouvelles
- Comprendre le pourquoi du re-factoring
- **Ne pas attendre le prochain TP**
  pour poser des questions sur un exercice
  (utiliser le forum !)
- **Attention au plagiat !**
  (pour le moindre morceau de code)

Le langage Java (1/2)

- Plus facile à apprendre que le C.
  Erreurs moins graves pour un débutant.
- Après le C en E2, vous pourrez apprendre par vous-mêmes le C++ ou le C#.
- Famille de langages la plus utilisée au monde (C : 1972. C++ : 1983. Java : 1995, 21 versions)
- JRE = Java Runtime Environment (exécution)
- JDK = Java Development Kit (compilation)

- **Inutile :** installer le JDK **11** sur son ordinateur personnel (gratuit, open source)

Le langage Java (2/2)

- On ne peut pas apprendre « tout » Java.
- Ni le langage, encore moins l'A.P.I.
  (l'essentiel, javadoc, « pied à l'étrier »)

- Besoin de tout : TP + TD + cours + projet

- Conception Orientée Objet (philosophie)
  petit à petit, surtout par le projet

- Parti pris, vocabulaire, exigences pédagogiques

La Java Virtual Machine

- Processeur virtuel = programme interpréteur d'instructions « assembleur »
- Fourni sur toutes les plateformes (PC, smartphone, box, carte à puce, …)
- Programme compilé tourne tel quel entre Mac / PC / Windows / Linux / …

- Erreurs à la compilation ou à l'exécution
- Dire un maximum de choses au compilateur ! (ne pas économiser des caractères)

Environnement de développement

- BlueJ, prononcez « bloudjè » (IDE)
- Installer **v5.0.3** (pas plus) sur son ordi. Perso. (gratuit, open source, inclut JDK **11**)
- Plus adapté qu'Eclipse ou NetBeans ou IntelliJ (pédagogique)
- Peu de menus (tous les regarder ?)
- Schéma des classes / interaction
- Object bench / interaction / inspect
- Tests automatiques
- Debugger, Code pad

iCampus (Moodle)

- Tout est sur iCampus, ma page perso (/E1/)
  semaine après semaine,
  séance après séance.

- Vous **devez** cliquer sur chaque séance,
  y compris Résa/Pers, pour lire les consignes
  (évidemment avant/pendant, pas après !).

- Toute unité est sur Blackboard (→ Moodle)
- Donc s'inscrire à l'unité sur Moodle
- (puis aussi à l'unité du projet Zuul)

Les bons conseils

- Il n'est pas attendu de vous que vous sachiez résoudre tous les problèmes par vous-mêmes.
- Ne pas accepter de ne pas comprendre qqch (retrouvez cette révolte qui est en vous !)
- **Prévenir** quand vous ne pouvez pas assister à une séance
  (en cas d'imprévu, s'excuser après)
- Et surtout : **rattraper au plus vite** !
  (1h encadrée ==> 2h en autonomie)

L’approche pédagogique

- Elle est évidemment non négociable.
- Toute l’équipe pédagogique est persuadée qu’elle est plus efficace que la « classique ».

- Nous comprenons qu’elle puisse ne pas convenir à une minorité d’étudiants.
- Nous essaierons de les convaincre dans un premier temps, mais ensuite, nous les aiderons à compenser les inconvénients pour eux de la méthode.
- **Votre persévérance est indispensable !**

Les contraintes absolues

1. Lire ses **mails** TOUS LES JOURS ! **[A3P:]**
2. **Toutes les séances sont utiles** : difficile de réussir en manquant un TP, TD, Cours, Résa.
3. Lire tous les mots de toutes les phrases des énoncés, des documents, ET DES MAILS.
   Unité Moodle : IGI-3007=A3P 2022
4. Bien gérer son temps :
   - travailler un peu **(quasiment) tous les jours**
   - ne pas rester bloqué plus d'une demi-heure sans poser de question (10mn en TP)

   (il est anormal d'y passer 20h/semaine !)

Bien commencer

1. L'unité démarre beaucoup plus intensément qu'il n'y paraît : beaucoup de notions nouvelles (même simples) et de **vocabulaire** (indispensable pour comprendre la suite) ; à apprendre/travailler **dès la 1ère semaine**.
2. *Chaque séance utilise ce qui a été vu précédemment. N'accumulez pas de retard !*
3. *Vous pouvez toujours essayer de rattraper un retard de cours peu avant un partiel, mais vous n'aurez pas acquis le savoir-faire nécessaire pour la partie programmation …*

Votre réussite est entre vos mains

Votre « pouvoir personnel » est immense,
2 attitudes possibles :

1. Utiliser toute son énergie et son temps à être capable de démontrer pourquoi on a échoué (garder les problèmes en réserve ➔ + de repos)

2. Utiliser toute son énergie et son temps pour essayer de réussir (résoudre les problèmes **dès qu'ils se présentent** ➔ + de travail)

3. Valider en ligne la charte de l'unité ==> **Résa0.2**

Dès que quelque chose ne va pas :

Dites-le moi au plus vite !

- Vous ne comprenez pas ce qui est demandé
- Vous ne comprenez pas comment faire
- Vous n’arrivez pas à résoudre un problème
- Un lien ne fonctionne pas
- Un document/renseignement manque
- Un intervenant pose problème (son nom !)
- Vous êtes perdu(e)…

Prochaine séance :
Résa 0.2

- Se connecter, puis lancer un navigateur
- **Page E1 ~~ou E3ST~~** dans la barre d'adresse
- Se connecter.
- E1 / S.I. / Informatique / IGI-1104=**IPO 2021**
- **M'inscrire**
- Cliquer sur Résa 0.2 => instructions
- Charte de l'unité, puis parcourir ressources.
- (une 2ème et dernière fois **M'inscrire**)

Merci de votre attention

Des questions ?