

# IFT1025 automne 2009

## Avant l'intra

Miklós Csűrös

6 octobre 2009

L'examen intra aura lieu mardi le **13 octobre, 15<sup>30</sup>–17<sup>30</sup> au local 1355** du Pavillon André-Aisenstadt.

**Aucune documentation ne sera permise.**

L'examen couvre les sujets ci-dessous.

### Préalables

Tout le matériel de IFT1015 ; *Big Java* chapitres 1–6, sections 8.3–8.8.

### Packages

**À lire :** *Big Java* section 8.9.

- \* notion de *package* (module)
- \* `import`, noms qualifiés

### Tableaux et la classe `ArrayList`

**À lire :** *Big Java* sections 7.1–7.7.

- \* type *tableau* — c'est un objet, passé comme référence
- \* parcours d'un tableau pour recherche d'un élément ou le max/min, décalage pour insertion ou déletion, extension dynamique
- \* la classe `java.util.ArrayList`

### Interfaces

**À lire :** *Big Java* sections 9.1–9.4.

- \* notion de l'interface en Java ; déclarations (méthodes publiques et constantes) ; implémentations
- \* hiérarchie d'interfaces
- \* polymorphisme, mot-clé `instanceof`

## Héritage

À lire : *Big Java* sections 10.1–10.7

- \* notion de héritage, sous-classes et super-classes ; hiérarchies de super-classes
- \* règles de héritage de méthodes, variables et constructeurs : *overriding* (redéfinition) et *shadowing*, *late binding* et *early binding*
- \* mot-clé `super`
- \* droits d'accès : `public`, `private`, `protected`, niveau module (*package*)
- \* classe abstraite
- \* redéfinition de méthodes de la classe `Object` : `toString`, `equals`

## Classes imbriquées

À lire : *Big Java* sections 9.5–9.9.

- \* classe statique imbriquée
- \* classe interne, classe locale, classe anonyme
- \* `this` qualifié, mot-clé `final`

## Entrée-sortie et exceptions

À lire : *Big Java* sections 11.1–11.7.

- \* modèle d'entrée-sortie en Java : flux de données (*data stream*)
- \* flux de caractères et d'octets
- \* ouvrir, lire, écrire, fermer des flux
- \* classes principales de types `Reader`, `Writer`, `InputStream`, `OutputStream`, combinaison de flux, classe `File`
- \* notions d'erreurs et d'exceptions, `RuntimeException`
- \* déclaration d'exceptions : `throws`
- \* lancer une exception, définition des exceptions «personnalisées»
- \* capturer une exception `try...catch...finally`

## Java Swing GUI

À lire : *Big Java* sections 2.11G–2.13G, 9.6G–9.10G

- \* éléments d'interface usager : `JFrame`, `JPanel`, `JButton`, `JFileChooser`
- \* dessin en Java : classe `Graphics`, méthode `paintComponent`
- \* événements, *listeners* (écouteurs) : `ActionListener`, `MouseListener`, `Timer`

## Conception de logiciels OO

À lire : *Big Java* sections 8.1–8.2, 12.1–12.5.

- \* notions de *coupling* and *cohesion*
- \* phases dans le cycle de vie du logiciel
- \* relations entre classes, diagramme UML
- \* documentation structurée, `javadoc`