

Mid-term Review

1 Mar 2010

CMPT166

Dr. Sean Ho

Trinity Western University

Review on one slide

- Java basics (ch1)
 - Console I/O (2), if/while/for (3)
 - Arrays (6), Exceptions (9)
- OO design: Classes, pub/priv/prot (4)
 - static, Alias/ref, Packages (5)
 - Inheritance (7), Polymorphism, abstract (8)
 - Interfaces, Inner classes (13)
- File I/O (10): text, byte, object (Serializable)
- Swing (17-18): events, listeners, layout, widgets

Java basics (ch1-3, 6, 9)

- Java system: JDK/JRE, bytecode, VM
 - 8 primitive types (and OO wrapper classes)
 - Style: naming, Javadoc comments
- I/O: print/ln/f, Formatter, Scanner, next*()
- if/else, switch, while, for(;;), break/continue
- Arrays: declare → allocate → instantiate elts
 - Python-style for loops: for (elt : array)
- Exceptions: try/throw/catch
 - Catch-or-declare rule
 - Subclassing Exception, auxiliary data

OO concepts (ch4-5, 7-8, 13)

- Creating classes: attributes/methods
 - `private` → (package) → `protected` → `public`
 - **Constructors**, calling `self()`, copy constructor
 - `static`: class methods/attribs; static import
- **Inheritance**: overriding, polymorphism
 - OO **design**: UML class diagram, “has a” vs. “is a” vs. “knows how to”
 - **Abstract**, (mthd, cls) `final` (attrib, mthd, cls)
 - **Interfaces**
 - **Inner** classes (e.g., use for event listeners)

File I/O (ch10)

- File object
- Text-based I/O:
 - `PrintStream`, `print/ln/f()`
 - `Scanner`, `next/Int/Double/Line()`
- Byte-based I/O: `File{Output,Input}Stream`
- Object-based I/O:
 - `ObjectOutputStream`, `.writeObject(obj)`
 - `ObjectInputStream`, `.readObject()` (`Object`)
- Exceptions (`IOException`): `FileNotFoundException`, `EOF`

Swing (ch17-18)

- Event programming model:
 - `main()` → JFrame subclass constructor
→ create/layout widgets
 - Events → event listeners
- Widgets: JPanel, JButton, JLabel, JTextBox, etc.
 - Action cmd: `.setText`, `.getActionCommand()`
 - Menus: MenuBar, Menu, MenuItem
- Layout managers:
 - Flow, Border, Grid, GridBag, Group
- Drawing: `paint/Component()`, shapes, colours