# Linguaggi di Programmazione

a.a. 14/15

docente: Gabriele Fici

gabriele.fici@unipa.it

# Informazioni sul corso:

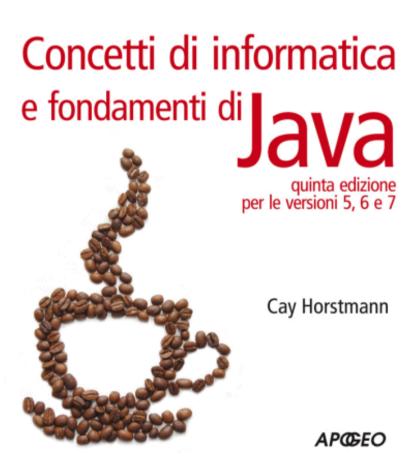
- 9 CFU (72 ore di lezioni frontali)
- 7 ore settimanali:
  Mer 8:30-10:30, Gio 10:30-13:30, Ven 8:30-10:30
- Ricevimento: Lun. 15:00-16:00 e Mer: 15:00-16:00 (mandare email per prendere appuntamento)

#### Obiettivi del corso:

- Studieremo il paradigma OOP (Object Oriented Programming)
- Impareremo il linguaggio di programmazione Java (Standard Edition 7)

Libro consigliato:





5 copie in Biblioteca

# Prerequisiti:

- Basi della programmazione
- Conoscenza di base ambiente Linux e/o Windows (in particolare comandi da console per muoversi tra le directory)
- Dimestichezza con un editor di testo, a vostra scelta
- Buona conoscenza dell'Inglese, non necessaria ma importante

# Java è:

- un linguaggio di programmazione OO
- un ambiente di esecuzione (tramite Java Virtual Machine)
- un insieme di librerie (molte librerie di sistema)

# Dove viene utilizzato Java?

- Nei personal computer
- Nei server sul web
- Nei sistemi "embedded" (televisori, cellulari, etc.)

• ...

# Alcuni dettagli:

- In realtà, più che un linguaggio di programmazione,
  Java è una tecnologia
- Useremo la versione Java Standard Edition (SE) 7 versione SE 8 già rilasciata, ininfluente per questo corso
- Java è stato sviluppato negli anni '90 da Sun, che ora è della Oracle
- Sito web: <a href="http://www.java.com/">http://www.java.com/</a>

- Sintassi ereditata da C, quindi familiare
- Approccio globalmente coerente, molta ripetitività
- Ma è veramente così "semplice"? Maggiori ostacoli all'inizio, per acquisire l'approccio OO

- Il paradigma orientato agli oggetti è un approccio basato sulle *classi*, che contengono l'informazione nei *dati*, funzionano attraverso *metodi*, e servono a creare *oggetti*
- Il paradigma orientato agli oggetti permette di riutilizzare facilmente codice scritto da altri programmatori

- Un <u>sistema distribuito</u> è un sistema che coinvolge contemporaneamente più computer, come ad esempio il web
- In effetti Java è stato originariamente concepito soprattutto per il web

- Java è un linguaggio <u>di alto livello</u>, deve cioè essere <u>interpretato</u> (ovvero trasformato in un linguaggio comprensibile dal computer)
- Vedremo che Java ha un approccio particolare, che fa uso di una macchina virtuale (Java Virtual Machine) per eseguire il codice

"Java is a simple, object-oriented, distributed, interpreted, robust, secure, architecture-neutral, portable, high-performance, multi-threaded, and dynamic language"

• I programmi scritti in Java sono <u>robusti</u>, nel senso che qualora contengano errori, essi non creano catastrofi che si propagano a tutto l'ambiente che li contiene (sistema operativo e altri programmi)

"Java is a simple, object-oriented, distributed, interpreted, robust, secure, architecture-neutral, portable, high-performance, multi-threaded, and dynamic language"

• L'approccio usato da Java permette l'<u>incapsulamento</u> dei dati, che non possono essere acceduti da chi non è stato autorizzato dal programmatore

"Java is a simple, object-oriented, distributed, interpreted, robust, secure, architecture-neutral, portable, high-performance, multi-threaded, and dynamic language"

 Java non ha prestazioni significativamente differenti se eseguito su architetture diverse

- I programmi scritti in Java possono essere eseguiti su computer con s.o. diversi, senza problemi di <u>porting</u> (adattamento a un'altra piattaforma)
- Il credo di Java è: "Write once, run everywhere" (ovvero scrivere codice che possa essere eseguito su qualsiasi piattaforma)

- Le performance di un linguaggio di programmazione non riguardano solo la sua velocità di esecuzione
- Java è abbastanza veloce in tutto, ma non eccelle per velocità in nessun ambito

"Java is a simple, object-oriented, distributed, interpreted, robust, secure, architecture-neutral, portable, high-performance, multi-threaded, and dynamic language"

• I programmi Java possono eseguire diverse parti di codice contemporaneamente, sfruttando il *multi-threading* dell'architettura (cioè la capacità di eseguire diversi compiti contmporaneamente)

"Java is a simple, object-oriented, distributed, interpreted, robust, secure, architecture-neutral, portable, high-performance, multi-threaded, and dynamic language"

• Java è un linguaggio dinamico. Molte azioni avvengono quando è effettivamente necessario che avvengano, e non all'inizio dell'esecuzione.