

Linguaggi dinamici

Giacomo Cabri
giacomo.cabri@unimore.it

Mauro Andreolini
mauro.andreolini@unimore.it

Home page del corso
<https://www.agentgroup.unimore.it/didattica/ld/>

LINGUAGGI DINAMICI

PARTE 1

INTRODUZIONE

Parte 1

Informazioni sul corso

Contenuti del corso

- **Fornire una presentazione dei principali linguaggi *dinamici* (Perl, Python), evidenziando le differenze con i linguaggi *statici* (C, C++, Java)**
 - **Architettura dell'ambiente di esecuzione**
 - **Sintassi e semantica, procedurale e ad oggetti**
 - **Caratteristiche innovative rispetto ai linguaggi statici: gestione run-time di memoria, codice, errori**
 - **Strumenti di supporto per la generazione e la manutenzione agile di software**

Obiettivi formativi del corso

- **Al termine del corso, lo studente acquisirà (auspicabilmente) le seguenti competenze:**
 - **saper sfruttare le caratteristiche innovative di tali linguaggi per la prototipazione rapida e la manutenzione di software**
 - **saper assimilare le caratteristiche salienti di *nuovi* linguaggi dinamici in tempi brevi**
 - **arricchire il proprio bagaglio culturale di base con tutta una serie di funzionalità utili in diversi contesti (scripting avanzato, client/server, sistemi Web-based, calcolo scientifico)**
 - **saper scegliere il linguaggio adatto allo scopo**

Modalità di svolgimento del corso

- **Prima parte: introduzione ai linguaggi dinamici**
 - **Architettura hw/sw di un linguaggio dinamico**
 - **Tipizzazione dei dati**
 - **Gestione run-time della memoria, del codice, degli errori**
 - **Strumenti di supporto per:**
 - **modularizzazione del software**
 - **ricerca di errori (debugging)**
 - **misurazione delle prestazioni (profiling)**
 - **verifica funzionale (unit testing)**

Modalità di svolgimento del corso

- **Seconda parte: Teoria e sperimentazioni**
- **Linguaggio Python**
 - **Costrutti di base**
 - **Studio delle caratteristiche dinamiche**
 - **Gestione file e directory**
 - **Organizzazione "ad oggetti"**
 - **Creazione di strutture dati complesse**
 - **Documentazione**
 - **Debugging**

Modalità di svolgimento del corso

- **Terza parte: Applicazioni avanzate**
- **Linguaggio Python**
 - **Metaprogramming**
 - **Text processing ed espressioni regolari**
 - **Interfacce grafiche**
 - **Applicazioni scientifiche**
 - **Programmazione remota e distribuita**
 - **Unit testing**

Modalità di svolgimento del corso

- **La prima parte del corso non prevede esercitazioni in laboratorio**
 - **Full-immersion nel mondo dei linguaggi dinamici**
- **La seconda e la terza parte del corso faranno uso massiccio e costante di esercitazioni in laboratorio**
- **In ciascuna esercitazione sono proposti esercizi che lo studente deve (provare a) risolvere**

Prerequisiti e materiale didattico

- **Prerequisiti obbligatori (in assenza dei quali l'esame non può essere sostenuto):**
 - Programmazione I
 - Programmazione II
 - Programmazione a oggetti
- **Prerequisito fortemente consigliato:**
 - Sistemi operativi

- **Materiale didattico**
 - Sul sito del corso

Software

- **Interprete Python**
- **Disponibili nella maggior parte delle distribuzioni Linux**
 - **Se non ci sono, installare i pacchetti**
- **Da installare in Windows**
 - **www.python.org**
- **Plugin per Eclipse**
 - **PyDev**
- **Chi vuole, può anche provare altri linguaggi dinamici**
 - **Perl, Ruby, ...**

Modalità di esame

- **Progetto:**
 - **Progetto ed implementazione di un software scritto in uno dei linguaggi visti nel corso**
 - Produzione di documentazione in stile “javadoc”**
 - **Produzione di unit test estesi che attestano la funzionalità completa del software**
 - **Consegna di una tesina (max. 10 pagine)**
 - **Capitolo 1: Descrizione del progetto**
 - **Capitolo 2: Modularizzazione**
 - **Capitolo 3: Scelte implementative**

Modalità di esame

- **Colloquio orale:**
 - **Tre domande**
 - **La prima domanda riguarda la discussione approfondita del progetto**
 - **La seconda e la terza domanda riguardano gli aspetti teorici e/o pratici del corso**
- **Voto finale:**
 - **È la media aritmetica del voto del progetto e del voto del colloquio orale**

Ricevimento studenti

- **Per dubbi semplici e puntuali**
 - spesso basta una email per chiarire i dubbi
- **Per dubbi più ampi**
 - orario di ricevimento (contattare il docente via mail)
- **Cercare di risolvere i problemi da soli**
- **Specificare bene il problema (“non funziona” non è utile...)**
- **Il docente non è un help desk gratuito**
- **Il ricevimento non è sostitutivo allo studio e alla ricerca personale di informazioni**