



---

ANNO ACCADEMICO: 2017/2018

---

INSEGNAMENTO: Tecniche Avanzate di Programmazione

---

TIPOLOGIA DI ATTIVITÀ FORMATIVA: Base

---

DOCENTE: GIANVITO SUMMA

---

e-mail: [gianvito.summa@gmail.com](mailto:gianvito.summa@gmail.com)

sito web: <http://informatica.unibas.it/moodle/>

telefono:

cell. di servizio (facoltativo):

---

Lingua di insegnamento: ITALIANO

---

n. CFU: 6

n. ore: 48

Sede: Potenza

Semestre: Annuale

Scuola: Ingegneria

CdS: LM Ingegneria Informatica

---

#### OBIETTIVI FORMATIVI E RISULTATI DI APPRENDIMENTO

Il corso riguarda le tecniche di programmazione avanzata in linguaggio ad oggetti (Java). Lo scopo del corso è fornire allo studente competenze per la progettazione e costruzione di sistemi applicativi di complessità medio-alta.

Alla fine del corso, il discente dovrà dimostrare di aver acquisito le seguenti capacità:

- Analisi e progettazione di sistemi applicativi complessi;
- Applicazione delle tecniche avanzate di programmazione.

Le tematiche principali che il discente dovrà acquisire e padroneggiare includono:

- Design Pattern;
- Inversione del Controllo;
- Thread e programmazione concorrente;
- Clonazione e serializzazione;
- Tecniche di refactoring;
- Metodologie di sviluppo agili e AUP.

Con specifico riferimento agli obiettivi indicati dai descrittori di Dublino relativamente alla:

- **Conoscenza e capacità di comprensione:**  
*lo studente deve dimostrare di analizzare un problema di complessità medio-alta scomponendolo in sotto problemi di dimensione ridotta al fine di individuare e riconoscere l'applicabilità di pattern progettuali e tecniche di programmazione;*
  - **Capacità di applicare conoscenza e comprensione:**  
*Lo studente deve dimostrare di essere in grado di progettare e realizzare un'applicazione che risolva un problema di complessità medio-alta;*
  - **Autonomia di giudizio:**  
*Lo studente deve essere in grado di saper valutare in maniera autonoma e critica i possibili scenari implementativi relativamente ad un problema di complessità medio-alta, riuscendo a determinare costi e benefici per ciascuno di essi ed adottando quello in grado di conferire al sistema le caratteristiche di efficienza, efficacia, facilità di manutenzione e scalabilità;*
  - **Abilità comunicative:**  
*Lo studente deve avere la capacità di spiegare come intende realizzare il sistema facendo uso delle capacità di presentare un progetto utilizzando correttamente il linguaggio scientifico-tecnico;*
  - **Capacità di apprendimento:**  
*Lo studente deve essere in grado di aggiornarsi di continuo attraverso la consultazione di testi, riviste, siti web e pubblicazioni di livello avanzato nell'ambito della programmazione ad oggetti, allo scopo di acquisire capacità e skill da impiegare nella definizione di soluzioni sempre più efficaci ed efficienti.*
- 

#### PREREQUISITI

È necessario avere acquisito ed assimilato le seguenti conoscenze fornite dal corso di "Programmazione a Oggetti II":

- Ereditarietà e polimorfismo.
- 

#### CONTENUTI DEL CORSO

Sono di seguito elencati gli argomenti del corso con le relative unità didattiche.

---



---

Design Pattern:

- Introduzione; Storia; Pattern e Framework; Pattern nelle API di Java; Dati e operazioni: Conclusioni.

Inversione del Controllo:

- Introduzione; Dependency Injectio; Aspetti metodologici; AOP; Conclusioni.

Refactoring:

- Introduzione; Tecniche; Esempio.

Processo di sviluppo:

- Introduzione; RUP; XP; AUP.

Clonazione e serializzazione.

Threading e sincronizzazione.

---

METODI DIDATTICI

Il corso prevede 48 ore di didattica frontale in aula.

---

MODALITÀ DI VERIFICA DELL'APPRENDIMENTO

L'obiettivo della prova d'esame consiste nel verificare il livello di raggiungimento degli obiettivi formativi precedentemente indicati.

L'esame è diviso in 3 parti che hanno luogo in diversi giorni, nell'ordine:

- una **prova scritta** sugli argomenti trattati nell'ambito del corso (progettazione e tecniche). La prova ha lo scopo di valutare lo studio della materia e la comprensione degli argomenti di base e ha carattere di selezione (lo studente che non mostri una sufficiente conoscenza degli argomenti non è ammesso alle prove successive); per superare la prova è necessario acquisire almeno 18 punti su 30. Non è consentito consultare testi o utilizzare PC, smartphone, calcolatrici ed altri dispositivi tecnologici;
- una **prova pratica** al calcolatore, con l'obiettivo di valutare se lo studente ha la capacità di analisi e progettazione necessarie allo sviluppo di un sistema a complessità medio-alta. Per superare la prova è necessario acquisire almeno 18 punti su 30; Il tempo previsto per la prova è di 5 ore; è possibile consultare la documentazione di Java ed il materiale del corso (tutto il materiale è a disposizione sui PC del laboratorio). La prova pratica è strutturata in due esercizi: il primo che richiede capacità di analisi e progettazione ed il secondo che richiede capacità applicative;
- una **prova orale** nella quale sarà valutata la capacità di collegare e confrontare aspetti diversi trattati durante il corso; per superare la prova è necessario acquisire almeno 18 punti su 30.

Lo studente che sostiene e supera con almeno la sufficienza (18 punti su 30) le due prove intermedie previste ha diritto ad un bonus di 2 punti sulla prova pratica ed è esonerato dalla prova scritta purché sfrutti il bonus sostenendo la prova pratica e la prova orale entro il III Appello (Settembre). Le prove intermedie consistono in questionari a risposta multipla ed in esercizi di analisi, progettazione e sviluppo.

---

TESTI DI RIFERIMENTO E DI APPROFONDIMENTO, MATERIALE DIDATTICO ON-LINE

Appunti forniti dal docente disponibili sul sito del Corso di Laurea, <http://informatica.unibas.it/moodle/> (al link "Tecniche Avanzate di Programmazione").

Testi di riferimento:

- Cooper -- The Design Patterns Java Companion;
- Eckel -- Thinking in Patterns;
- Martin Fowler -- Refactoring -- Improving the Design of Existing Code -- 1999 – Addison-Wesley

Per ulteriori dettagli consultare la sezione dedicata ai "Riferimenti Utili" riportata per i vari Argomenti del corso, sul portale della formazione Moodle sopra indicato.

---

METODI E MODALITÀ DI GESTIONE DEI RAPPORTI CON GLI STUDENTI

All'inizio del corso, dopo aver descritto obiettivi, programma e metodi di verifica, il docente mette a disposizione degli studenti il materiale didattico mediante il portale della formazione Moodle del Corso di Laurea (cfr. Sito Web), previa iscrizione al medesimo corso.

Orario di ricevimento: lunedì e mercoledì dalle 18 alle 19.30 presso l'aula studio "Docenti a contratto" del DiMIE (III

---



---

---

piano). Oltre all'orario di ricevimento settimanale, il docente è disponibile in ogni momento per un contatto con gli studenti, attraverso la propria e-mail.

---

---

**DATE DI ESAME PREVISTE<sup>1</sup>**

I prova intermedia: 21 febbraio 2018

II prova intermedia: 27 giugno 2018

**Appelli:**

I: 4 e 6 luglio 2018

II: 18 e 20 luglio 2018

III: 12 e 14 settembre 2018

IV: 12 e 14 dicembre 2018

V: febbraio 2019

VI: maggio 2019

---

---

SEMINARI DI ESPERTI ESTERNI    SI     NO

---

---

ALTRE INFORMAZIONI

---

---

<sup>1</sup> Potrebbero subire variazioni: consultare la pagina web del docente o del Dipartimento/Scuola per eventuali aggiornamenti