

---

# Testi del Syllabus

---

Docente

Matricola:

---

Anno offerta: **2013/2014**

Insegnamento: **05924 - SISTEMI DI ELABORAZIONE**

Corso di studio: **5015 - INGEGNERIA INFORMATICA**

Anno regolamento: **2013**

CFU: **6**

Settore: **ING-INF/05**

Tipo attività: **B - Caratterizzante**

Partizione studenti: **-**

Anno corso: **1**

Periodo: **I° semestre**

Sede: **SEDE DIDATTICA DI PARMA**

---

<b>Tipo testo</b>	<b>Testo</b>
<b>Lingua insegnamento</b>	Italiano
<b>Contenuti</b>	<ol style="list-style-type: none"><li>1. Sistemi e modelli</li><li>2. Elaborazione automatica dell'informazione</li><li>3. Calcolo parallelo</li><li>4. Calcolo distribuito</li><li>5. Sistemi peer-to-peer</li><li>6. Sviluppo di applicazioni peer-to-peer con Sip2Peer</li><li>7. Calcolo autonomico</li><li>8. Tecniche di simulazione</li><li>9. Simulazioni ad eventi discreti con DEUS</li></ol>
<b>Testi di riferimento</b>	<p>M. Amoretti, dispense in inglese sugli argomenti del corso. B.P. Zeigler, H. Praehofer, T.G. Kim, "Theory of Modeling and Simulation", Second Edition, Academic Press, 2000. G. Bolch et al., "Queueing Networks and Markov Chains", Second Edition, Wiley Interscience, 2006. C. Ghezzi, D. Mandrioli, "Informatica Teorica", Città Studi, 1989. D. E. Culler, J. Pal Singh, "Parallel Computer Architecture: A Hardware/Software Approach", Morgan Kaufman, 1999</p>
<b>Obiettivi formativi</b>	<p>La finalità del corso è quella di definire e caratterizzare i sistemi di elaborazione dell'informazione. In particolare verrà fornita allo studente la capacità di comprendere:</p> <ul style="list-style-type: none"><li>- modelli teorici (a tempo discreto, a tempo continuo e orientate agli eventi)</li><li>- modelli di programmazione parallela e distribuita</li><li>- sistemi di tipo client/server e sistemi di tipo peer-to-peer</li><li>- sistemi autonomici</li><li>- tecniche di simulazione</li></ul> <p>Lo studente imparerà ad applicare tali competenze per:</p> <ul style="list-style-type: none"><li>- sviluppare modelli di sistemi di elaborazione, in particolare per studiarne le prestazioni</li><li>- programmare sistemi paralleli e distribuiti</li><li>- progettare sistemi distribuiti di tipo client/server e di tipo peer-to-peer</li><li>- progettare sistemi di elaborazione dotati di caratteristiche autonome</li><li>- sviluppare simulazioni a eventi discreti</li></ul>
<b>Metodi didattici</b>	<p>Il corso viene svolto con lezioni frontali in cui il docente spiega gli argomenti mostrando delle diapositive oppure scrivendo alla lavagna.</p> <p>La parte del corso riguardante la modellizzazione di sistemi tramite Catene di Markov comprende anche la soluzione alla lavagna di alcuni esercizi.</p> <p>La parte del corso dedicata ai sistemi peer-to-peer si conclude con un paio di esercitazioni pratiche in cui gli studenti sono invitati a utilizzare il middleware Sip2Peer per sviluppare semplici applicazioni.</p> <p>Le lezioni sulle tecniche di simulazione si concludono con un paio di esercitazioni pratiche in cui gli studenti sono invitati a utilizzare il tool di simulazione DEUS.</p>
<b>Altre informazioni</b>	Il materiale didattico e di supporto alle lezioni è disponibile sul sito <a href="http://lea.unipr.it">lea.unipr.it</a>

## Tipo testo

### Modalità di verifica dell'apprendimento

## Testo

Non sono previste prove in itinere.

E' prevista una prova scritta (3 ore) articolata su 6-7 quesiti a risposta libera,

relativi alla parte teorica del corso.

E' previsto lo svolgimento di una tesina legata agli argomenti del corso.

Tale tesina comporta l'utilizzo o lo sviluppo di software, la stesura di una relazione e una presentazione alla commissione (con il supporto di diapositive).

Ciascuna tesina può essere svolta da un singolo studente o da un gruppo di due studenti.

### Programma esteso

Contenuti Sistemi di Elaborazione (42 ore)

#### 1. Sistemi e modelli

- 1.1. Spazio degli stati
- 1.2. Modelli deterministici e modelli stocastici
- 1.3. Sistemi complessi
- 1.4. Sistemi complessi adattativi
- 1.5. Modelli Markoviani
  - 1.5.1. Catene di Markov a tempo discreto
  - 1.5.2. Catene di Markov a tempo continuo
  - 1.5.3. Modelli di Markov con premio
- 1.6. Dinamiche di popolazione
- 1.7. Topologie di rete
- 1.8. Modelli DEVS

#### 2. Elaborazione automatica dell'informazione

- 2.1. Cenni di teoria dell'informazione
- 2.2. Tassonomia dei sistemi di elaborazione
- 2.3. Automi
- 2.4. Architettura di Von Neumann

#### 3. Calcolo parallelo

- 3.1. Concetti generali
- 3.2. Modelli di programmazione parallela
- 3.3. Architettura NUMA; Onyx2
- 3.4. Parallelismo massivo, CM2
- 3.5. Message Passing Interface (MPI)
- 3.6. Sistemi multicore, Cell, General Purpose GPU Programming
- 3.7. CUDA

#### 4. Calcolo distribuito

- 4.1. Cluster computing
- 4.2. Grid computing
- 4.3. Cloud computing
- 4.4. Pervasive computing

#### 5. Sistemi peer-to-peer

- 5.1. Variabili di stato
- 5.2. Dinamiche delle reti peer-to-peer
- 5.3. Problematiche di progettazione
- 5.4. Strategie di progettazione degli schemi di overlay
- 5.5. Schemi di overlay pi noti (Napster, BitTorrent, eMule, JXTA, Skype, Chord, Kademia)

#### 6. Sviluppo di applicazioni peer-to-peer con Sip2Peer

#### 7. Calcolo autonomico

- 7.1. I quattro principi dell'autonomic computing
- 7.2. MAPE-K
- 7.3. NAM e NAM4J
- 7.4. Distributed Remodeling Framework

#### 8. Tecniche di simulazione

- 8.1. Concetti generali sulle simulazioni

## ***Tipo testo***

## ***Testo***

8.2. Simulazioni ad eventi discreti

9. Simulazioni ad eventi discreti con DEUS

---



# Testi in inglese

<b>Tipo testo</b>	<b>Testo</b>
<b>Lingua insegnamento</b>	Italian
<b>Contenuti</b>	<ol style="list-style-type: none"><li>1. Systems and models</li><li>2. Automatic data processing</li><li>3. parallel computing</li><li>4. distributed computing</li><li>5. Peer-to-peer</li><li>6. Development of peer-to-peer with Sip2Peer</li><li>7. autonomic computing</li><li>8. Simulation Techniques</li><li>9. Discrete event simulations with DEUS</li></ol>
<b>Testi di riferimento</b>	<p>M. Amoretti, lecture notes in English on the topics of the course. B.P. Zeigler, H. Praehofer, T.G. Kim, "Theory of Modeling and Simulation", Second Edition, Academic Press, 2000. G. Bolch et al., "Queueing Networks and Markov Chains", Second Edition, Wiley Interscience, 2006. C. Ghezzi, D. Mandrioli, "Informatica Teorica", Città Studi, 1989. D. E. Culler, J. Pal Singh, "Parallel Computer Architecture: A Hardware/Software Approach", Morgan Kaufman, 1999.</p>
<b>Obiettivi formativi</b>	<p>The purpose of the course is to define and characterize the information processing systems. In particular, the student will be given the ability to understand:</p> <ul style="list-style-type: none"><li>- Theoretical models (discrete-time, continuous-time and event-driven)</li><li>- Models of parallel and distributed programming</li><li>- Systems for client / server and peer-to-peer</li><li>- Autonomic systems</li><li>- Simulation techniques</li></ul> <p>The student will learn to apply such a knowledge to:</p> <ul style="list-style-type: none"><li>- develop computing systems models, in particular for performance analysis</li><li>- program parallel and distributed systems</li><li>- design client/server and peer-to-peer distributed systems</li><li>- design autonomic computing systems</li><li>- develop discrete event simulations</li></ul>
<b>Metodi didattici</b>	<p>Lectures are given by the teacher, which illustrates the topics with the support of slides, or by writing on the blackboard. Lectures on Markov Chains include the solution of some exercises. Lectures on peer-to-peer systems are completed by a tutorial on developing applications with the Sip2Peer middleware. Lectures about simulation techniques are completed by a tutorial on the DEUS simulation environment.</p>
<b>Altre informazioni</b>	Lecture notes, slides and exercises are available on <a href="http://lea.unipr.it">lea.unipr.it</a>
<b>Modalità di verifica dell'apprendimento</b>	<p>There will not be exams during the course.</p> <p>There will be a written exam (3 hours), with 6-7 open questions related to theoretical part of the course.</p> <p>The students will have to work (alone or in couples) on a small project, requiring software use/development, the writing of a report and a final presentation.</p>

## **Tipo testo**

### **Programma esteso**

## **Testo**

Lectures on Parallel and Distributed Computing (42 hours)

1. Systems and models
  - 1.1. State space
  - 1.2. Deterministic and stochastic models
  - 1.3. Complex systems
  - 1.4. Complex adaptive systems
  - 1.5. Markov models
    - 1.5.1. Discrete time Markov chains
    - 1.5.2. Continuous time Markov chains
    - 1.5.3. Markov reward models
  - 1.6. Population dynamics
  - 1.7. Network topologies
  - 1.8. DEVS models
2. Computing systems
  - 2.1. Information theory
  - 2.2. Taxonomy of computing systems
  - 2.3. Automata
  - 2.4. Von Neumann architecture
3. Parallel computing
  - 3.1. General concepts
  - 3.2. Parallel programming models
  - 3.3. NUMA architecture; Onyx2
  - 3.4. Massive parallelism, CM2
  - 3.5. Message Passing Interface (MPI)
  - 3.6. Multicore systems, Cell, General Purpose GPU Programming
  - 3.7. CUDA
4. Distributed computing
  - 4.1. Cluster computing
  - 4.2. Grid computing
  - 4.3. Cloud computing
  - 4.4. Pervasive computing
5. Peer-to-peer systems
  - 5.1. State variables
  - 5.2. Dynamics of peer-to-peer networks
  - 5.3. Design issues
  - 5.4. Design strategies for overlay schemes
  - 5.5. Popular overlay schemes (Napster, BitTorrent, eMule, JXTA, Skype, Chord, Kademia)
6. Peer-to-peer application development with Sip2Peer
7. Autonomic computing
  - 7.1. The four principles of autonomic computing
  - 7.2. MAPE-K
  - 7.3. NAM and NAM4J
  - 7.4. Distributed Remodeling Framework
8. Simulation techniques
  - 8.1. General concepts about simulation
  - 8.2. Discrete event simulation
9. Discrete event simulation with DEUS