

Informazioni generali sul corso di Sistemi Operativi



Corso di Laurea in Ingegneria Informatica
Università degli Studi di Napoli Federico II
Anno Accademico 2024/2025, Canale San Giovanni

Riferimenti



- Professore Associato, Dipartimento DIETI
- Aree di ricerca
 - Software bugs & vulnerabilities
 - Adversary emulation, red teaming
 - Security and AI
- Altre attività di formazione:
 - Software Security (Laurea Magistrale Ing. Informatica)
 - Data Security (Laurea Magistrale Data Science)
 - CyberChallenge.IT
 - Cyber HackAdemy @ UNINA



UNIVERSITA' DEGLI STUDI DI NAPOLI
FEDERICO II
DOCENTI



NATELLA ROBERTO

Profilo

[Riferimenti](#)

[Curriculum](#)

[Pubblicazioni](#)

Bacheca

[Avvisi](#)

[Orari ricevimento](#)

Didattica

[Insegnamenti](#)

[Schede insegnamento](#)

[Programmi ante 2022-23](#)

[Appelli d'esame](#)

[Orario delle lezioni](#)

[Iscrizione alle lezioni](#)

[Registro delle lezioni](#)

[Materiale didattico](#)

Riferimenti

Dipartimento

Dipartimento di Ingegneria elettrica e delle Tecnologie dell'Informazione

Ruolo

Professore di sistemi di elaborazione delle informazioni (ING-INF/05)

Email

roberto.natella@unina.it

Url Breve

<https://www.docenti.unina.it/roberto.natella>

Altre informazioni inserite dal docente

Sito web personale:

<http://wpage.unina.it/roberto.natella/>

Profilo LinkedIn:

<http://it.linkedin.com/in/robertonatella/>

GitHub:

<https://github.com/rnatella/>



Microsoft Teams

- Utilizzeremo **Microsoft Teams** per
 - Condividere il materiale del corso
 - Comunicare avvisi
 - Ricevimento



Codice Teams:
go8dwfa



Modalità Didattica - Indicazioni

- I corsi sono **esclusivamente in presenza**
- Salvo circostanze particolari:
 - Sciopero generale dei mezzi di trasporto pubblico
 - Condizioni meteo avverse (allerta rossa o arancione)



Obiettivi formativi del corso

- Fondamenti teorici dei sistemi operativi
 - Gestione delle risorse di calcolo (CPU, memoria, I/O)
 - Astrazioni fornite dai SO (processi, thread, filesystem, risorse IPC, etc.)
 - Problematiche di sicurezza
- Tecniche di **programmazione concorrente**
- Utilizzo del sistema operativo **Linux**

Approccio didattico



- Lezioni frontali sugli aspetti teorici
 - Domande/risposte
 - Discussione di esempi
 - Quiz
- Approfondimenti pratici sulla programmazione in ambiente **Linux**
- Esercitazioni in aula



github
SOCIAL CODING



Esercitazioni


- Esercitazione 0 (**Introduzione tools**)
- Esercitazione 1 (**Semafori**)
- Esercitazione 2 (**Monitor**)
- Esercitazione 3 (**Threads**)
- Esercitazione 4 (**Messaggi**)
- Esercitazione 5 (**Messaggi+Threads**)

Le esercitazioni sono parte del programma,
si raccomanda di svolgerle tutte!



Programma del corso

- Al termine del corso, sarà pubblicato un **programma**
 - Elenco dettagliato degli argomenti
 - Informazioni sugli esercizi da svolgere


DIPARTIMENTO DI INGEGNERIA ELETTRICA
E DELLE TECNOLOGIE DELL'INFORMAZIONE

Corso di Laurea in Ingegneria Informatica, Canale San Giovanni

CORSO DI SISTEMI OPERATIVI
Prof. Roberto Natella
Programma dell'A.A. 2021/2022

Ultimo aggiornamento: 17 dicembre 2021

Obiettivi Formativi

Il corso fornisce le conoscenze che sono alla base della teoria dei Sistemi Operativi e delle tecniche di programmazione concorrente. I contenuti teorici riguardano i modelli architetturali dei Sistemi Operativi, la gestione e lo scheduling dei processi e dei thread, le tecniche per la competizione e cooperazione tra processi, il deadlock, la gestione della memoria, la gestione dell'Input/Output, i File System e le problematiche di sicurezza. Le esercitazioni e le attività di laboratorio sono sviluppate in ambiente Linux e consistono in applicazioni di programmazione concorrente e la programmazione di moduli del kernel Linux.

Contenuti

Concetti Introduttivi. Evoluzione storica dei S.O. - Mono e multiprogrammazione - Batch, time sharing, real-time, mobile, cloud computing - Richiami di elementi di architettura a supporto di un S.O. (modalità di I/O, gestione delle interruzioni, modalità utente e supervisore, gerarchie di memoria) - Virtualizzazione delle risorse nei S.O. - Il kernel - Chiamata a supervisore - Architetture a livelli, monolitiche, modulari, microkernel.

*Lucidi delle lezioni: SO-1, SO-2, SO-3
Libro di testo ANCILLOTTI-1: Capitolo 1*

Gestione e Scheduling dei Processi e Thread. Concetto di processo - Stati di un processo - Descrittore di un processo - Code di processi - Il cambiamento di contesto - La gestione dei processi nei sistemi operativi Linux e Windows - Scheduling della CPU a breve, medio e lungo termine - Parametri di valutazione degli algoritmi di scheduling - Starvation - Preemption - Algoritmi First Come First Served, Round Robin, Shortest Process Next, Shortest Remaining Time, a code multiple con retroazione - Confronto tra algoritmi di scheduling monoprocesso - Scheduling tradizionale UNIX - Gli Scheduler O(1) e CFS del sistema operativo Linux - Scheduling nel sistema operativo Windows - Scheduling multiprocessore: Architetture SMP, multicore, e hyperthreading, scheduling con load sharing e dynamic load balancing, gang scheduling - Concetto di thread - Processi e thread - Stati di un thread - Thread a livello utente e a livello del nucleo - Modelli di programmazione multithreading - Primitive per la gestione dei threads - Cenni alla gestione dei thread nei sistemi Linux, Windows, Java.

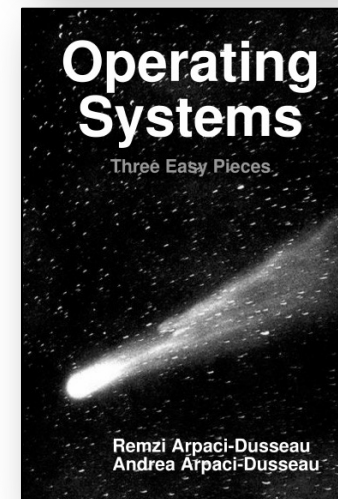
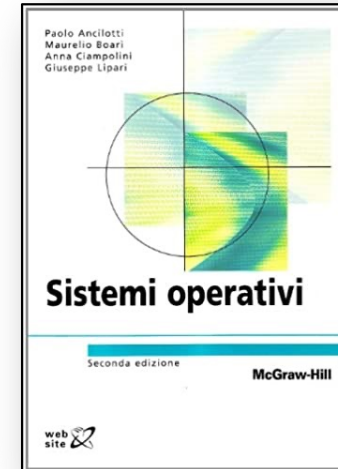
*Lucidi delle lezioni: SO-4, SO-5, SO-6, SO-7, SO-8
Libro di testo ANCILLOTTI-1: Capitoli 2, 9, 10
Dispensa b) "Scheduling del processore"
Dispensa k) "Process Scheduling in Linux"
Dispensa j) "Scheduling multiprocessore"*

1



Libri di testo consigliati

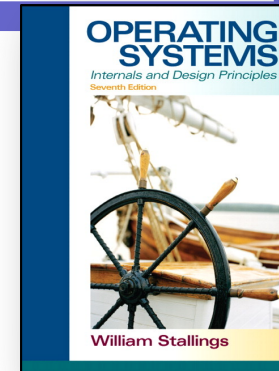
- Ancillotti, Boari, Ciampolini, Lipari
Sistemi Operativi
McGraw-Hill, 2a edizione, 2008
(in italiano)
- R.H. Arpaci-Dusseau,
A.C. Arpaci-Dusseau
***Operating Systems:
Three Easy Pieces***
www.ostep.org, v1.01, 2020
(gratuito in PDF, in lingua inglese)





Libri di testo (approfondimenti)

- Stallings - **Operating Systems: Internals and Design Principles** – Prentice Hall (in inglese)
- Silberschatz, Galvin, Gagne - **Sistemi operativi. Concetti ed esempi** – Pearson Education Italia
- Tanenbaum, Bos – **I moderni Sistemi Operativi** – Prentice Hall





Libri di testo (approfondimenti)

- Ancillotti, Boari
**Programmazione
concorrente e distribuita**
McGraw-Hill, 2007





Materiale on-line

- Il materiale del corso sarà reso disponibile tramite Teams, e tramite la piattaforma github.com
- https://github.com/rnatella/so_esempi



Materiale on-line

Prove di esame degli scorsi anni:

https://github.com/rnatella/esercizi_linux

Modello ad ambiente globale, produttore-consumatore e mutua esclusione

- [Una struttura dati stack thread-safe](#) [con soluzione]
- [Calcolo parallelo su un vettore condiviso](#) [con soluzione]
- [Produttore/consumatore con vettore di stato, con monitor](#) [con soluzione]
- [Produttore/consumatore con vettore di stato, con semafori](#) [con soluzione]
- [Produttore-consumatore con priorità](#) [con soluzione]
- [Coppia di buffer](#) [con soluzione]
- [Produttore/consumatore con elementi chiave-valore](#) [con soluzione]
- [Prelievi multipli](#) [con soluzione]
- [Produzione di un vettore variabile di elementi](#)
- [Pool di thread worker, con vettore circolare](#) [con soluzione]
- [Pool di thread worker, con vettore di stato](#) [con soluzione]
- [Pool di thread worker, con vettori circolari di operandi e risultati](#)
- [Allocatore di memoria](#)
- [Produttore/consumatore asimmetrico, con vettore di stato](#) [con soluzione]
- [Produttore/consumatore asimmetrico, con vettore circolare](#) [con soluzione]
- [Produttore/consumatore con vettore di stato, e diverse elaborazioni](#)
- [Media e deviazione standard di un vettore](#)
- [Vettore di stato, con più tipologie di prodotto e condition variables](#)
- [Vettore di condition variables](#) [con soluzione]
- [Prodotto scalare con vettori dinamici](#)
- [Gestione del sovraccarico](#) [con soluzione]



Materiale on-line

Sul portale "Federica"

<https://lms.federica.eu/enrol/index.php?id=224>

sono disponibili brevi video-lezioni tratte dal corso

The screenshot displays the Federica Web Learning interface. At the top, there's a navigation bar with the logo, a search bar, and links for 'Accedi', 'ITA', and 'ENG'. Below this, a menu bar includes 'TUTTI I MOOC', 'PARTNER', and 'BLOG'. The main content area is titled 'I Sistemi Operativi Moderni D. Cotroneo'. It features a section 'Il Corso' with a description of the course content, its start and end dates (1/9/2020 to 28/2/2021), and a list of topics. To the right, there's a 'Presentazione' section with a video player showing a preview of the first lesson. Below the video, there's a call to action: 'Il corso è aperto. Esplora l'anteprima e guarda i contenuti della prima lezione. Non perdere la prossima lezione. Iscriviti!'. At the bottom, there's a link to '1. Introduzione al corso' and a button labeled 'ISCRIVITI'.



Ricevimento studenti

- Il **ricevimento** si tiene indicativamente ogni settimana
- Salvo eventuali impegni concomitanti
- Via Teams o in presenza

Seleziona data e ora

marzo 2022



LUN	MAR	MER	GIO	VEN	SAB	DOM
	1	2	3	4	5	6
7	8	9	10	11	12	13
14	15	16	17	18	19	20
21	22	23	24	25	26	27
28	29	30	31			

OFFERTO DA
Calendly

Si prega di **prenotarsi** tramite il sito:

<https://calendly.com/roberto-natella/ricevimento-studenti>



Modalità d'esame

Prova pratica + Prova orale

La prova pratica è orientata ad accertare l'acquisizione dei principi di **programmazione concorrente** e delle capacità di **sviluppare software in ambiente Linux**.

PRENOTAZIONE OBBLIGATORIA

tramite il sistema SEGREPASS