



## SCHEDA DI TRASPARENZA DELL'INSEGNAMENTO

Insegnamento	Sistemi Operativi
Livello e corso di studio	Laurea Triennale in Ingegneria Elettronica e Informatica (L8)
Settore scientifico disciplinare (SSD)	ING-INF/05
Anno di corso	III
Numero totale di crediti	6 CFU
Propedeuticità	–
Docente	Gabriele Romaniello <a href="https://ricerca.unicusano.it/author/gabriele-romaniello/">https://ricerca.unicusano.it/author/gabriele-romaniello/</a> E-mail: <a href="mailto:gabriele.romaniello@unicusano.it">gabriele.romaniello@unicusano.it</a> (preferire l'invio di messaggi in piattaforma alle e-mail) Orario di ricevimento: Consultare il calendario videoconferenze sul sito d'Ateneo o contattare il docente in caso di necessità ricevimento aggiuntivo.
Presentazione	Il Corso di Sistemi Operativi ha lo scopo di far acquisire allo studente una buona conoscenza dei principi che governano la progettazione di tale tipologia di sistemi e permettono di modellarne il comportamento. Il Corso propone i concetti basilari dell'architettura hardware e software di un calcolatore, e li declina nello studio delle astrazioni e, più in generale, dei meccanismi di uso comune nei sistemi operativi, con particolare riferimento ai sistemi UNIX e Linux. Inoltre, obiettivo formativo del Corso è fornire allo studente una conoscenza nel dettaglio sulla programmazione concorrente ed i meccanismi alla sua base, come gli strumenti di sincronizzazione, le interfacce di programmazione di sistema, e la shell. Le Etivity associate al Corso sviluppano le competenze necessarie a progettare ed implementare sistemi basati sul kernel Linux.
Obiettivi formativi	Il Corso di Sistemi Operativi ha i seguenti obiettivi formativi: 1. Illustrare gli obiettivi di un sistema operativo ed il suo ruolo in un sistema di calcolo 2. Illustrare le principali tecniche e metodologie, sia software che hardware, a supporto di tali obiettivi 3. Illustrare le problematiche legate ai sistemi operativi moderni 4. Illustrare i principali strumenti di sincronizzazione, quali semafori (binari e contatori) e variabili di condizione, per la progettazione ed implementazione di programmi concorrenti 5. Illustrare l'uso della shell allo scopo di acquisire familiarità col sistema operativo Linux e padronanza nello scripting
Prerequisiti	La frequenza al Corso richiede la conoscenza dei concetti fondamentali di programmazione come apprese nel corso di Sistemi di Elaborazioni. Al riguardo, si consiglia di rivedere tali nozioni,



	<p>propedeutiche per l'apprendimento e l'approfondimento degli strumenti di sincronizzazione nei programmi concorrenti.</p>
Risultati di apprendimento attesi	<p><b>Conoscenza e capacità di comprensione (knowledge and understanding)</b> Lo studente al termine del Corso avrà conoscenza dei principi e concetti alla base dei sistemi operativi per calcolatori, ed avrà acquisito la capacità di comprendere problematiche quali l'organizzazione di un sistema di calcolo moderno, la gestione delle risorse di un sistema di calcolo (memoria, CPU, dispositivi esterni), e la gestione e la sincronizzazione dei processi e dei thread in un sistema multiprogrammato e a condivisione del tempo. Inoltre, tramite le Etivity incentrate sul kernel Linux gli studenti acquisiranno una conoscenza pratica delle più moderne tecniche nel campo dei sistemi operativi</p> <p><b>Applicazione delle conoscenze (applying knowledge and understanding)</b> Lo studente sarà in grado di utilizzare le metodologie apprese per analizzare le prestazioni di un sistema operativo in un particolare contesto applicativo; sarà inoltre in grado di formulare alternative o proporre soluzioni originali a problemi legati al funzionamento dei sistemi di calcolo complessi; saprà porre e sostenere argomentazioni nell'ambito dei sistemi operativi, evidenziando vantaggi e svantaggi di particolari soluzioni implementative. Le Etivity prevedono l'applicazione delle conoscenze teoriche a problemi pratici da risolvere in ambiente Linux.</p> <p><b>Capacità di trarre conclusioni (ability to draw conclusions)</b> Lo studente sarà in grado di seguire i trend moderni nell'ambito della progettazione di sistemi operativi, di raccogliere i dati necessari alla valutazione delle prestazioni di un particolare sistema operativo, e di interpretare i risultati della valutazione, e di elaborare i requisiti necessari alla progettazione di un nuovo sistema operativo, così come di valutare l'efficacia di diverse soluzioni alternative per individuare quella più appropriata al modello d'uso previsto.</p> <p><b>Abilità comunicative (communication skills)</b> Lo studente sarà in grado di descrivere e sostenere conversazioni su tematiche e problematiche relative ai moderni sistemi operativi, sempre adoperando una terminologia adeguata, ed individuando correttamente gli indicatori rilevanti nel confronto tra diversi sistemi operativi, allo scopo di offrire possibili soluzioni.</p> <p><b>Capacità di apprendere (learning skills)</b> Lo studente al termine del Corso avrà conoscenza delle nozioni fondamentali necessarie per la progettazione di sistemi operativi e la programmazione concorrente, e dell'importanza di adeguati supporti hardware. Tutto ciò gli consentirà di proseguire gli studi ingegneristici con un elevato grado di autonomia e gli fornirà le basi per poter apprendere quanto verrà proposto nei corsi specialistici di ingegneria informatica, con particolare riferimento agli argomenti legati alla progettazione e programmazione di sistemi di virtualizzazione delle risorse.</p>
Organizzazione dell'insegnamento	<p>Il Corso è sviluppato attraverso le lezioni preregistrate audio-video che compongono, insieme a slide e dispense, i materiali di studio disponibili in piattaforma. Sono poi proposti dei test di autovalutazione, di tipo asincrono, che corredano le lezioni preregistrate e consentono agli studenti di accertare sia la comprensione, sia il grado di conoscenza acquisita dei contenuti di ognuna delle lezioni. La didattica interattiva è svolta nel forum della "classe virtuale" e comprende 3 Etivity che</p>



	<p>applicano le conoscenze acquisite nelle lezioni di teoria alle attività di progettazione ed implementazione. In particolare, il Corso di Sistemi Operativi prevede 6 Crediti formativi. Il carico totale di studio per questo modulo di insegnamento è di circa 150 ore così suddivise in:</p> <ul style="list-style-type: none"><li>- circa 107 ore per la visualizzazione e lo studio del materiale videoregistrato;</li><li>- circa 36 ore di Didattica Interattiva per l'elaborazione e la consegna di 3 Etivity;</li><li>- circa 5 ore di Didattica Interattiva per l'esecuzione dei test di autovalutazione.</li></ul>
Contenuti del corso	<p><b>Modulo 1 – Richiami di Programmazione (impegno di 12 ore)</b> dove sono affrontati i seguenti argomenti: Introduzione ai sistemi operativi, Hardware del calcolatore, Linguaggio assembly (cenni), Linguaggio C (ripasso).</p> <p><b>Modulo 2 – Processi e Thread (impegno di 15 ore)</b> dove sono affrontati i seguenti argomenti: Concetto di processo, Comunicazione tra processi, Introduzione ai thread, Introduzione allo scheduling dei processi, Algoritmi per lo scheduling dei processi.</p> <p><b>Modulo 3 – Memoria virtuale (impegno di 14 ore)</b> dove sono affrontati i seguenti argomenti: Memoria fisica, Organizzazione della memoria virtuale (paginazione), Organizzazione della memoria virtuale (segmentazione), Gestione della memoria virtuale</p> <p><b>Modulo 4 – Programmazione concorrente (impegno di 28 ore)</b> dove sono affrontati i seguenti argomenti: Programmazione concorrente (Dekker/Peterson), Monitor, Paradigmi di programmazione concorrente, Problemi di stallo e di rinvio indefinito. Esercitazioni di programmazione in Pthread.</p> <p><b>Etivity 1 (15 ore di carico di studio)</b> – Progetto ed implementazione di un programma concorrente</p> <p><b>Modulo 5 – I/O e Filesystem (impegno di 15 ore)</b> dove sono affrontati i seguenti argomenti: Gestione dell'I/O (cenni), Gestione della memoria secondaria, File system, Metodi di allocazione dei file.</p> <p><b>Modulo 6 – Linux (impegno di 25 ore)</b> dove sono affrontati i seguenti argomenti: Kernel Linux, Shell. Esercitazioni di scripting.</p> <p><b>Etivity 2 (15 ore di carico di studio)</b> – Progetto ed implementazione di uno script Bash</p> <p><b>Etivity 3 (6 ore di carico di studio)</b> – Simulazione esame</p>
Materiali di studio	<p><b>Materiale didattico a cura del docente</b></p> <p>Il materiale didattico è presente in piattaforma e ricopre interamente il programma. Ciascun modulo contiene dispense, slide e videolezioni in cui il docente commenta le slide. Tale materiale contiene tutti gli elementi necessari per affrontare lo studio della materia.</p> <p><b>Testi consigliati</b></p>



	<p>A. Silberschatz; P. Baer Galvin; G. Gagne, "Sistemi operativi. Concetti ed esempi", Pearson, IX Ed. M. Kerrisk, "The Linux Programming Interface: A Linux and UNIX System Programming Handbook", No Starch Press</p>
<b>Modalità di verifica dell'apprendimento</b>	<p>L'esame consiste di norma nello svolgimento di una prova scritta tendente ad accertare le capacità di analisi e rielaborazione dei concetti acquisiti e nello svolgimento di una serie di attività (e-tivity) caricate all'interno delle classi virtuali. I risultati di apprendimento attesi circa le conoscenze della materia e la capacità di applicarle sono valutate dalla prova scritta, mentre le abilità comunicative, la capacità di trarre conclusioni e la capacità di autoapprendimento sono valutate in itinere attraverso le e-tivity.</p>
<b>Criteri per l'assegnazione dell'elaborato finale</b>	<p>L'assegnazione dell'elaborato finale avverrà sulla base di un colloquio con il docente in cui lo studente manifesterà i propri specifici interessi in relazione a qualche argomento che intende approfondire.</p>