



UNIVERSITÀ DEGLI STUDI
DI TRENTO

Dipartimento di Ingegneria e Scienza dell'Informazione
Anno accademico 2020/2021

Sistemi operativi [145830]

Nessun partizionamento

Corso di studio Ingegneria Informatica, delle Comunicazioni ed Elettronica

Ordinamento Ingegneria Informatica, delle Comunicazioni ed Elettronica

Percorso Informatica

Docenti: MAURO BRUNATO (Tit.), DOCENTE DA DEFINIRE

Numero ore: 96

Periodo: Secondo Semestre

Crediti: 12

Settori: ING-INF/05

Obiettivi formativi e risultati di apprendimento attesi

Obiettivo dell'insegnamento è quello di fornire i concetti fondamentali che sono alla base dei moderni sistemi operativi, con particolare enfasi sulla gestione della concorrenza e della memoria. Ogni concetto sarà trattato dal punto di vista teorico. Per i concetti più importanti la trattazione riguarderà anche aspetti più pratici in modo da familiarizzare con loro la complessità realizzativa.

Al termine dell'insegnamento, lo studente sarà in grado di comprendere l'architettura di un sistema operativo, di capire come sono progettate e implementate le principali funzionalità di un sistema operativo, di confrontare e valutare le diverse soluzioni progettuali e implementative.

Prerequisiti

Una conoscenza della programmazione in C e delle architetture dei calcolatori costituisce una base per la comprensione dei contenuti del corso.

Contenuti/programma del corso

- * Introduzione: Ruolo del sistema operativo e sua evoluzione. Elementi architettureali. Struttura e funzioni di un sistema operativo.
- * Gestione dei Processi: Processi. Stati dei processi. Cambiamento di contesto. Creazione e terminazione di processi. Thread: thread a livello utente e a livello kernel. Cooperazione e comunicazione fra processi: memoria condivisa, messaggi. Comunicazione diretta ed indiretta.
- * Scheduling: Modello a ciclo di burst di CPU-I/O. Scheduling a lungo, medio, breve termine. Scheduling con prelazione e cooperativo. Criteri di scheduling. Algoritmi di scheduling: FCFS, SJF, a priorità, HRRN, RR, a code multiple con e senza feedback. Valutazione degli algoritmi: modelli deterministici e probabilistici, simulazione.
- * Sincronizzazione fra processi: Coerenza di dati condivisi, operazioni atomiche. Sezioni critiche. Approccio software alla mutua esclusione: algoritmi di Peterson e Dekker, algoritmo del panettiere. Supporto hardware per la mutua esclusione: test and set, swap. Costrutti per sincronizzazione: semafori e monitor. Alcuni problemi tipici di sincronizzazione: produttore/consumatore, lettori/scrittore, problema dei filosofi.
- * Deadlock: Condizioni per l'insorgere di un deadlock. Rappresentazione dello stato di un sistema con grafi di allocazione. Tecniche di prevenzione, rilevazione e ripristino. Algoritmo del banchiere.
- * Gestione della memoria: Memoria primaria. Indirizzamento logico e fisico. Rilocalizzazione, binding degli indirizzi. Swapping. Allocazione contigua della memoria. Frammentazione interna ed esterna. Paginazione. Supporti hardware alla paginazione: TLB. Tabella delle pagine. Paginazione a più livelli. Segmentazione. Tabella dei segmenti. Segmentazione con paginazione.
- * Memoria Virtuale: Paginazione su richiesta. Gestione di page fault. Algoritmi di sostituzione delle pagine: FIFO, ottimale, LRU, approssimazioni LRU. Buffering di pagine. Allocazione di frame in memoria fisica, allocazione locale o globale. Thrashing. Località dei riferimenti. Modello del working set. Controllo della frequenza di page fault. Blocco di pagine in memoria.
- * Memoria secondaria Struttura logica e fisica dei dischi. Tempo di latenza. Scheduling del disco: algoritmi FCFS,



UNIVERSITÀ DEGLI STUDI DI TRENTO

SSTF, SCAN, C-SCAN, LOOK, C-LOOK. Gestione della memoria di paginazione. Strutture RAID.

* File System: Concetto di file, attributi e operazioni relative. Tipi di file. Accesso sequenziale e diretto. Concetto di directory. Struttura di directory. Protezioni nell'accesso a file. Attributi e modalità di accesso. Semantica della consistenza. Struttura di un file system. Montaggio di un file system. Metodi di allocazione dello spazio su disco: contiguo, concatenato, indicizzato. Gestione dello spazio libero su disco: tramite vettore di bit, tramite liste. Realizzazione delle directory: liste lineari, tabelle hash.

* Sistema di I/O: Sistemi di Input/Output, Hardware per I/O. Tecniche di I/O: programmato, con interrupt, con DMA. Device driver e interfaccia verso le applicazioni. Servizi del kernel per I/O: scheduling, buffering, caching, spooling.

Casi di studio:

- * Unix e Linux
- * Windows
- * Progetto di laboratorio usando Linux

Metodi didattici utilizzati e attività di apprendimento richieste allo studente.

L'attività didattica comprende lezioni frontali in aula che illustrano i principi e i concetti teorici.

Durante le lezioni in aula verranno anche presentati esempi di programmi e di sistemi operativi esistenti come utile ausilio per la comprensione della teoria dei sistemi operativi.

Oltre alle lezioni frontali in aula, il corso prevede ogni settimana anche lezioni in laboratorio, in cui verranno presentati, discussi e realizzati dagli studenti esercizi pratici per comprendere come sono realizzati alcuni concetti teorici discussi durante le lezioni in aula.

Durante le lezioni in laboratorio, gli studenti dovranno anche realizzare in gruppo un progetto relativo ad un argomento inerente ai sistemi operativi che include una fase implementativa. Il progetto è parte integrante del corso. Il materiale didattico (lecture notes, esercizi) è reso disponibile nel sito del corso sul sistema ESSE3.

Metodi di accertamento e criteri di valutazione

L'esame consiste in una prova scritta e la realizzazione di un progetto.

La prova scritta, valutata da 0 a 30/30, comprende esercizi per la soluzione di problemi di gestione della memoria, del file system, problemi di sincronizzazione e concorrenza e domande a carattere teorico a risposta aperta volte a valutare la capacità dello studente di esporre in modo critico i concetti visti a lezione.

Il progetto, valutato da 0 a 30/30, deve essere realizzato durante le lezioni in laboratorio del corso e discusso subito al termine dell'insegnamento. La valutazione del progetto è individuale.

Il progetto è volto a valutare la capacità di applicare e valutare in modo critico le conoscenze acquisite durante le lezioni teoriche.

Il voto finale è dato dalla media del voto della prova scritta e del voto del progetto.

Testi di riferimento/Bibliografia

* Operating System Concepts 9th Edition, Abraham Silberschatz, Peter Baer Galvin, Greg Gagne, John Wiley & Sons, Inc., ISBN: 0-471-69466-5, 2005 (in inglese)

in alternativa la versione in italiano

Sistemi Operativi, 10 Edizione, Abraham Silberschatz, Peter Baer Galvin, Greg Gagne, Pearson Education Italia

* Understanding the Linux Kernel, D.P. Bovet & M. Cesati, 3 Edizione (in italiano), O'Reilly, ISBN: 0-596-00565-2

Utili riferimenti sono anche:

* S. Tanenbaum, Modern Operating Systems, 4th edition, Prentice-Hall. ISBN 0-13-031358-0 (in inglese)

in alternativa la versione in italiano

S. Tanenbaum, Modern Operating Systems, Prentice-Hall, 4' Edizione italiana Moderni sistemi operativi, Jackson Libri.

* P. Ancillotti, M. Boari Principi e tecniche di programmazione concorrente, Ed. UTET Libreria

* The C Programming Language, Second Edition, Brian W. Kernighan e Dennis M. Ritchie. Prentice Hall, Inc., 1988. ISBN 0-13-110362-8

* Linux for Programmers and Users, G. Glass and K. Ables, Pearson Prentice Hall, ISBN: 0-13-185748-7.

Altre informazioni

Stampa del 04/09/2020