

Corso di Sistemi Operativi (A.A. 2008-2009)

Prerequisiti

Il corso presume la conoscenza del linguaggio C++ a livello intermedio, e la conoscenza a livello utente dei sistemi operativi Microsoft.

Propedeuticità obbligatorie

E' obbligatorio il sostenimento preliminare degli esami di Programmazione a Oggetti e Architettura dei Calcolatori.

Obiettivi

Le nozioni primariamente trasmesse allo studente sono:

1. I concetti di base di architettura dei sistemi operativi, con particolare riferimento a gestione di memoria, dispositivi hardware, e unità di calcolo; sono fatti specifici riferimenti ai due sistemi operativi Windows e Linux.
2. L'amministrazione di sistema e programmazione in ambiente Unix, con particolare riguardo a Linux, attraverso l'insegnamento di un adeguato linguaggio di scripting quale Perl.
3. Le principali metodologie di programmazione concorrente.

Metodi di valutazione

Lo studente viene valutato attraverso due prove. Una prima prova scritta include l'implementazione di un algoritmo parallelo che risolva un problema dato. Altresì viene sottoposto un quesito a risposta aperta sui fondamenti di sistemi operativi o, in alternativa, un breve esercizio sull'implementazione di parti del sistema operativo. La seconda prova viene svolta al calcolatore e prevede l'implementazione di alcuni script di manutenzione e amministrazione di un sistema Linux, da redigere in Perl.

Programma del corso

- Definizione di sistema operativo. Componenti base di un sistema operativo. Meccanismi di interrupt e DMA.
- Concetto di processo e di thread. Operazioni di task switching. Il Program Control Block.
- Diagramma a stati dell'attività di un thread. Multitasking con e senza prelazione. Scheduling dei processi nell'architettura Win32.
- Sincronizzazione tra processi. Monitor. Problemi tipici di sincronizzazione: buffer limitato, lettori e scrittori, problema dei cinque filosofi.
- Caratterizzazione delle situazioni di stallo tra processi. Grafo dell'allocazione delle risorse. Situazioni di starvation.
- Gestione della memoria. Paginazione: politiche di swap in e swap out delle pagine. Protezione della memoria e separazione dei processi.
- Caricamento in memoria di programmi. Rilocazione. Chiamata a funzioni di sistema.
- File Systems: strutture dati di supporto alla gestione di file e directory. Mappatura tra dispositivi fisici e entità su livello di astrazione superiore.
- Layout del file system Linux. Attributi e permessi di accesso. Gestione dei processi.
- Comandi shell Linux: gestione dei file e dei permessi di accesso, piping e redirectione. Comandi di gestione dei processi. Avvio di processi in background. Variabili d'ambiente, passaggio di parametri su linea di comando.
- Il linguaggio Perl: costrutti, strutture dati native, espressioni regolari, interfacciamento con il sistema operativo ospite.
- Struttura del registry di Windows. Primitive di accesso al registry.

- Reverse engineering del codice e analisi dei software malware. Tool per il monitoraggio dell'attività dei processi (regmon, filemon, procexp).

Linguaggio di riferimento: C++ (Compilatore Microsoft Visual C++, GNU C++), Perl

Testi di riferimento:

- H.M.Deitel, P.J.Deitel, D.R.Choffnes, *Sistemi Operativi*, Terza edizione, Pearson Education, 2005.
- Randal L. Schwartz, Tom Phoenix, Brian d. foy *Learning Perl*, O' Reilly.
- Solomon, Russinovich. *Inside Microsoft Windows 2000 (III edizione)*. Microsoft Press
- Dispense fornite dal docente.

Sito web di riferimento: http://www.mat.unical.it/informatica/Reti_di_Calcolatori