



UNIVERSITÀ DI PISA

ARCHITETTURE E SISTEMI OPERATIVI

MASSIMO TORQUATI

| | |
|-----------------|-------------|
| Anno accademico | 2022/23 |
| CdS | INFORMATICA |
| Codice | 725AA |
| CFU | 15 |

| | | | | |
|-------------------------------------|-----------|---------|-----|---------------------------------------|
| Moduli | Settore/i | Tipo | Ore | Docente/i |
| ARCHITETTURE E SISTEMI OPERATIVI | INF/01 | LEZIONI | 120 | GABRIELE MENCAGLI MASSIMO TORQUATI |

Obiettivi di apprendimento

Conoscenze

Lo studente acquisirà conoscenze relative ai fondamenti scientifici e tecnologici delle Architetture dei Sistemi di Elaborazione e dei Sistemi Operativi moderni. Relativamente alle Architetture, verranno presentate le caratteristiche dei diversi livelli dei sistemi, le loro relazioni con gli strumenti di programmazione e la strutturazione dei sistemi in termini di moduli concorrenti e cooperanti. Relativamente ai Sistemi Operativi, verranno presentati i servizi offerti dai sistemi e gli algoritmi e le tecniche utilizzate nella loro implementazione. Lo studente, acquisirà inoltre i concetti e le metodologie della programmazione concorrente con threads e processi e di sistema utilizzando il linguaggio C.

Modalità di verifica delle conoscenze

La valutazione avverrà tramite una prova scritta ed una prova orale. Nella prova scritta verrà valutata la capacità dello studente di comprendere e di risolvere alcuni esercizi che prevedono sia la progettazione di semplici moduli con relativa valutazione delle performance che stati ipotetici in un sistema operativo che dovranno essere valutati e risolti attraverso le tecniche presentate nel corso ed utilizzando la terminologia appropriata.

Metodi utilizzati per la verifica:

- Test periodici (prove intermedie)
- Prova finale: scritto (per chi non ha superato le prove intermedie) ed orale

Ulteriori informazioni:

Sono previste delle prove di verifica intermedia. Se superate con esito positivo, le prove intermedie sostituiscono la prova finale scritta.

Capacità

Lo studente svilupperà la capacità di progettare e valutare semplici sistemi di elaborazione e la capacità di disegnare e scrivere programmi concorrenti sia con threads che con processi che utilizzano chiamate di sistema POSIX per l'uso dei dispositivi e per l'interazione con il Sistema Operativo.

Modalità di verifica delle capacità

Le capacità acquisite dallo studente verranno verificate tramite:

- Home work con auto valutazione
- Prove di verifica intermedia
- Prova finale scritta e prova orale

Comportamenti

Il corso non si pone obiettivi di apprendimento in termini di comportamenti attesi.

Modalità di verifica dei comportamenti

Non sono previste prove per la verifica dei comportamenti.

Prerequisiti (conoscenze iniziali)

Buona conoscenza della programmazione con linguaggi imperativi. Saper valutare la complessità degli algoritmi. Nozioni di base di analisi matematica.



UNIVERSITÀ DI PISA

Indicazioni metodologiche

Le lezioni avvengono in aula alla presenza del docente. Le attività di apprendimento comprendono:

- seguire le lezioni di teoria e le esercitazioni
- svolgere gli homework
- discutere con i docenti eventuali problematiche o necessità di chiarimenti
- studio individuale

Tutto il materiale didattico viene reso disponibile sul Microsoft Teams del corso. Gli studenti possono iscriversi al Teams (usando le loro credenziali UNIPI) usando il codice **9aihs9y**

Frequenza delle lezioni: fortemente raccomandata

Metodi di erogazione della didattica:

- lezioni ed esercitazioni frontali
- ricevimento studenti

Programma (contenuti dell'insegnamento)

Durante il corso verranno affrontati i seguenti argomenti:

- fondamenti dei sistemi di elaborazione
- linguaggi per la descrizione dell'hardware
- assembler
- micro architettura di base
- performance dei processori
- gerarchie di memoria
- parallelismo a livello di istruzione (ILP) e cenni ai sistemi multicore
- fondamenti dei sistemi operativi
- tecniche per il coordinamento e la gestione di risorse
- tecniche e metodologie per la programmazione di sistema
- meccanismi di sincronizzazione
- meccanismi di astrazione e strutturazione a livelli
- gestione dell'IO
- gestione della memoria
- chiamate di sistema (POSIX), (p)thread
- gestione del file system
- schedulazione
- esempi di implementazione dal mondo Unix e Windows

Bibliografia e materiale didattico

- S. Harris, D. Harris, "Digital design and computer architecture: ARM edition", Morgan Kaufmann 2017. Anche in versione italiana pubblicata da Zanichelli
- T. Anderson, M. Dahlin : "Operating Systems: principles and practice", Recursive Books Ltd, 2013 (disponibile solo in Inglese)
- Materiale didattico fornito durante le lezioni (slides, appunti, esercizi) e condiviso sulla piattaforma Microsoft Teams del corso.

Indicazioni per non frequentanti

Gli studenti non frequentanti hanno accesso a tutto il materiale didattico (inclusa la registrazione delle lezioni) sul Microsoft Teams del corso. Gli studenti possono iscriversi al Teams (usando le loro credenziali UNIPI) usando il codice **9aihs9y**

Modalità d'esame

La valutazione dello studente avverrà tramite una prova scritta ed una prova orale con i docenti.

Durante il corso si svolgeranno, alcune prove scritte di verifica intermedia. Lo studente che ottiene un voto sufficiente alle prove di verifica intermedia è esentato dallo svolgimento della prova scritta, e dovrà sostenere solo la prova orale in uno degli appelli dello stesso anno accademico.

La validità del voto conseguito con le prove di verifica intermedia cessa al termine dell'anno accademico. Ogni prova di verifica intermedia consiste nello svolgimento di alcuni esercizi riguardanti gli argomenti trattati nel corso fino a quel momento.

Si fa presente che per l'ammissione a tutte le prove d'esame, incluse le prove di verifica intermedia, è obbligatoria l'iscrizione secondo le modalità stabilite dal corso di laurea.