

# Programmazione Didattica Preventiva

Docenti: **Prof. FRACCOLA Christian; Prof. ATTOLINO Domenico**

Materia: **Tecnologie e Progettazione di Sistemi Informatici e di Telecomunicazione (TPSIT)**

Classe: **4E (ITT) – indirizzo Informatica e Telecomunicazioni (articolazione Informatica)**

Anno Scolastico: **2018/2019**

Ore Settimanali: **4**

**Metodi** Lezione frontale in interazione. Lezione di laboratorio. Alcune lezioni potranno essere tenute in lingua inglese.

**Mezzi** Dispense da parte del docente (anche in forma elettronica), esercitazioni al computer, software (con prevalenza di software open-source o freeware), libro di testo, Internet, strumentazione di laboratorio.

**Spazi** Aula. Laboratorio di Informatica.

**Verifiche** Test a scelta multipla o con domande aperte, interrogazioni. Alcune verifiche potranno essere somministrate (integralmente o parzialmente) in lingua inglese.

**Obiettivi Trasversali** Conoscenza della terminologia tecnica in italiano ed in inglese. Saper leggere e capire documentazione tecnica di livello medio in inglese.

**Collegamenti** Informatica. Sistemi e Reti. TPSIT. Inglese (microlingua).

## Interdisciplinari

**Ripasso e approfondimento:** Gestione del processore: definizione di processo, stati di un processo, diagramma degli stati di un processo, PCB e suo contenuto (PID, stato corrente, program counter, registri, priorità, puntatori a risorse, puntatori alla memoria), schedulazione dei processi, algoritmi di scheduling (FCFS, SJF, con priorità, round robin, MLFQ), concetti di context switch e di dispatcher. Gestione della memoria centrale: caricamento di un programma in memoria( indirizzo fisico, offset, indirizzo logico, binding, linking), allocazione della memoria: tecnica a partizione fissa (vantaggi e svantaggi), tecnica a partizione variabile (vantaggi e svantaggi); frammentazione interna, frammentazione esterna; memoria virtuale, tecnica di paginazione della memoria virtuale (paginazione, segmentazione), l'MMU. Il File System: Componenti del File System: definizione di File System e di file, descrittore del file (nome, identificatore, tipo, locazione, proprietario, permessi, dimensione), operazioni sui file(creazione, ricerca, scrittura, lettura, posizionamento, cancellazione, troncamento, accodamento, rinomina, spostamento, copia), metodi di accesso (sequenziale, diretto, indicizzato) directory (sua struttura e operazioni eseguibili su essa), diritti e protezione dei file.

**Struttura del File System:** residenza del File System, allocazione dei file (contigua, concatenata, indicizzata, FAT), vantaggi e svantaggi dei tipi di allocazione, cenni su NTFS e FAT32. Sicurezza del File System: la struttura RAID, RAID-0, RAID-1, RAID0+1, RAID2. Struttura delle memorie di massa: il disco fisso di tipo magnetico e sue caratteristiche fisiche (dischi, facce, tracce, settori, cluster, testine), tempi di accesso all'informazione memorizzata su disco (tempo di seek, tempo di latenza), dischi ottici (CD e DVD), struttura fisica di un disco ottico (spirale, pit, land). Processi e risorse: classificazione dei processi (indipendenti,cooperanti, in competizione), le risorse, legami tra processi e risorse, classificazione delle richieste( secondo il numero: singola, multipla; secondo il tipo di richiesta: bloccante, non bloccante), classificazione delle assegnazioni( statica e dinamica), classificazione delle risorse (mutua esclusività: seriali, non seriali; per modalità di utilizzo: preemptive e nonpre-emptive), grafo di Holt, riducibilità di un grafo di Holt.

**Thread:** classificazione dei processi (leggeri e pesanti), definizione di thread, il TCB e le sue componenti, evoluzioni dei sistemi basati su single threading o multithreading, diagramma degli stati di un thread. Elaborazione concorrente dei processi: processi sequenziali e paralleli, grafo delle precedenze, ordinamento parziale e ordinamento totale, semplificazione delle precedenze, linguaggi di programmazione concorrenti, uso dei costrutti: forkjoin, join(count), cobegin-coend, equivalenza e limiti dei costrutti fork-join e cobegin-coend. Comunicazione tra processi: modelli di interazione concorrente (modello a memoria comune, modello a scambio di messaggi, modello client-server) classificazione dei modelli a scambio di messaggio (comunicazione sincrona, comunicazione asincrona: rendez-vous stretto, rendez-vous esteso; comunicazione asimmetrica, comunicazione asimmetrica). Cenni sulla sincronizzazione tra i processi: i semafori, starvation e deadlock.

## *Moduli Didattici*

<i>Modulo</i>	<i>Contenuti</i>	<i>Obiettivi Minimi</i>	<i>Tempi (ore)</i>
<b>Sistemi Operativi</b>	Introduzione ai sistemi operativi. Programmi, processi e risorse di un sistema. Interprete dei comandi di un sistema operativo, con enfasi sulla shell di Windows e di Linux (BASH).	Conoscere i concetti teorici e le problematiche ad essi associate. Saper utilizzare l'interprete dei comandi di Windows e di Linux per interagire con il sistema operativo.	10
<b>Programmazione della shell dei sistemi operativi.</b>	Programmazione della shell di Windows.	Conoscere i concetti teorici e le problematiche ad essi associate. Saper realizzare script della shell di livello medio.	20
<b>Linguaggi di programmazione</b>	Classificazione ed aspetti realizzativi dei linguaggi di programmazione. Linguaggi compilati, interpretati ed approcci ibridi.	Conoscere i concetti teorici e le problematiche ad essi associate.	15
<b>Elementi di Elettronica Generale.</b>	Conoscenza del funzionamento ai terminali dei principali componenti elettronici e dei relativi circuiti applicativi: diodi a giunzione, transistori (BJT e MOSFET), amplificatori operazionali. Lo studio è finalizzato particolarmente alla conoscenza dei circuiti di interfacciamento con i microcontrollori.	Conoscenza operativa dei componenti e dei circuiti studiati.	10
<b>Programmazione dei microcontrollori e piattaforma Arduino</b>	Conoscenza dell'ambiente di sviluppo Arduino soprattutto in relazione alle funzionalità specifiche della scheda a microcontrollore Arduino UNO. Conoscenza delle tecniche di programmazione C/C++ dei microcontrollori. Programmazione ed interfacciamento con il mondo esterno della scheda Arduino UNO.	Conoscere i concetti teorici e le problematiche ad essi associate. Saper montare semplici circuiti da collegare alla scheda Arduino UNO e saper programmare	25