

Programma degli argomenti svolti

anno scolastico: **2013/2014**
classe: **3F (ITT)**
materia: **Tecnologia e Progettazione di Sistemi Informatici e di Telecomunicazioni**
docente: **Prof. Lorenzo Donati, Prof. Domenico Attolino**

Nota: il presente programma non riflette pedissequamente la distribuzione temporale degli argomenti durante l'anno scolastico, bensì è organizzato secondo blocchi tematici omogenei. In grassetto sono indicati gli argomenti indispensabili per il raggiungimento degli obiettivi didattici minimi.

□ **Rappresentazione ed elaborazione dell'informazione digitale**

○ **Rappresentazione dell'informazione digitale**

- ◆ Sistemi di numerazione. I sistemi decimale, binario, ed esadecimale.
- ◆ Rappresentazione dei numeri naturali in forma binaria. Operazioni di somma e sottrazione tra numeri naturali in binario.
- ◆ Rappresentazione dei numeri interi in complemento a 2. Somme e sottrazioni tra numeri interi in complemento a 2.
- ◆ Conversione tra sistemi di numerazione: da binario a decimale e viceversa, da binario ad esadecimale e viceversa, da binario ad esadecimale e viceversa.
- ◆ Esempi sul calcolo degli indirizzi di memoria usando il sistema esadecimale.

○ **Reti logiche e struttura del calcolatore elettronico**

- ◆ Analisi di alcuni datasheet di circuiti integrati digitali, con particolare attenzione ai tempi di ritardo ed ai diagrammi di temporizzazione.
- ◆ Multiplexer, demultiplexer e decoder binari.
- ◆ Reti logiche programmabili.
- ◆ Reti sequenziali sincrone ed asincrone.
- ◆ Problematiche legate alla progettazione di reti sequenziali complesse: primato delle reti sincrone (economie di scala, affidabilità di funzionamento, facilità d'uso, abbattimento dei costi, etc.).

□ **Laboratorio di Progettazione Informatica e di Telecomunicazioni**

○ **Strumenti software di impiego generale**

- ◆ Uso del programma Graph per il tracciamento di grafici di funzioni. Applicazione alla visualizzazione dei segnali deterministici più comuni nell'ambito delle telecomunicazioni.

○ **Fondamenti di programmazione LabView**

- ◆ Elementi base sugli strumenti virtuali LabView e sull'interfaccia grafica di programmazione.

- ◆ Uso dell'ambiente LabView per la realizzazione di semplici simulatori di reti logiche.

○ **Fondamenti di programmazione Scilab**

- ◆ Operazioni aritmetiche di base sui numeri reali e sui numeri complessi.
- ◆ comandi e funzioni base: *help, clc, clf, mode, typeof*.
- ◆ Variabili speciali predefinite: *%e, %pi, %i, %T, %F*.
- ◆ Funzioni matematiche di base: *sqrt, exp, log, log10, abs, ceil, floor, int, round, dec2bin, real, imag, sin, cos, tan*.
- ◆ Stringhe ed operazioni su di esse: concatenazione (operatore +), conversione (funzione string).
- ◆ Funzioni per l'I/O: *mprintf, disp, input*.
- ◆ Matrici: sintassi e semantica. Operazioni e funzioni base sulle matrici: *length, size*, trasposizione (operatore '), operazioni algebriche. Concatenazione di matrici, estrazione di elementi e di sottomatrici.
- ◆ Funzioni per la generazione di matrici particolari: *linspace, ones, zeros*.
- ◆ Funzionalità grafiche: funzioni *plot* e *replot*.
- ◆ Controllo di flusso: istruzioni *if, while* e *for*.
- ◆ Definizione di funzioni in Scilab.
- ◆ Uso di Scilab per la visualizzazione dei segnali e la loro manipolazione.

○ **Circuiti in corrente continua**

- ◆ Concetto di tensione, corrente e resistenza.
- ◆ Legge di Ohm.
- ◆ Montaggio di semplici circuiti resistivi ed esecuzione di misure su di essi.
- ◆ Funzionamento a grandi linee di un diodo LED e suo circuito di alimentazione in CC mediante resistenza limitatrice.
- ◆ Montaggio e collaudo di circuiti di accensione di diodi LED.

□ **Linguaggi Informatici**

○ **Classificazione dei linguaggi**

- ◆ Linguaggi dichiarativi – caratteristiche ed esempi: linguaggi di markup, linguaggi funzionali e linguaggi logici. Cenni storici ed applicativi sui seguenti linguaggi: HTML, LISP, Scheme, PROLOG, SQL, linguaggi degli spreadsheet, TeX/LaTeX, XML, PHP.
- ◆ Linguaggi imperativi – caratteristiche ed esempi: linguaggi di Von Neumann e linguaggi ad oggetti. Cenni storici ed applicativi sui seguenti linguaggi: Fortran, Basic, C, C++, Java, Forth, Cobol, Pascal, Perl, Javascript, Lua, Python.

○ **Caratteristiche ed implementazione dei linguaggi**

- ◆ Tipi di memoria dal punto di vista dell'implementazione di un linguaggio: memoria automatica e stack; memoria dinamica e heap (con enfasi sul C/C++). Cenni sulla funzione standard *C malloc*.
- ◆ Linguaggi dinamici (linguaggi con sistema di tipi dinamico e gestione automatica della memoria): cenni sul Lua, Javascript e Python. Cenni sui sistemi di garbage collection. Vantaggi e svantaggi dei linguaggi dinamici.
- ◆ Esecuzione del codice: approccio compilativo, approccio interpretativo, approcci ibridi.
- ◆ Compilatori ed interpreti.
- ◆ Macchine virtuali ed interpreti di bytecode (Java, Javascript, Python, Lua). Compilazione JIT.
- ◆ Meccanismo di creazione del codice eseguibile a partire dal codice sorgente in C/C++: compilazione e linking. File oggetto. Cenni al linking dinamico.

□ **Fondamenti sui sistemi operativi**

○ **Concetti generali**

- ◆ Concetto di condivisione e gestione delle risorse del sistema. Sistemi operativi monoutente, multiutente, uniprocesso, multiprocesso.

- ◆ Overhead di sistema.
 - ◆ Protocolli di comunicazione.
 - ◆ Caching e gestione della CPU. Gestione della memoria.
 - ◆ Dispositivi fisici e logici. Driver di dispositivo.
 - ◆ Serializzazione e spooling. Code di stampa.
 - ◆ Chiamate di sistema.
 - ◆ Filesystem.
 - ◆ Concetto di struttura dati con disciplina di gestione LIFO: gli stack in astratto e lo stack hardware gestito dalla CPU. Problemi di gestione di uno stack (overflow, underflow).
 - ◆ Il meccanismo delle interruzioni.
 - ◆ Il controllore DMA.
 - ◆ I/O sincrono ed asincrono.
 - ◆ Processi nei sistemi multiprocesso e commutazione di contesto.
 - ◆ Processi genitori e processi figli.
 - ◆ Ambiente di un processo. Ereditarietà dell'ambiente.
- ### ○ **Nozioni fondamentali sull'interprete dei comandi dei sistemi Windows (shell di Windows)**
- ◆ Comandi base: *help, cd (chdir), cls, copy, date, del, dir, echo, md (mkdir), more, move, rd (rmdir), ren (rename), time, tree, type, ver, vol*.
 - ◆ Variabili d'ambiente e loro manipolazione mediante il comando *set*.
 - ◆ Concetto di percorso di ricerca degli eseguibili, comando *path* e variabile di ambiente *PATH*.
 - ◆ Cenni all'uso dei comandi *cmd* e *start* per eseguire programmi.

Bolzano, 6 giugno 2014

Il Docente:

Prof. Lorenzo Donati

Prof. Domenico Attolino

Gli Studenti:

<i>Cognome e Nome</i>	<i>Firma</i>