

# Programma del corso di "Laboratorio di Calcolo" AA 2000-2001

L'obiettivo del corso è quello di fornire le nozioni di base necessarie alla realizzazione di programmi di media complessità utili per il calcolo scientifico e l'analisi dati. Sebbene nelle esercitazioni si utilizzerà il linguaggio C in ambiente Unix (Linux) il corso non si propone come un corso di C. Si cercherà al contrario di mettere in evidenza il legame tra le varie tecniche di programmazione e le operazioni effettivamente svolte dal calcolatore, in modo che i concetti introdotti non risultino limitati ad un ambito particolare ma siano il più possibile di utilità generale. Lo studio della sintassi e dei dettagli del linguaggio C verrà il più possibile demandato alle esercitazioni di laboratorio individuali ed inserito nel contesto delle problematiche di calcolo che verranno via via affrontate.

Gli elementi di struttura e funzionamento dei sistemi a microprocessore saranno trattati a livello elementare, e con lo scopo di illustrare il legame tra le operazioni svolte fisicamente dalla macchina e le istruzioni dei linguaggi evoluti.

Verranno introdotte librerie software per la realizzazione di grafici ed istogrammi e verranno illustrate le tecniche di base del calcolo numerico (integrazione, soluzione di equazioni differenziali, metodi di Montecarlo). Inoltre verranno illustrati i principi base del funzionamento della porta seriale e parallela che saranno poi utilizzate in laboratorio per la realizzazione di semplici esperienze (misure di tensione e corrente, generazione di tensioni tramite DAC, misure di tempo).

La prova di esame sarà divisa in due parti: lo svolgimento di una tesina (1 credito), a gruppi di 2 studenti, consistente nella realizzazione di un programma che utilizzi le tecniche illustrate nel corso (applicazioni ad argomenti trattati nei corsi di Fisica I, Analisi I Geometria ed Esperimentazioni I) e lo svolgimento di una prova pratica individuale mirata a valutare la capacità di realizzazione e di messa a punto di semplici programmi.

## **Programma delle Lezioni (Primo Semestre, 2 cr.)**

- Introduzione al corso; scopi e metodologie
- Segnali analogici e digitali; rappresentazione dei numeri interi con segnali digitali (numerazione in base 2); la rappresentazione in complemento a due.
- Struttura dei sistemi a microprocessore; la CPU; connessione CPU-memoria; i programmi in linguaggio macchina; connessione con dispositivi esterni (mappatura in memoria ed interruptus).
- Svantaggi dei linguaggi macchina; linguaggi evoluti; librerie di programmi; il compilatore ed il linker; i sistemi operativi.
- Rappresentazione dei dati in memoria: cenni alle rappresentazioni a virgola mobile, variabili booleane, codice ASCII; introduzione all'algebra di Boole; operazioni logiche fondamentali. Cenni ai sistemi combinatori e al flip-flop.
- Struttura generale dei linguaggi evoluti: sintassi, struttura, controllo di flusso, variabili, funzioni.

- Caratteristiche delle variabili nei linguaggi evoluti: tipo, dimensione, modo di allocazione, “scope”; definizione di tipi composti; puntatori e allocazione dinamica delle variabili.
- Programmazione strutturata; scomposizione del programma in moduli e functions. Metodi di comunicazione tra le functions: parametri e variabili globali. Variabili locali e controllo del metodo di allocazione. Cenni all’approccio object-oriented.
- Tecniche di base per l’allocazione dinamica della memoria: allocazione di vettori, cambio di dimensione dei vettori, liste, alberi.
- Tecniche algoritmiche di base: ordinamento, ricerca sequenziale e a chiave; metodo ricorsivo per la ricerca o l’elaborazione in un albero.
- Tecniche elementari di calcolo numerico: ricerca del minimo di una funzione, integrazione con il metodo di Simpson, soluzione numerica di sistemi di equazioni differenziali (metodo di Runge-Kutta). Uso della libreria MINUIT per la ricerca del minimo di funzioni a molti parametri. Applicazione al metodo dei minimi quadrati.

### **Esercitazioni di Laboratorio (Primo Semestre, 1 cr.)**

1. Presa di contatto con il sistema operativo e con l’interfaccia utente: semplici esercizi di manipolazione di files e directories, uso del text editor (Emacs), compilazione ed esecuzione di un programma elementare in C.
2. Realizzazione di semplici procedure di manipolazione numerica; uso di cicli e blocchi “if-then-else”; utilizzo elementare di un debugger simbolico (gdb).
3. Input output da terminale e da file; lettura e decodifica di files con chiavi alfanumeriche; utilizzo delle stringhe di caratteri.
4. Utilizzo di librerie grafiche (HPLOT) per la realizzazione di semplici grafici di punti sperimentali e di funzioni.
5. Allocazione dinamica di vettori; ri-allocazione.
6. Esercizi su strutture dati allocate dinamicamente: liste e/o alberi.
7. Esercizi di integrazione numerica di funzioni.
8. Minimizzazione di funzioni; uso di MINUIT.
9. Realizzazione di un programma che esegua un fit di una funzione parametrica arbitraria a un set di punti sperimentali letto da file ed elaborato.

### **Esercitazioni di Laboratorio (Secondo Semestre, 1 cr in laboratorio + 1 cr di presentazione delle esperienze in aula)**

1. Soluzione numerica di sistemi di equazioni differenziali lineari.
2. Realizzazione di istogrammi a partire da serie di misure con errore statistico.
3. Generazione di numeri casuali; simulazione di un processo con distribuzione binomiale.
4. Estrazione di numeri casuali secondo una distribuzione data: metodo di reiezione e metodo di inversione.
5. Rudimenti dell’utilizzo della porta parallela e seriale. Comunicazione con un multimetro digitale tramite la porta seriale.
6. Utilizzo del multimetro per la realizzazione di un campionatore. Misura automatica della curva di raffreddamento di un liquido.

7. Utilizzo di un DAC collegato alla porta parallela. Curva di calibrazione del DAC. Misura automatica della caratteristica I-V di un componente utilizzando un DAC ed un multimetro digitale.
8. Misure di tempo con il computer: polling ed interrupt. Misura automatica della frequenza di un oscillatore.