

**C.d.L. in MATEMATICA**  
**C.d.L. in MATEMATICA ed INFORMATICA**  
**Programmazione 1**

*Prova scritta del 28/7/2004*

1) (a) Si supponga di rappresentare un intervallo chiuso di interi come una struttura di nome `Intv` costituita da due campi interi, `inf` e `sup`, contenenti rispettivamente il limite inferiore e quello superiore dell'intervallo. Realizzare una funzione di nome `INTINT` che, presi come suoi parametri un array `A` di intervalli, la sua dimensione `n` ed un intervallo `INuovo`, calcola in `INuovo` l'intersezione tra tutti gli intervalli contenuti in `A`. La funzione restituisce `0` se l'intervallo risultante è vuoto, e `1` altrimenti. SUGG.: si dichiarare `INuovo` come parametro per riferimento.

(b) Descrivere la funzione anche tramite un diagramma di flusso.

2) Realizzare una funzione di nome `CARICA` che legge un file contenente coppie di numeri interi rappresentanti gli estremi di altrettanti intervalli chiusi e li memorizza in un array `M` di strutture di tipo `Intv` nel modo seguente: richiede all'utente il nome del file da leggere (max. 32 caratteri); se il file non esiste, concatena l'estensione `.dat` al nome del file fornito dall'utente e prova ad aprire il file col nuovo nome; se l'apertura ha successo, legge tutte le coppie di numeri dal file e le memorizza una alla volta negli elementi di `M`; la funzione restituisce come suo risultato il numero di elementi letti oppure `-1` nel caso in cui l'apertura del file non abbia avuto successo. Eventuali intervalli in cui l'estremo inferiore sia maggiore dell'estremo superiore non sono considerati corretti e non devono essere registrati nell'array `M`. Si supponga che `M` abbia dimensione massima `100` e nel caso si raggiunga tale dimensione la funzione termina immediatamente. SUGG.: si dichiarare `M` come parametro della funzione.

3) Scrivere un programma principale che: (1) presenta all'utente (su standard output) un menù di possibili operazioni su intervalli; (2) esegue l'operazione scelta; (3) ripete da (1) finché non viene scelta l'operazione di "smetti". Le possibili operazioni sono:

- a. legge una sequenza di intervalli da un file e li memorizza in un array `Dati` utilizzando la funzione `CARICA`; se il file specificato non esiste, richiede un nuovo nome di file;
- b. determina e stampa (su standard output) l'intersezione di tutti gli intervalli caricati in `Dati` oppure un opportuno messaggio se l'intersezione è vuota (si utilizzi obbligatoriamente la funzione `INTINT` dell'esercizio (1));
- c. richiede all'utente un intervallo e lo aggiunge, se possibile, a quelli già memorizzati in `Dati`;
- d. smetti.