

Prova scritta del 13/6/2008

*Strutturare adeguatamente i programmi ed evidenziarne la strutturazione mediante indentazione.
Inserire anche adeguati commenti*

1) (a) Realizzare una funzione booleana di nome `TERMINA` che, presi come suoi parametri due stringhe `A` e `B`, determina se `B` è un suffisso di `A` (cioè, se `B` coincide esattamente con gli ultimi `m` elementi di `A`; ad es., se `A` è `"prova.txt"`, la stringa `".txt"` è un suffisso di `A` mentre `".tx"` e `".tex"` non lo sono). N.B. Utilizzare soltanto stringhe tipo `C` (= array di caratteri ...).

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

2) Si consideri un file di testo di nome `"nomi_file.txt"` contenente sequenze di caratteri (qualsiasi) separate tra loro da esattamente un carattere `'#'`. Scrivere un programma che legge dal file `"nomi_file.txt"`, una alla volta, le stringhe in esso contenute, controlla per ciascuna stringa, tramite la funzione `TERMINA`, se la stringa termina o meno con la sequenza di caratteri `".doc"` e, in caso affermativo, memorizza la stringa letta in un array `A` (di dimensione massima `1000`). Al termine il programma provvede a stampare su standard output tutte le stringhe memorizzate in `A`, una stringa per ogni riga di stampa, in ordine inverso (cioè partendo dall'ultima inserita in `A`). N.B. Il carattere delimitatore `'#'` non deve essere inserito nella stringa letta. La lunghezza massima di ciascuna stringa è `32`: stringhe più lunghe vengono troncate ed il programma continua dalla stringa successiva, dando un'opportuna segnalazione all'utente.

3) Scrivere un programma che permetta di operare su una matrice `M` di dimensioni `10 x 20` i cui elementi sono strutture (`struct`) costituite da due campi, `num` di tipo reale e `c` di tipo booleano. Il campo `c` indica se il corrispondente campo `num` è inizializzato o no, e cioè se contiene un valore significativo (`c = true`) oppure no (`c = false`). Inizialmente tutti gli elementi della matrice sono non-inizializzati. Il programma propone all'utente un menù di possibili operazioni:

1. assegna il valore `x` all'elemento di `M` di indici `i` e `j` (l'elemento `Mi,j` deve essere non-inizializzato e diventa inizializzato al termine dell'operazione);
2. elimina l'elemento di indici `i` e `j` (l'elemento `Mi,j` deve essere inizializzato e diventa non-inizializzato al termine dell'operazione);
3. per ogni riga di `M`, calcola e stampa la media aritmetica dei campi `num` degli elementi di `M` inizializzati (e cioè in cui il campo `c` ha valore `true`);
4. smette.

Il programma presenta all'utente (su standard output) il menù delle possibili operazioni, esegue l'operazione scelta e quindi ripete dall'inizio finché non viene scelta l'operazione 4. `x`, `i` e `j` sono forniti dall'utente (su standard input) all'inizio dell'operazione scelta (si supponga, per semplicità, che i valori di `i` e `j` specificati siano sicuramente corretti). Se l'operazione non può essere eseguita a causa della inizializzazione/non-inizializzazione degli elementi di `M` selezionati viene dato un opportuno messaggio d'errore ed il programma continua ripresentando il menù delle operazioni.