

## Prova scritta del 25/2/2014

*Convenzioni (obbligatorie). Utilizzare i nomi indicati nel testo -  
“Indentare” in modo opportuno i programmi - Inserire adeguati commenti  
Utilizzare soltanto stringhe “tipo C”*

1) (a) Realizzare una funzione di nome `prodotto` che, presi come suoi parametri un array di interi `A`, un array di caratteri `B`, il numero `n` e `m` degli elementi in `A` e in `B`, e un array `R` di elementi di tipo `struct` con due campi, `num` e `car`, rispettivamente di tipo intero e carattere, calcola e memorizza in `R` il prodotto cartesiano di `A` e `B`. Si ricordi che il prodotto cartesiano di due insiemi `A` e `B` è l'insieme delle coppie ordinate  $(a, b)$  con  $a$  in `A` e  $b$  in `B` e cioè:  $\{(a, b) : a \in A \wedge b \in B\}$ . La funzione restituisce come suo risultato il numero di elementi memorizzati in `R`.

(b) Realizzare anche un programma principale di prova che dichiara un array `S1` con valore  $\{1, 5, 7\}$ , un array `S2` con valore  $\{'a', 'b'\}$ , calcola il prodotto cartesiano di `S1` e `S2` utilizzando la funzione `prodotto`, e quindi stampa il prodotto di `S1` e `S2` con (esattamente) il seguente formato di stampa:  $\{(1, a), (1, b), (5, a), (5, b), (7, a), (7, b)\}$ .

(c) Descrivere la funzione `prodotto` tramite diagramma di flusso.

2) Scrivere un programma principale che richiede all'utente il nome di due file contenenti rispettivamente dati di tipo intero e di tipo carattere, legge i due file in due array (ciascuno di dim. max *1000*), di nome `S1` e `S2`, e quindi, utilizzando la funzione `prodotto`, costruisce un nuovo array, di nome `S3`, contenente il prodotto cartesiano di `S1` e `S2`. Il programma quindi scrive su un nuovo file tutti i dati presenti in `S3`, separati da caratteri “spazio” o “a capo”. Il nome del nuovo file è ottenuto concatenando il prefisso “`prodotto_di_`” con i nomi dei due file di input separati da un carattere “`_`” (ad es., se i file di input si chiamano “`dati1`” e “`dati2`”, il file di output si chiamerà “`prodotto_di_dati1_dati2`”). N.B. Nella lettura del secondo file, i caratteri “spazio” e “a capo” devono essere ignorati. I nomi dei file di input non devono eccedere i *64* caratteri e possono contenere anche caratteri “spazio”. Si assuma che i dati memorizzati nei file di input siano corretti e che i file siano non vuoti. Nel caso in cui uno dei file di input non esista, il programma termina immediatamente.

3) Realizzare una funzione booleana di nome `uguali` che, presi come suoi parametri due liste concatenate `l1` e `l2` i cui elementi hanno campo informazione di tipo stringa (di lung. max *100*), confronta le due liste e se sono uguali (e cioè hanno gli stessi valori dei campi informazione, nello stesso ordine) restituisce `true`, altrimenti restituisce `false`. N.B. Le liste `l1` e `l2` (una o entrambe) possono essere vuote.