

Laboratorio di programmazione e Informatica 1 - A.A. 2016-2017 -

Prova di laboratorio: I appello - 15 giugno 2017

ISTRUZIONI: La prima riga di ogni programma C deve contenere il proprio nome e cognome. Tutti i programmi devono essere strutturati in funzioni e completi di commenti che spieghino il procedimento.

Creare una cartella < *CognomeNome* > e copiare all'interno i file sorgente dei programmi (.C o .cpp). La cartella andrà copiata nella pennetta del docente.

ATTENZIONE! Non saranno valutati programmi che non passano la fase di compilazione. Si consiglia pertanto di "mettere sotto commento" le parti di programma che danno errore in compilazione.

Tempo a disposizione: 2 ore e mezza.

ESERCIZI

Esercizio 1 (17 punti) *Cornici in matrici*

Definisco *bordo* di una matrice l'insieme delle posizioni della prima e ultima riga e della prima ed ultima colonna. Il bordo di una matrice viene anche chiamata *cornice 0*. In generale, una cornice di una matrice è l'insieme degli elementi equidistanti dai bordi. Nell'esempio è mostrata una matrice quadrata 5×5 , in cui compare 0 in corrispondenza di tutti gli elementi sulla cornice 0 (il bordo) , 1 sugli elementi sulla cornice 1 e così via.

```
0 0 0 0 0
0 1 1 1 0
0 1 2 1 0
0 1 1 1 0
0 0 0 0 0
```

- Scrivere una funzione C di prototipo: `int SommaCornice(int [][] A, int h)` che ricevendo in input una matrice *A* di interi di dimensione $n \times n$ ed un intero $h > 0$ calcola la somma degli elementi della cornice *h* e ne restituisce la somma.
- Scrivere un programma in C che:
 1. Genera una matrice 15x15 a valori interi random tra -20 e 30 tale che in ogni riga il primo e l'ultimo elemento siano uguali e gli elementi "interni" siano diversi dal primo e dall'ultimo.
 2. Stampa la matrice ben formattata sullo schermo.
 3. Utilizza la funzione `SommaCornice` per individuare se esiste una cornice la cui somma è maggiore della somma degli elementi del bordo (cioè della cornice 0).

Esercizio 2 (13 punti) *Cancellare sottostringhe*

Scrivere un programma in linguaggio C che legge dall'utente 2 stringhe di 0 e 1 di max 40 caratteri. Il programma dovrà calcolare se una delle due stringhe può essere sovrapposta sull'altra e in caso positivo la stringa sovrapposta dovrà essere cancellata dalla stringa più lunga.

Esempio: Se l'utente inserisce le stringhe:

```
1000100100100010000011111      e      100000111
```

Il programma dovrà scrivere:

La seconda stringa si sovrappone sulla prima nella posizione 15.

Cancellando la sovrapposizione, la nuova stringa e': 1000100100100011 .