

Laboratorio di programmazione e Informatica 1

- A.A. 2023-2024 -

Primo appello - 5 febbraio 2024

ISTRUZIONI:

- Gli esercizi vanno chiamati come `CognomeNomeEse1.c`, `CognomeNomeEse2.c`, `CognomeNomeEse3.c` e consegnati tramite la chiavetta Usb del docente. (Consegnare soltanto i file `.c`)
- La prima riga di ogni programma `C` deve contenere il proprio nome e cognome come commento.
- I programmi devono essere strutturati in funzioni e completi di commenti che spieghino il procedimento.
- Quando è richiesto lo studio della complessità dell'algoritmo implementato, questa va descritta alla fine del programma, sempre sotto commento.
- Ogni esercizio vale 10 punti. Bisogna fare ≥ 18 punti con almeno 5 punti su ogni esercizio.
- **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: 3 ore.

Esercizio 1

Data una matrice quadrata A di dimensione $m \times m$, per ogni indice $i = 0, 1, \dots, n$, definisco il corrispondente vettore *CROSS* tale che $cross[i]$ contiene la somma di tutti gli elementi della riga i e della colonna i . $\forall i = 0, \dots, n-1$ (l'elemento sulla diagonale lo sommo una sola volta!).

1. Scrivere una funzione `CalcolaCross` che, data la matrice A calcola il corrispondente vettore *CROSS* definito come sopra.
2. Scrivere una funzione:

```
int Min1e2(int C[], int *min1, int *x1, int *min2, int* x2)
```

che, dato un vettore calcola il valore più piccolo $min1$, il secondo valore più piccolo $min2$, e le rispettive posizioni $x1$, $x2$. I due valori primo e secondo minimo possono essere anche coincidenti ma ovviamente le posizioni saranno diverse.

3. Scrivere una funzione in C `void Genera(int a[DIM][DIM])` che genera una matrice di DIM righe e DIM colonne a valori interi random tra 21 e 71 tale che in ogni colonna non ci siano elementi consecutivi uguali.
4. Scrivere un programma in linguaggio C (funzione `main`) che:
 - Usando la funzione `Genera`, genera una matrice di dimensione 20x20 e stampa tale matrice sullo schermo.
 - Usando la funzione `CalcolaCross`, genera il vettore sopra definito e lo stampa sullo schermo.
 - Utilizza la funzione `Min1e2` sull vettore generato e stampa i risultati calcolati sullo schermo.

Tutte le stampe devono essere fatte dal `main`.

5. Discutere la complessità degli algoritmi implementati in funzione alla dimensione della matrice.

Esercizio 2

Date due stringhe u e v della stessa lunghezza, si dice che hanno un *mismatch in posizione i* se in quella posizione le stringhe contengono un simbolo diverso. Ad esempio le stringhe *pizza* e *pazza* hanno un mismatch in posizione 2 (iniziando a contare da 1), mentre le stringhe *pizza* e *pezzo* hanno due mismatch, rispettivamente in posizione 2 e 5. Le stringhe binarie 0011001101 e 0000001111 hanno 3 mismatch.

- Si scriva una funzione C che prende due stringhe della stessa lunghezza e restituisce il numero di mismatch tra queste.
- Si scriva un programma C (funzione main) che richiede all'utente di inserire due stringhe. Se le stringhe sono della stessa lunghezza, usa la funzione precedente per calcolare il numero di mismatch e lo scrive sullo schermo, se hanno lunghezza diversa, lo segnala e richiede altre due stringhe.

Non è consentito utilizzare funzioni predefinite sulle stringhe.

Esercizio 3

Sia data una lista di interi.

1. Si implementi una funzione in C `SdoppiaLista`, che prende in input una lista A (data come un puntatore al suo primo nodo) di valori interi $a_1, a_2, \dots, a_n$ e crea una nuova lista B estraendo da A tutti gli elementi che sono strettamente maggiori di a_1 . Alla fine quindi la lista A conterrà gli elementi che sono minori o uguali ad a_1 . Gli elementi in B devono trovarsi nello stesso ordine in cui erano in A . La funzione deve essere di complessità lineare rispetto alla lunghezza della lista in input (quindi NON deve richiamare altre funzioni lineari di inserimento o cancellazione al suo interno).
2. Si inserisca tale funzione nel codice fornito `listeEsame1app.c` rinominando il file come `CognomeNomeEse3.c`