

Preparazione Midterm

2025/11/05 – Preparazione Parziale

L. Magri, L. Alessandrini

Credits: A. Montenegro per gli esercizi

Tutorato

Cos'è, come funziona

Tutorato

Data: 7 novembre dalle ore 14:00 alle ore 17:00

Come Funziona

- Online nella stanza dell' Ing. Peretti
(<https://politecnicomilano.webex.com/meet/edoardo.peretti>)
- Inviare le domande in anticipo prima di ogni sessione nel form alla pagina <https://forms.office.com/e/hkn2ap8100>

Per link e form aggiornati consultare la pagina

- <https://alessandriniluca.github.io/tutorato.html>

1) Nativi americani

Si consideri la seguente definizione di strutture per rappresentare un insieme di popoli nativi americani, ciascuno caratterizzato da un nome, un numero di abitanti e la lingua principale parlata:

```
typedef struct {  
    char nome[N];          // Nome della lingua parlata  
    char vitalita[N];      // Es. "Viva", "In pericolo", "Estinta" https://ich.unesco.org/doc/src/00120-EN.pdf  
} lingua_t;  
  
typedef struct{  
    char nome[N]; // nome della popolazione  
    int pop; // numero di abitanti  
    lingua_t lingua; // lingua parlata  
} popolo_t;  
  
typedef struct {  
    popolo_t v[DIM]; // Vettore dei popoli  
    int n;           // Numero effettivo di popoli presenti  
} popolazione_t;
```

1) Nativi americani: analisi demografica

Scrivere una funzione `analizzaPopoli` che prende in ingresso una popolazione, ed ulteriori variabili aggiuntive se necessarie, e riporta al chiamante:

- Il nome del popolo con più abitanti tra quelli della popolazione
- Il nome del popolo con il numero di abitanti più vicino alla media degli abitanti dei popoli della popolazione in esame

Hint: per individuare il popolo più vicino alla popolazione media, confrontare la differenza assoluta tra il numero di abitanti di ciascun popolo e il valore medio.

1) Nativi americani: filtraggio per vitalità linguistica

Scrivere una funzione `filtraPopoliLingua` che prende in ingresso una popolazione ed eventuali parametri aggiuntivi ed elimina tutti i popoli la cui lingua **non** corrisponde al livello di vitalità indicato da `vitalitaRichiesta`.

La funzione restituisce il numero di popoli rimasti dopo il filtraggio.

1) Nativi americani: main

1. Definire e inizializzare una struttura popolazione_t contenente almeno cinque popoli nativi americani (es. Navajo, Apache, Sioux, Cherokee, Hopi), specificando per ciascuno:

- il numero di abitanti,
- il nome della lingua parlata,
- e il relativo livello di vitalità linguistica.

2. Invocare la funzione analizzaPopoli e stampare:

- il nome del popolo con più abitanti;
- il nome del popolo con popolazione più vicina alla media.

3. Richiedere all'utente di inserire un livello di vitalità linguistica (ad esempio "Viva", "In pericolo", "Estinta").

4. Invocare filtraPopoliLingua per eliminare i popoli la cui lingua non corrisponde al livello specificato.

5. Stampare a video i popoli rimasti dopo il filtraggio, indicando per ciascuno il nome, la lingua e il livello

2

TDE Settembre 2024

Testo

Si scriva una funzione **conta** che riceve in ingresso una stringa **S** contenente più parole separate da virgole e contenenti solamente lettere minuscole (oltre ad eventuali parametri aggiuntivi utili per lo sviluppo della funzione).

La funzione **conta** deve restituire al chiamante la lettera dell'alfabeto che compare più frequentemente tra le iniziali delle parole (si supponga per semplicità sia unica).

Si scriva quindi una seconda funzione **elimina** che riceve in ingresso la stringa **S** (contenente più parole e solo caratteri minuscoli), un carattere **C** (oltre ad eventuali parametri aggiuntivi ritenuti utili per lo sviluppo della funzione), e riporta al chiamante una stringa **T** contenente le sole parole di **S** che iniziano per **C**, separate da virgole. Nel main si stampi il vettore restituito dalla funzione **elimina**.

3

TDE Gennaio 2023

Testo

Si una funzione riordina che prende in ingresso due vettori **u** e **v** (più eventuali variabili necessarie) e che riordina gli elementi del primo vettore passato in input (**u** nel nostro caso) nel seguente modo:

- tutti gli elementi di **u** che non compaiono in **v** saranno nelle prime posizioni di **u** dopo l'invocazione
- tutti gli elementi di **u** che compaiono in **v** saranno nelle ultime posizioni di **u** dopo l'invocazione

Si faccia riferimento all'esempio sotto per eventuali dubbi su come modificare **u**.
Si invochi quindi la funzione relativamente al seguente codice e si stampi il contenuto del vettore **u** a schermo.

N.B. non è possibile stampare all'interno della funzione riordina

$u = [4, 1, 3, 7, 0]$ e $v = [1, 6, 8, 4] \rightarrow u = [3, 7, 0, 4, 1]$

Thank You!

Mail: luca1.alessandrini@polimi.it

Website: <https://alessandriniluca.github.io/>