

Compito di Programmazione I - Bioinformatica

24 febbraio 2023 (tempo disponibile: 2 ore)

Esercizio 1 (31 punti) (si consegni `marziano.c`)

Un array di caratteri è detto *alfabetico* se i suoi caratteri sono tutti lettere minuscole o maiuscole dell'alfabeto inglese (eventualmente ripetute). Per esempio, l'array `['i','P','b','J','j','E']` è alfabetico mentre l'array `['j','(','b','j','A','e']` non è alfabetico.

Un array di caratteri è detto *marziano* se è alfabetico e, inoltre, è composto da una prima parte che contiene consonanti, in ordine alfabetico inverso, e da una seconda parte che contiene vocali (italiane), sempre in ordine alfabetico inverso (si ricordi che nell'ordine alfabetico le maiuscole vengono prima delle minuscole). Per esempio, l'array

$$\left[\underbrace{'j','b','P','J'}_{\text{consonanti in ordine alfabetico inverso}}, \underbrace{'i','E'}_{\text{vocali in ordine alfabetico inverso}} \right]$$

è marziano. Invece l'array `['i','P','b','J','j','E']` non è marziano, perché la vocale `i` precede la consonante `J`. Neanche l'array `['b','j','P','J','i','E']` è marziano, perché le consonanti non sono in ordine alfabetico inverso. E neanche l'array `['j','b','P','J','E','i']` è marziano, perché le vocali non sono in ordine alfabetico inverso.

Si completino le cinque funzioni del programma `marziano.c`:

```
// aggiungete gli #include necessari
#include "marziano.h"

// inizializza l'array indicato, lungo length,
// in modo che diventi un array alfabetico casuale
// (caratteri alfabetici minuscoli o maiuscoli, eventualmente ripetuti)
void init_random(char arr[], int length) {
    // completare
}

// stampa l'array indicato, su una riga, senza spazi fra i caratteri,
// andando a capo alla fine
void print(char arr[], int length) {
    // completare
}

// riceve un array alfabetico arr e ne sposta i caratteri in modo che
// arr diventi marziano (prima le consonanti, in ordine alfabetico inverso,
// poi le vocali, in ordine alfabetico inverso)
void ordina_marziano(char arr[], int length) {
    // completare
}

// stampa una lista di caratteri, senza spazio fra i caratteri,
// andando a capo alla fine
```

```

void print_list(struct element_t *l) {
    // completare
}

// riceve un array marziano lungo length e restituisce una lista
// che contiene solo le vocali dell'array e senza ripetizioni:
// una vocale viene inserita nella lista solo la prima volta che
// compare, mentre la seconda volta non viene inserita nella lista
struct element_t *vocali_non_ripetute(char arr[], int length) {
    // completare
}

```

I file marziano.h e main.c sono già scritti e completi, non vanno modificati e non vanno consegnati. Se servisse, si possono aggiungere funzioni ausiliarie dentro marziano.c.

Se tutto è corretto, un esempio di esecuzione del main.c potrebbe essere:

```

Inserisci la lunghezza dell'array, non negativa: 30
                Array alfabetico casuale: lvPqQgG0IadSbSeUafTKuRtbIafQYI
                Array trasformato in marziano: vtqlgffdbbYTSSRQQPKGueaaaUOIII
Lista derivata dall'array eliminando le consonanti e le vocali ripetute: ueaUOI

```