

Compito di Programmazione - Bioinformatica

5 febbraio 2025 (tempo disponibile: 2 ore)

(si consegnino list.c, list.h e main.c)

Esercizio 1 (20 punti)

Si completi il seguente file list.c. Si noti che la definizione della struttura List è già fornita dentro list.h.

```
// aggiungete gli #include necessari

struct List *create(int head, struct List *tail) {
    // ritorna una nuova lista con la testa e la coda indicate
}

void print(struct List *this) {
    // stampa gli elementi di this, dal primo all'ultimo, su una
    // riga, separati da spazi, andando a capo alla fine;
    // si noti che this potrebbe essere NULL
}

struct List *from(int arr[], int length) {
    // ritorna una lista contenente esattamente gli elementi
    // di arr, lungo length, nell'ordine in cui sono presenti
    // nell'array; se length fosse 0, dovra' ritornare NULL
}

struct List *filter(struct List *this, int threshold) {
    // ritorna una lista contenente i soli elementi di this
    // che sono maggiori o uguali a threshold;
    // si noti che this puo' essere NULL e che il risultato
    // sara' NULL se non ci fossero elementi maggiori o uguali
    // a threshold

    // QUESTA FUNZIONE DEVE ESSERE RICORSIVA
}

struct List *duplicate(struct List *this) {
    // ritorna una lista identica a this ma con gli elementi
    // duplicati; per esempio, se this fosse la lista 1 2 3, allora
    // questa funzione deve ritornare la lista 1 1 2 2 3 3;
    // si noti che this puo' essere NULL e che il risultato sara'
    // NULL se this fosse vuota

    // QUESTA FUNZIONE DEVE ESSERE RICORSIVA
}
```

Esercizio 2 (2 punti)

Si completi il file `list.h` con le dichiarazioni delle funzioni di `list.c`.

Esercizio 3 (9 punti)

Si scriva un file `main.c` che dichiara una funzione `main` che, usando le funzioni di `list.c`:

1. crea una lista `l` contenente esattamente gli elementi 7, 17, -11, 13, 17;
2. stampa `l`;
3. filtra la lista `l` ai soli elementi maggiori o uguali a 10, chiamando `f` il risultato;
4. stampa `f`;
5. stampa `l`;
6. filtra la lista `l` ai soli elementi maggiori o uguali a 40, chiamando `g` il risultato;
7. stampa `g`;
8. crea una lista ottenuta duplicando gli elementi di `l`, chiamando `d` il risultato;
9. stampa `d`;
10. stampa `l`.

Se tutto è corretto, l'esecuzione della compilazione congiunta di `list.c` e di `main.c` dovrà stampare:

```
l = 7 17 -11 13 17
f = 17 13 17
l = 7 17 -11 13 17
g =
d = 7 7 17 17 -11 -11 13 13 17 17
l = 7 17 -11 13 17
```

Se servisse, negli esercizi si possono aggiungere funzioni ausiliarie.