

Parziale di Programmazione I - BioInformatica

30 gennaio 2019 (tempo disponibile: 2 ore)

Esercizio 1 (16 punti)

Si scriva un programma `swap.c` che implementa le seguenti tre funzioni su array di caratteri:

```
// inizializza arr, lungo length, con caratteri alfabetici
// inglesi minuscoli scelti a caso
void init_random(char arr[], int length);

// stampa su un'unica riga i caratteri nell'array arr, lungo length, poi va a capo
void print(char arr[], int length);

// scambia ogni elemento in posizione pari di arr, lungo length,
// con quello in posizione dispari che lo segue.
// Se length fosse dispari, l'ultimo elemento restera' nella sua posizione
void swap(char arr[], int length);
```

Per esempio, se `arr` fosse `{'f','m','d','j','a'}`, chiamando `swap(arr, 5)` l'array `arr` dovrà diventare `{'m','f','j','d','a'}`.

Si scriva quindi un file di header `swap.h` che dichiara le precedenti tre funzioni.

Esercizio 2 (16 punti)

Si scriva un programma `main_swap.c` che include le funzioni dell'Esercizio 1 tramite il file di header `swap.h`. Il programma `main_swap.c` deve contenere una funzione iniziale `main` che esegue le seguenti operazioni:

1. legge da tastiera la lunghezza `length` di un array, richiedendola ad oltranza se fosse inserita negativa;
2. crea un array `elements` di `length` caratteri;
3. usa la funzione `init_random` per inizializzare casualmente gli elementi di `elements`;
4. usa la funzione `print` per stampare `elements`;
5. usa la funzione `swap` per scambiare ogni elemento di `elements` in posizione pari con quello in posizione dispari che lo segue;
6. usa la funzione `print` per stampare nuovamente `elements`.