

Parziale di Programmazione I - Bioinformatica

1 febbraio 2021, turno delle 16:00 (tempo disponibile: 2 ore)

Esercizio 1 (20 punti) (si consegnino `harshad.c` e `harshad.h`)

Si scriva un programma `harshad.c` che implementa le seguenti funzioni:

```
// inizializza arr, lungo length, con numeri interi lunghi casuali tra 0 a 999,  
// usando srand() e rand(), facendo in modo che alla fine  
// non ci siano elementi consecutivi uguali nell'array  
void init_random(long arr[], int length);  
  
// stampa su un'unica riga il contenuto dell'array arr, lungo length, poi va a capo  
void print(long arr[], int length);  
  
// determina se n e' un numero Harshad, cioe' e' positivo e divisibile  
// per la somma delle proprie cifre. Per esempio, 1729 e' Harshad  
// poiche' 1+7+2+9 fa 19 e 1729 e' divisibile per 19  
int is_harshad(long n);  
  
// modifica l'array, lungo length, in modo da riempirlo con i primi  
// length numeri Harshad  
void fill_with_harshad(long arr[], int length);
```

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

Esercizio 2 (12 punti) (si consegnino `main.c`)

Si scriva un programma `main.c` che include le funzioni dell'Esercizio 1 tramite il file `harshad.h`. Il programma `main.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` interi lunghi;
3. chiama la funzione `init_random` per inizializzare `elements` in modo casuale;
4. chiama la funzione `print` per stampare `elements`;
5. chiama la funzione `fill_with_harshad` con l'array `elements` come parametro;
6. chiama la funzione `print` per stampare `elements`.