

Esame Completo di Programmazione I - Bioinformatica

1 febbraio 2021 (tempo disponibile: 2 ore)

Esercizio 1 (18 punti) (si consegna penta.c e penta.h)

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

```
// inizializza arr, lungo length, con numeri interi casuali tra 0 a 999,  
// usando srand() e rand()  
void init_random(int arr[], int length);  
  
// stampa su un'unica riga il contenuto dell'array arr, lungo length, poi va a capo  
void print(int arr[], int length);  
  
// determina se il numero n non negativo e' pentafratto,  
// cioe' se ha almeno 5 divisori interi positivi  
int is_pentafract(int n);  
  
// modifica l'array, lungo length, in modo da spostare al suo inizio i suoi elementi  
// pentafratti e alla sua fine i suoi elementi non pentafratti  
void pentafract_first(int arr[], int length);
```

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

Esercizio 2 (14 punti) (si consegna main.c)

Si scriva un programma `main.c` che include le funzioni dell'Esercizio 1 tramite il file `penta.h`. Il programma `main.c` deve definire una struttura

```
struct list {  
    int x;  
    struct list *tail;  
};
```

e deve implementare due funzioni:

```
// alloca una struct list contenente x e tail e ne restituisce il puntatore in memoria  
struct list *construct(int x, struct list *tail) { ... }  
  
// determina se la lista l contiene almeno un elemento x pentafratto  
// (si chiami la funzione opportuna dell'Esercizio 1)  
int at_least_one(struct list *l) { ... }
```

Inoltre `main.c` deve contenere una funzione iniziale `main` che esegue le seguenti operazioni:

1. crea una lista $5 \rightarrow 12 \rightarrow 6 \rightarrow 15$;
2. chiama `at_least_one` passando tale lista come parametro;
3. stampa il valore ritornato dalla chiamata di funzione del punto precedente.