

Parziale di Programmazione I - Bioinformatica

31 gennaio 2022, turno delle 9:00 (tempo disponibile: 2 ore)

Esercizio 1 (18 punti) (si consegnino `second.c` e `second.h`)

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

```
// inizializza l'array, lungo length, con numeri casuali tra 0 e 9 inclusi
void init(int array[], int length) { ... }

// stampa l'array, lungo length, su una riga e poi va a capo
void print(int array[], int length) { ... }

// restituisce l'elemento dell'array, lungo length, il cui valore
// e' il secondo piu' grande fra quelli contenuti nell'array;
// se non esistesse, restituisce -1.
// L'array non deve venire modificato
int second(int array[], int length) { ... }
```

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

Per esempio, il seguente file `main.c` (già fornito e da non modificare):

```
#include <stdio.h>
#include "second.h"

int main(void) {
    int array1[8];
    init(array1, 8);
    printf("array1: ");
    print(array1, 8);
    printf("Secondo piu' grande elemento di array1: %i\n", second(array1, 8));
    int array2[] = { 8, 8, 4, 8 };
    printf("array2: ");
    print(array2, 4);
    printf("Secondo piu' grande elemento di array2: %i\n", second(array2, 4));
    int array3[] = { 8, 8, 8 };
    printf("array3: ");
    print(array3, 3);
    printf("Secondo piu' grande elemento di array3: %i\n", second(array3, 4));
    int array4[] = { };
    printf("array4: ");
    print(array4, 0);
    printf("Secondo piu' grande elemento di array4: %i\n", second(array4, 0));
    return 0;
}
```

stampa qualcosa del tipo:

```
array1: 3 1 7 2 6 1 7 5
Secondo piu' grande elemento di array1: 6
array2: 8 8 4 8
Secondo piu' grande elemento di array2: 4
array3: 8 8 8
Secondo piu' grande elemento di array3: -1
array4:
Secondo piu' grande elemento di array4: -1
```

Esercizio 2 (13 punti) (si consegna bar.c)

Si completi il seguente programma `bar.c` in modo che la funzione `bar` stampi una riga di n asterischi seguiti da $2n$ chioccioline. **La funzione `bar` deve essere ricorsiva:**

```
#include <stdio.h>

// funzione ricorsiva che stampa n asterischi seguiti da 2*n chioccioline
// sia assuma che n sia non negativo
void bar(int n) {
    // DA COMPLETARE, DEVE ESSERE RICORSIVA
}

int main(void) {
    // DA COMPLETARE:
    // 1) legge n >= 0 da tastiera, richiedendolo a oltranza se negativo
    // 2) chiama bar(n)
    // 3) va a capo
    return 0;
}
```

Per esempio, un utilizzo del programma potrebbe essere il seguente:

```
Inserisci n >= 0: 5
*****@@@@@@@@@@@@
```