

# Compito di Programmazione I - BioInformatica

21 Giugno 2018 (tempo disponibile: 2 ore)

## Esercizio 1 (6 punti)

Che numero stampa l'esecuzione del seguente programma?

```
#include <stdio.h>

int main(void) {
    int original[] = { -8, 6, 7, -5, 8, -1, 0, 4, 3, 1 };
    int counter = 0, i;
    for (i = 0; i < 10; i++) {
        int x = original[i];
        if (x < 0)
            x = -x;

        counter += x % 2;
    }
    printf("%d\n", counter); // cosa stampa?
    return 0;
}
```

## Esercizio 2 (10 punti)

Si scriva una funzione `max_array` che riceve due array di interi `original` e `result` e la loro lunghezza `length` (uguale per entrambi). La funzione deve modificare `result` in modo che ogni suo elemento di indice  $i$  diventi il massimo degli elementi di `original` con indice compreso fra 0 e  $i$  inclusi. La funzione non deve modificare gli elementi di `original`. Se `length` è minore o uguale a 0, la funzione non deve fare nulla. Per esempio, se `original` è l'array  $\{-2, 6, 7, 5, 8, -3, 0, -4, 0, -1\}$  e quindi `length` è 10, alla fine della funzione `result` deve contenere  $\{-2, 6, 7, 7, 8, 8, 8, 8, 8, 8\}$ .

## Esercizio 3 (12 punti)

Si definisca una struttura `studente` che implementa uno studente. Si scrivano i file `studente.h` e `studente.c` implementando le funzioni:

- `struct studente *construct_studente(char *nome)` che restituisce un nuovo studente con il nome indicato;
- `void destruct_studente(struct studente *this)` che dealloca lo studente `this`;
- `void fa_esame(struct studente *this, int voto)`, che registra il voto indicato per lo studente `this`, se il voto è fra 18 e 30 inclusi, e non fa nulla altrimenti; uno studente può fare al più 20 esami: oltre tale soglia, questa funzione non registra più ulteriori esami;
- `float media(struct studente *this)`, che restituisce la media dei voti degli esami sostenuti dallo studente `this`; se lo studente non ha ancora fatto esami, restituisce 0.0;
- `char *toString(struct studente *this)`, che restituisce una nuova stringa fatta dal nome dello studente `this` seguito dalla media degli esami sostenuti da `this`.

Se tutto è corretto, l'esecuzione del seguente programma:

```
#include <stdlib.h>
#include <stdio.h>
#include "studente.h"

int main(void) {
    struct studente *s1 = construct_studente("Giacomo");
    struct studente *s2 = construct_studente("Elisa");
    char *s;
    fa_esame(s1, 18);
    fa_esame(s1, 15); // non viene registrato
    fa_esame(s2, 30); fa_esame(s1, 25); fa_esame(s2, 22);
    fa_esame(s2, 29); fa_esame(s2, 27);
    printf("%s\n", s = toString(s1));
    free(s);
    printf("%s\n", s = toString(s2));
    free(s);
    destruct_studente(s1); destruct_studente(s2);
    return 0;
}
```

deve stampare:

```
Giacomo  21.50
Elisa    27.00
```

## Esercizio 4 (4 punti)

Cosa stampa il seguente programma C?

```
#include <stdio.h>

struct date {
    int day;
    int month;
    int year;
}

void changeDay(struct date d) {
    d.day = 12;
}

int main(void) {
    struct date today;
    today.day = 21;
    today.month = 6;
    today.year = 2018;
    changeDay(today);
    printf("La data modificata e': %i/%2i/%i", today.day, today.month, today.year);
    return 0;
}
```