

Parziale di Programmazione I - Bioinformatica

31 gennaio 2022, turno delle 15:00 (tempo disponibile: 2 ore)

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

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

```
// inizializza l'array, lungo length, con numeri casuali,  
// in modo che il primo elemento sia tra 0 e 10 inclusi  
// e che ogni altro elemento sia maggiore o uguale del precedente  
// e minore o uguale del precedente+10  
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) { ... }  
  
// determina se gli elementi di array sono tutti distinti,  
// supponendo che l'array, lungo length, sia ordinato non decrescente;  
// l'array non deve venire modificato  
int distinct(int array[], int length) {  
    // sia da ora per scontato che l'array sia ordinato non decrescente  
    ...  
}
```

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

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

```
#include <stdio.h>  
#include "distinct.h"  
  
int main(void) {  
    int array1[8];  
    init(array1, 8);  
    printf("array1: ");  
    print(array1, 8);  
    printf("Tutti distinti in array1? %i\n", distinct(array1, 8));  
    int array2[] = { 4, 5, 8, 8 };  
    printf("array2: ");  
    print(array2, 4);  
    printf("Tutti distinti in array2? %i\n", distinct(array2, 4));  
    int array3[] = { 9, 12, 13, 19 };  
    printf("array3: ");  
    print(array3, 4);  
    printf("Tutti distinti in array3? %i\n", distinct(array3, 4));  
    int array4[] = { };  
    printf("array4: ");
```

```

print(array4, 0);
printf("Tutti distinti in array4? %i\n", distinct(array4, 0));
return 0;
}

```

stampa qualcosa del tipo:

```

array1: 0 7 7 17 21 29 29 38
Tutti distinti in array1? 0
array2: 4 5 8 8
Tutti distinti in array2? 0
array3: 9 12 13 19
Tutti distinti in array3? 1
array4:
Tutti distinti in array4? 1

```

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

Si completi il seguente programma `sum.c` in modo che la funzione `sum` calcoli la sommatoria

$$0^2 + 1^2 + 2^2 + 3^2 + \dots + n^2.$$

La funzione `sum` deve essere ricorsiva:

```

#include <stdio.h>

long sum(int n) {
    // si dia per scontato che n >= 0
    // DA COMPLETARE: QUESTA FUNZIONE DEVE ESSERE RICORSIVA
}

int main(void) {
    // DA COMPLETARE:
    // 1) legge un intero n >= 0, se non lo fosse lo richiede ad oltranza
    // 2) chiama sum(n) e ne stampa il risultato
    return 0;
}

```

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

```

Inserisci n >= 0: 10
sommatoria = 385

```