



## Programmazione. Esame del 2025-01-20

Punti: 8/10

✓ **Esatto** 1/1 Punti

1

Qual è la complessità spaziale (quantità di memoria supplementare) della funzione `merge` utilizzata dall'algoritmo `merge-sort` ?

- ☒ lineare
- ☐ poli-logaritmica
- ☐ costante
- ☐ quadratica



✓ **Esatto** 1/1 Punti

2



☐ quadratica

✓ **Esatto** 1/1 Punti

2

Sia `a` una lista contenente `bool`, `int` e `str`, cosa restituisce `a_func(a)` ?

```
def a_func(a):  
    d = {}  
    for x in a:  
        if x in d:  
            d[x] += 1  
        else:  
            d[x] = 1  
    return max(a, key=lambda x: d[x])
```



☒ l'elemento che compare più frequentemente in `a` ✓

☐ la stringa più lunga in `a`

☐ il numero di volte in cui compare l'elemento più frequente in `a`

☐ il massimo in `a`

☐ il tipo che compare più frequentemente in `a`

✗ **Non corretto** 0/1 Punti

3



✗ **Non corretto** 0/1 Punti

3

Sia  $N$  una variabile `int` inizializzata con un valore positivo, qual è il costo computazionale, in termini di tempo di calcolo, del seguente frammento di codice C?

```
int size_a = N;
float *a = malloc(sizeof(float)*size_a);

for (int i = 0; i < 2*N; i++){
    if (i == size_a){
        float *t = a;
        a = malloc(sizeof(float)*(i+1));
        size_a = i+1;
        for(int j = 0; j < size_a; j++)
            a[j] = t[j];
        free(t);
    }
    a[i] = i;
}
```

- ☐ Il tempo di calcolo non dipende da  $N$
- ☐ Lineare in  $N$  nel caso peggiore, costante nel caso migliore.
- ☒ Lineare in  $N$  nel caso medio, quadratico in  $N$  nel caso peggiore
- ☐ Sempre quadratico in  $N$ . ✓
- ☐ Sempre lineare in  $N$

✓ **Esatto** 1/1 Punti



☐ Sempre quadratico in **N**.



☐ Sempre lineare in **N**

✓ **Esatto** 1/1 Punti

4

Qual è la complessità temporale (tempo di calcolo) dell'algoritmo `merge` nel caso in cui le sequenze da fondere abbiano dimensione `n` e `m`?

☐  $O(\min(n,m))$

☒  $O(\max(n,m))$



☐  $O(n*m)$

☐  $O(m \log n)$

☐  $O(n \log m)$

✗ **Non corretto** 0/1 Punti

5

Si consideri il seguente frammento di codice C, quale tra le seguenti affermazioni è vera?

```
int *f( int *k ){  
    *k = 3*k;
```



✗ **Non corretto** 0/1 Punti

5

Si consideri il seguente frammento di codice C, quale tra le seguenti affermazioni è vera?

```
int *f( int *k ){
    *k = 3*k;
    return k;
}

void main(){
    int m,n;

    /*
     * ...
     * Operazioni in cui viene assegnato un valore a n
     * ...
     * */

    m = *f(&n);
}
```

- ☐ **n e m** sono uguali
- ☐ **m** è 3 volte **n**
- ☐ **n** è 3 volte **m**
- ☐ **\*n** è 3 volte **\*m**
- ☒ **n** ed **m** puntano allo stesso indirizzo
- ☐ **m** è un array



✓ Esatto 1/1 Punti

6

In quale caso l'esecuzione di `myfunc(b)` genera un errore?

```
def myfunc( a ):
    n = len(a)
    if n < 2:
        return a
    return myfunc(a[:n//2]+ a[n//2:])
```

- ☐ mai
- ☐ se **b** è una qualsiasi stringa
- ☐ se il primo elemento di **b** è uno zero
- ☒ se **b** è una lista con almeno due elementi ✓
- ☐ se **b** è una tupla vuota

✓ Esatto 1/1 Punti

7

Sia `L` una lista concatenata di `n` interi implementata nel linguaggio `C`. Si consideri l'operazione **reset** che prende in input l'indirizzo del primo elemento di `L` ed un intero `k` e azzerà il valore dell'elemento in posizione `k` della lista. Qual è il costo computazionale in termini di tempo di calcolo della migliore implementazione di **reset**?





☐ se **b** è una tupla vuota

✓ **Esatto** 1/1 Punti

7

Sia **L** una lista concatenata di **n** interi implementata nel linguaggio **C**. Si consideri l'operazione **reset** che prende in input l'indirizzo del primo elemento di **L** ed un intero **k** e azzerà il valore dell'elemento in posizione **k** della lista. Qual è il costo computazionale in termini di tempo di calcolo della migliore implementazione di **reset**?

- ☒ lineare in **k** ✓
- ☐ costante
- ☐ logaritmico in **k**
- ☐ quadratico in **n**
- ☐ lineare in **n**

✓ **Esatto** 1/1 Punti

8

**a** è una lista Python contenente  $\sqrt{n}$  interi ordinati in modo crescente seguiti da altri  $\sqrt{n}$  interi non ordinati. Qual è il costo del miglior algoritmo che ordini la lista **a**?



☐ lineare in  $n$

✓ **Esatto** 1/1 Punti

8

`a` è una lista Python contenente  $\sqrt{n}$  interi ordinati in modo crescente seguiti da altri  $\sqrt{n}$  interi non ordinati. Qual è il costo del miglior algoritmo che ordini la lista `a` ?

☐  $O(n)$

☐  $O(n^2)$

☐  $O(n \log n)$

☐  $O(\sqrt{n})$

☒  $O(\sqrt{n} \log n)$



✓ **Esatto** 1/1 Punti

9

Le funzioni `intersect0` e `intersect1` calcolano lo stesso valore, dire quale delle seguenti affermazioni è falsa.

```
def intersect0( a , b):  
    for x in a:  
        if x in b:
```





✓ Esatto 1/1 Punt

9

Le funzioni `intersect0` e `intersect1` calcolano lo stesso valore, dire quale delle seguenti affermazioni è falsa.

```
def intersect0( a , b):  
    for x in a:  
        if x in b:  
            return True  
    return False  
  
def intersect1( a, b ):  
    A, B, c = {}, {}, 0  
    for x in a:  
        A[x] = None  
    for x in b:  
        B[x] = None  
  
    for x in A.keys():  
        if x in B:  
            return True  
  
    return False
```

- ☐ La memoria supplementare di **intersert0** è costante
- ☐ Nel caso medio, il tempo di calcolo di **intersert1** è lineare
- ☐ Nel caso peggiore, il tempo di calcolo di **intersert0** è quadratico
- ☒ Nel caso medio, il tempo di calcolo di **intersert0** è lineare ✓
- ☐ La memoria supplementare di **intersert1** è lineare

✓ **Esatto** 1/1 Punti

10

Sia `a` una lista Python di lunghezza `n > 123`, dire quale delle seguenti affermazioni è falsa dopo aver eseguito il seguente frammento di codice.

```
b = a
b.append(n)
a = b + [ 2*n ]
```

☐ `max(a) >= max(b)`☒ `a[-1] == b[-1]` ✓☐ `min(a) == min(b)`☐ `len(a) != len(b)`☐ `len(b) < max(a)`

Questo contenuto è creato dal proprietario del modulo. I dati inoltrati verranno inviati al proprietario del modulo. Microsoft non è responsabile per la privacy o le procedure di sicurezza dei propri clienti, incluse quelle del proprietario di questo modulo. Non fornire mai la password.

**Microsoft Forms** | Indagini, quiz e sondaggi alimentati dall'intelligenza artificiale [Crea un modulo personalizzato](#)

[Privacy e cookie](#) | [Condizioni per l'utilizzo](#) | [Accessibilità](#)