

# Es queue

lunedì 6 giugno 2022 17:24

Copiare da Q1 a Q2

Copia(Q1, Q2):

```
While not QueueEmpty(Q1):  
    R = dequeue(Q1)  
    Enqueue(Q2, r)
```

Copiare Q1 a Q2 senza distruggere Q1

Copia(Q1, Q2):

```
# Valore di stallo  
Enqueue(Q1, -1)
```

```
Do {  
    R = dequeue(q1)  
    If r != -1  
        Enqueue(q2, r)  
        Enqueue(q1, r)  
}while r != -1
```

Scrivere algoritmo che,

Ricevuta coda Q che potrebbe avere elementi ripetuti

E uno stack S con elementi tutti diversi

Elimini tutti gli elementi che appaiono anche nello stack

Elimina(Q, S):

```
# Fino a che non è vuoto  
While not stackEmpty(S) and not queueEmpty(Q):  
    Begin  
        # Prendiamo il valore dallo stack  
        R = pop(S)  
        # Fino a che non è vuota  
        While not queueEmpty(Q):  
            # Prendi il valore  
            R2 = dequeue(Q)  
            # Memorizzala solo se è diverso da R  
            If R2 != R:  
                Enqueue(Q2, R2)  
        # Ripristiniamo stack in una stack di appoggio  
        Push(S2, r)  
        # Ripristiniamo queue  
        While not queueEmpty(q2):  
            R3 = dequeue(q2)  
            Enqueue(q, r3)  
    End
```

N

$$\sum_{i=1}^n tw2$$

$$\sum_{i=1}^n tw3$$

|Q| = m

|S| = n

Migliore:

- La coda contiene 1 elemento ripetuto m volte ed è l'elemento in cima allo stack S

$$\sum_{i=1}^n tw2 = m + 0 + \dots + 0 \rightarrow m$$
$$\sum_{i=1}^n tw3 = 0 + \dots + 0 \rightarrow 0$$

$T_m(n, m) = 5n + 3m = \Omega(n + m)$

Peggior:

- Coda e stack non hanno mai elementi in comune

$$\sum_{i=1}^n tw2 = m * n = \sum_{i=1}^n tw3$$

$$Tp(m,n)=5n+7n*m=O(n*m)$$

Mettendo caso  $n = m$

$$\Omega(n + n) = \Omega(2n)$$

$$O(n * n) = O(n^2)$$