Titolo nota 17/03/2016

$A[1..n]$ $[1, n+1]$ ordinato

1) $O(n)$ intero mancante $n+1$ poteva essere l'intero mancante

Cerca1 (A);

$i = 1;$

while ($i \leq n$ && $A[i] == i$) $i{+}{+}$;

return i;

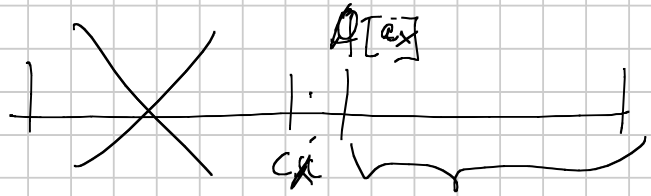
caso ottimo: $\Theta(1)$

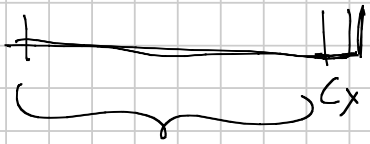
caso pessimo: $\Theta(n)$

caso medio: $\dfrac{1+2+3 \ldots + n + n}{n+1} = \dfrac{1}{n+1} \sum_{i=1}^{n} i + n =$

$= \dfrac{1}{\cancel{n+1}} \left(\dfrac{n\cancel{(n+1)}}{2} + n \right) = \dfrac{n}{2} + \dfrac{n}{n+1} = \Theta(n)$

Titolo nota 17/03/2016





Ricerca (A)

Ricerca (~~sinistra~~, n/2)
Ricerca (~~destra~~ n/2

$$T(n) = 2T\left(\frac{n}{2}\right) + \Theta(1)$$

$$T(n) = \Theta(n)$$

alg. lineare

Titolo nota 17/03/2016

Cerca 2 (A, sx, dx): sulla base di Ricerca Binaria

terminazione
```
if (sx > dx) return n+1;    Sinistra (test su sottoinsiemi di size 1)
   if (sx == dx)
       if (A[sx] == sx) return n+1;
       else return sx;
cx = (sx+dx)/2;
if (A[cx] == cx) return Cerca 2(A, cx+1, dx);
   else return Cerca 2(A, sx, cx);
```

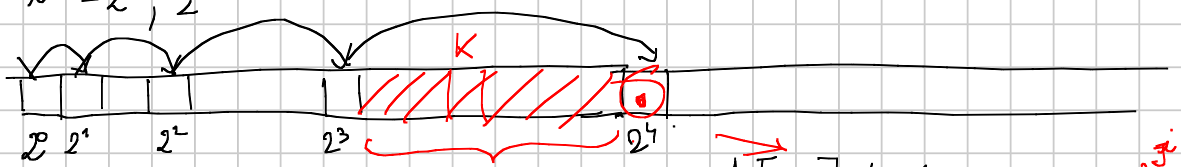
$$T(n) = \begin{cases} \Theta(1) & \text{per } n \leq 1 \\ T\left(\frac{n}{2}\right) + \Theta(1) \end{cases}$$

soluz. $T(n) = \Theta(\log n)$

se K è l'intero mancante $\Theta(\log K)$

$K = 10$ $n = 1000$ se $K = \Theta(n)$

$i = 2^0, 2^1$



$A[16] \neq 16$

$2^j - 2^{j-1} = 2^{j-1} \simeq K$

$2^{j-1} < K \leq 2^j$

$j = \lceil \log K \rceil$

- Il numero di ~~per~~ passi per localizzare K è $\lceil \log K \rceil$
- Ricerca Binaria (Cerca 2 nella zona rossa) $\Theta(\log K)$

Titolo nota 17/03/2016

Cerca3(A):

$i = 1;$

while ($i \leq n$ && $A[i] == i$) $i = 2 * i;$

if ($i > n$) return Cerca2 $(A, \frac{i}{2}+1, n);$

<u>else</u> // $A[i] \neq i$

return Cerca2 $(A, \frac{i}{2}+1, i);$

MergeSort2 (a)

⋮

MergeSort2 (a, sx, cx);

QuickSort (a, cx+1, dx);

merge (- - -)

$$T(n) = T\left(\frac{n}{2}\right) + \underbrace{\cancel{\Theta(n)} + \Theta\left(\frac{n^2}{\cancel{2}}\right)}_{\Theta(n^2)}$$

$$T(n) = T\left(\frac{n}{2}\right) + \Theta(n^2)$$

$a = 1 \quad b = 2 \quad n^{\log_2 1} = n^0$

$f(n) = n^2$

$n^2 = \Omega(n^{0+\varepsilon}) \quad \varepsilon \leq 2$

caso 3:

$$\boxed{T(n) = \Theta(n^2)}$$

se vale la regolarità

$a f\left(\frac{n}{b}\right) \leq c f(n)$

$\left(\frac{n}{2}\right)^2 \leq c n^2$

$\frac{n^2}{4} \leq c n^2 \qquad \frac{1}{4} \leq c < 1$

verificata

n monete 1 falsa sicuramente perché + leggera

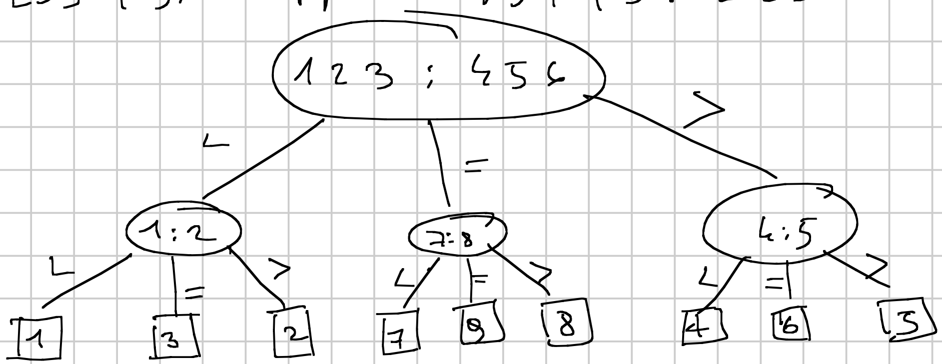
1. $n = 2^i$ si dividono 2 gruppi e si continua ricorsivamente sul gruppo + leggero $\Theta(\log_2 n)$

$n = 3^i$ $\Theta(\log_3 n)$ migliore del precedente

$n \neq 3^i$ $\lfloor \frac{n}{3} \rfloor \lfloor \frac{n}{3} \rfloor \lceil \frac{n}{3} \rceil$ oppure $\lceil \frac{n}{3} \rceil \lceil \frac{n}{3} \rceil \lfloor \frac{n}{3} \rfloor$

2. $n = 9$ 2 pesate

n° soluzioni = 9



Titolo nota

17/03/2016