

Merge : due seq di numeri ordinati

$A = a_1 \dots a_m$      $B = b_1 \dots b_n$  , ottenere una lista unica ordinata

$i \leq 1, j \leq 1$

while (entrambe le liste non sono vuote)

if  $a_i \leq b_j$  appendi alla lista L,  $i \leftarrow i+1$

else appendi  $b_j$  alla lista L,  $j \leftarrow j+1$

appendi a fine della lista non vuota ad L

Dopo ogni confronto la lista L aumenta di 1

$O(m)$

non

Tempo logaritmico :  $O(\log n)$

Eo. Ricerca Binaria : Data lista ord  $A = a_1 \dots a_n$  ed un valore key, determinare l'indice  $i$  per cui  $a_i = key$ , se non esiste

first  $\leftarrow 1$ , last  $\leftarrow n$

while (first  $\leq$  last)

mid = (first + last) / 2; (calcola punto medio)

if (key >  $a_{mid}$ )

first = mid + 1; (sposta la ricerca nella parte di destra)

else if (key <  $a_{mid}$ )

last = mid - 1; (sposta la ricerca nella parte di sinistra)

else return (mid)

return non c'è

al più l'algoritmo effettua  $n/2^k \leq 1$  (x non più)  $\Rightarrow k = O(\log n)$

al più su  $n/2$  all'inizio e poi su  $n/4 = n/2^2$  dopo le

k-esime iterazioni, ed in totale effettua  $n/2^k$  elementi