

Insertion sort

venerdì 18 marzo 2022 15:47

E' chiamato l'ordinamento delle carte.

Partiamo dal 2 numero e, lo confrontiamo con il 1. E' più piccolo? Va a sinistra, più grande? Rimani lì.

Ora, si va al 3 numero e lo confrontiamo prima con il 2 numero, più grande? Finiamo. Più piccolo?

Lo confrontiamo con il primo numero e continua così.

5	1	3	4
---	---	---	---

 ->

1	5	3	4
---	---	---	---

1	3	5	4
---	---	---	---

 ->

1	3	4	5
---	---	---	---

Quindi: Dato una posizione N, controlliamo tutte le posizioni antecedenti fino a che troviamo un valore più piccolo
E noi sappiamo che il nostro valore dovrà essere messo dopo questo valore più piccolo.

Algoritmo:

Void insertionSort(V[])

For j = 2 to n:

APP = V[j]

i = j - 1

While (APP < V[i]) AND (i > 0)

V[i+1] = V[i]

i--;

V[i+1] = APP

C1(n-1)

C2(n-1)

C3(n-1)

3C4+Twj

$\rightarrow \sum_{j=2}^n Twj$

$\rightarrow C5*(n-1)$

$$T_{inSort} = 4c(n-1) + 3c \sum_{j=2}^n Twj$$

Caso migliore

$Twj = 0 \forall j \rightarrow$ il while è sempre false

$$T_m(n) = 4c(n-1) + 3c \sum 0 = 4c(n-1) \approx n$$

Caso peggiore

Il while viene eseguito finché $i > 0$

$\rightarrow Twj = j - 1$

$$T_p(n) = 4c(n-1) + 3c \sum_{j=2}^n (j-1) \rightarrow scritta \rightarrow 1 + 2 + \dots + (n-1)$$

$$= 4c(n-1) + 3c \sum_{j=1}^{n-1} j \rightarrow 4c + 3c * \frac{(n-1)(n)}{2} \approx n^2$$

$$\text{Caso medio } Twj = \frac{j-1}{2}$$