



**UNIVERSITÀ
DEGLI STUDI
DI TRIESTE**

MODULO 2: Algoritmi

Prof.ssa Giulia Cisotto

giulia.cisotto@units.it

Trieste, 14 maggio 2025

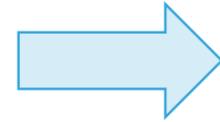


UNIVERSITÀ
DEGLI STUDI
DI TRIESTE

- ▶ Algoritmi di ordinamento: introduzione
- ▶ Bogosort, Insertion sort, Merge Sort

Il PROBLEMA dell'ORDINAMENTO è un task comune a molte applicazioni reali
... per questo abbiamo diversi algoritmi in grado di risolverlo! Non serve reinventare la ruota!

Algoritmi semplici, ma non molto efficienti



Bogosort (stupid sort/monkey sort/permutation sort), **Insertion sort**, **selection sort**, **bubblesort**

Algoritmi più efficienti di tipo comparativo



Mergesort, quicksort, heapsort

Algoritmi di tipo non comparativo, particolarmente efficienti in casi specifici



Couting sort, radix sort, bucket sort

ORDINARE UN MAZZO DI CARTE

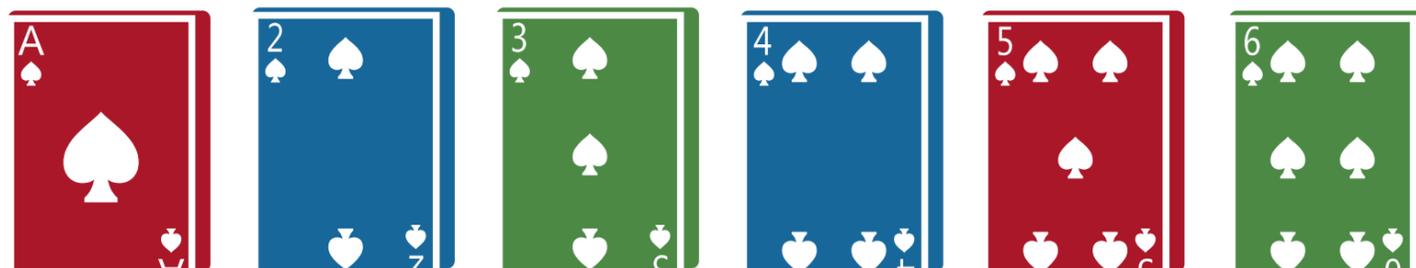


Per ogni permutazione delle carte del mazzo

1. Permuta le carte
2. Se le carte sono ordinate termina l'esecuzione
3. Altrimenti riportale alla configurazione iniziale



MOLTO TEMPO DOPO...



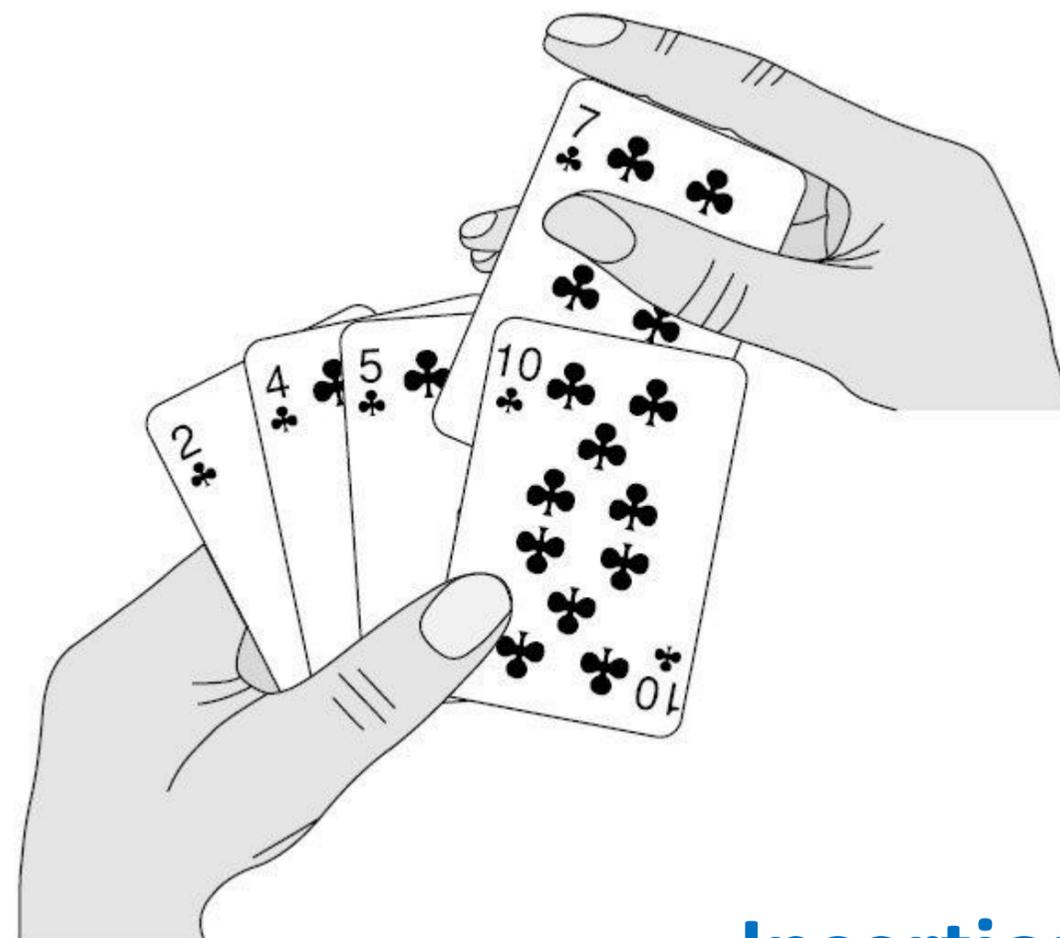
PRESTAZIONI

- ▶ **CORRETTEZZA:** l'algoritmo funziona (necessariamente troverà una permutazione per cui le carte sono ordinate).
- ▶ **EFFICIENZA:** quante comparazioni dobbiamo fare con n carte?
 - ▶ **Caso migliore:** circa n
(per verificare che le carte siano ordinate)
 - ▶ **Caso peggiore:** fino ad $n!$
(ci sono $n!$ permutazioni e potremmo riuscire ad avere le carte ordinate solo all'ultima)

ORDINARE UN MAZZO DI CARTE

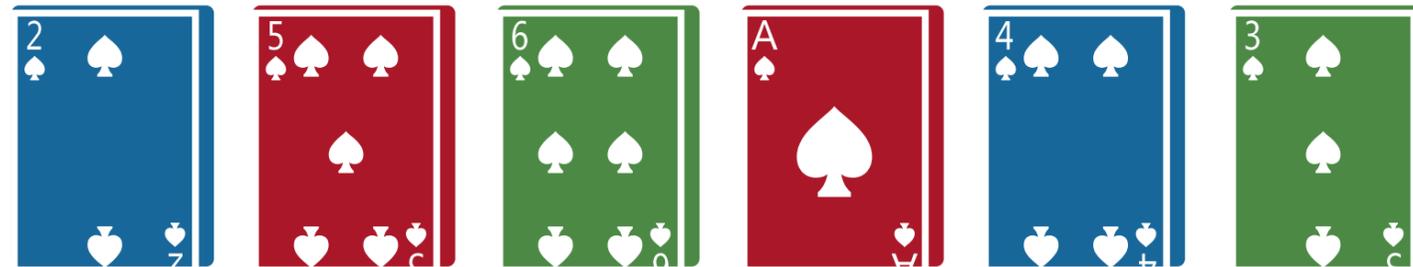


Possiamo pensare ad un modo alternativo?



Insertion sort!

INSERTION SORT: ESEMPIO



1. Inizialmente non si hanno carte ordinate
2. Finché ci rimangono carte non ordinate
 1. Prendi una carta dalla lista di carte non ordinata e chiamala X
 2. Metti X in fondo alla lista di carte ordinate
 3. Finché la carta che precede X ha valore maggiore del valore di X, scambia X con la carta che lo precede

INSERTION SORT: ESEMPIO



1. Inizialmente non si hanno carte ordinate
2. Finché ci rimangono carte non ordinate
 1. Prendi una carta dalla lista di carte non ordinata e chiamala X
 2. Metti X in fondo alla lista di carte ordinate
 3. Finché la carta che precede X ha valore maggiore del valore di X, scambia X con la carta che lo precede

INSERTION SORT: ESEMPIO

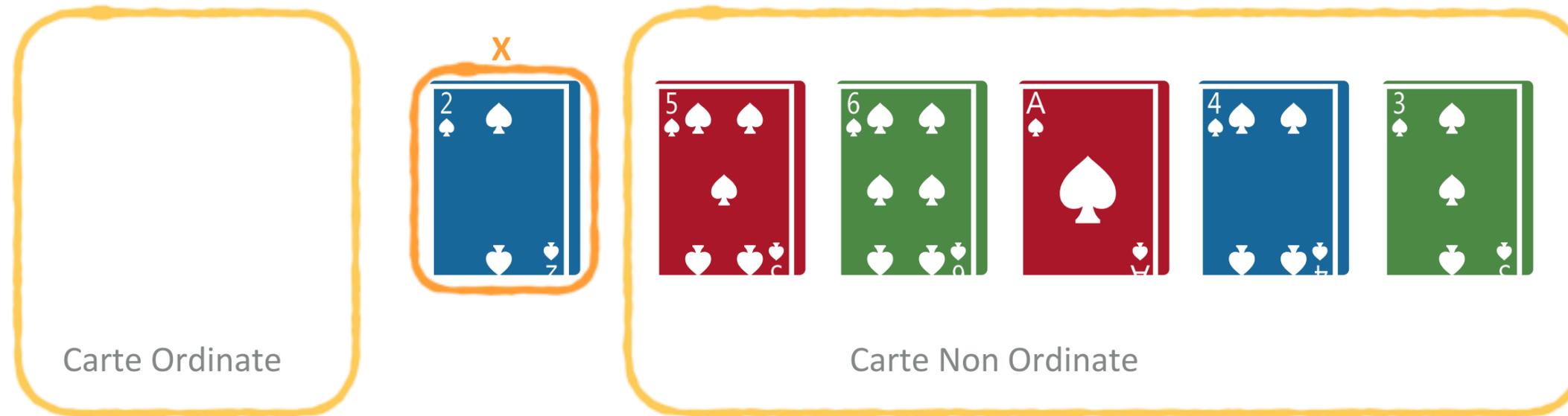


1. Inizialmente non si hanno carte ordinate

→ 2. Finché ci rimangono carte non ordinate

1. Prendi una carta dalla lista di carte non ordinata e chiamala X
2. Metti X in fondo alla lista di carte ordinate
3. Finché la carta che precede X ha valore maggiore del valore di X, scambia X con la carta che lo precede

INSERTION SORT: ESEMPIO



1. Inizialmente non si hanno carte ordinate

2. Finché ci rimangono carte non ordinate



1. Prendi una carta dalla lista di carte non ordinate e chiamala X

2. Metti X in fondo alla lista di carte ordinate

3. Finché la carta che precede X ha valore maggiore del valore di X, scambia X con la carta che lo precede

INSERTION SORT: ESEMPIO



1. Inizialmente non si hanno carte ordinate
2. Finché ci rimangono carte non ordinate
 1. Prendi una carta dalla lista di carte non ordinata e chiamala X
 - 2. Metti X in fondo alla lista di carte ordinate
 3. Finché la carta che precede X ha valore maggiore del valore di X, scambia X con la carta che lo precede

INSERTION SORT: ESEMPIO

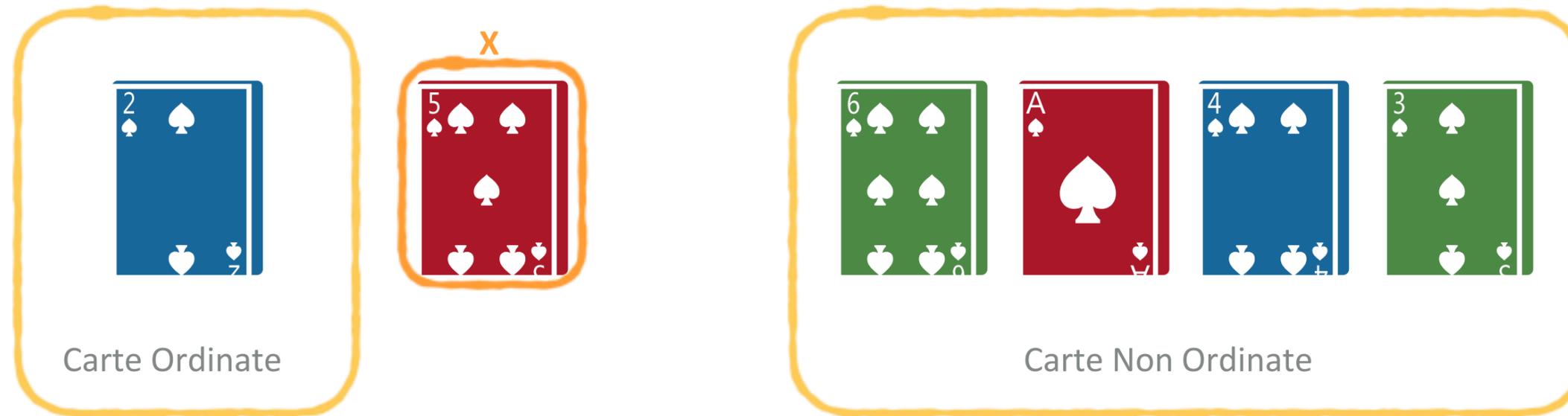


1. Inizialmente non si hanno carte ordinate
2. Finché ci rimangono carte non ordinate

**NON DOBBIAMO FARE NIENTE
(NON CI SONO ALTRE CARTE GIÀ IN ORDINE)**

1. Prendi una carta dalla lista di carte non ordinata e chiamala X
2. Metti X in fondo alla lista di carte ordinate
- 3. Finché la carta che precede X ha valore maggiore del valore di X, scambia X con la carta che lo precede

INSERTION SORT: ESEMPIO



1. Inizialmente non si hanno carte ordinate

2. Finché ci rimangono carte non ordinate

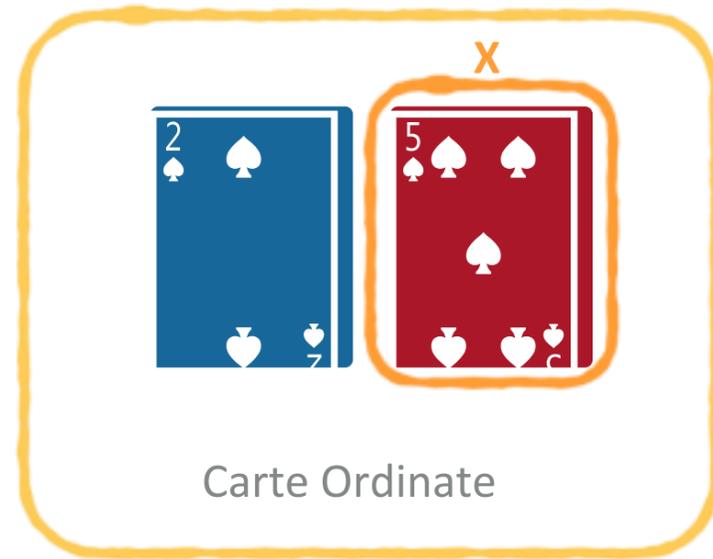


1. Prendi una carta dalla lista di carte non ordinate e chiamala X

2. Metti X in fondo alla lista di carte ordinate

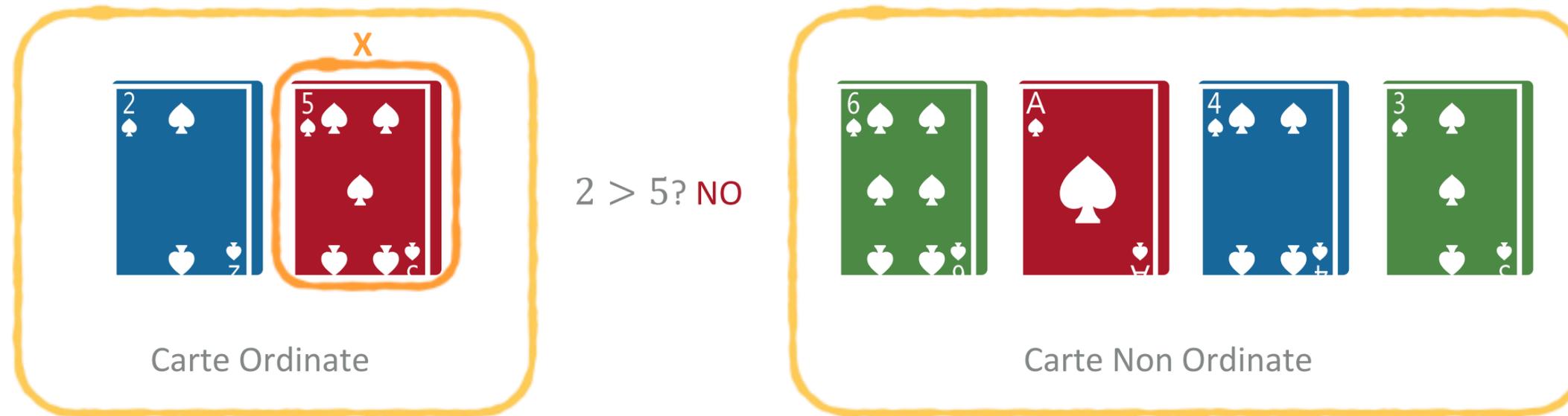
3. Finché la carta che precede X ha valore maggiore del valore di X, scambia X con la carta che lo precede

INSERTION SORT: ESEMPIO



1. Inizialmente non si hanno carte ordinate
2. Finché ci rimangono carte non ordinate
 1. Prendi una carta dalla lista di carte non ordinata e chiamala X
 - 2. Metti X in fondo alla lista di carte ordinate
 3. Finché la carta che precede X ha valore maggiore del valore di X, scambia X con la carta che lo precede

INSERTION SORT: ESEMPIO

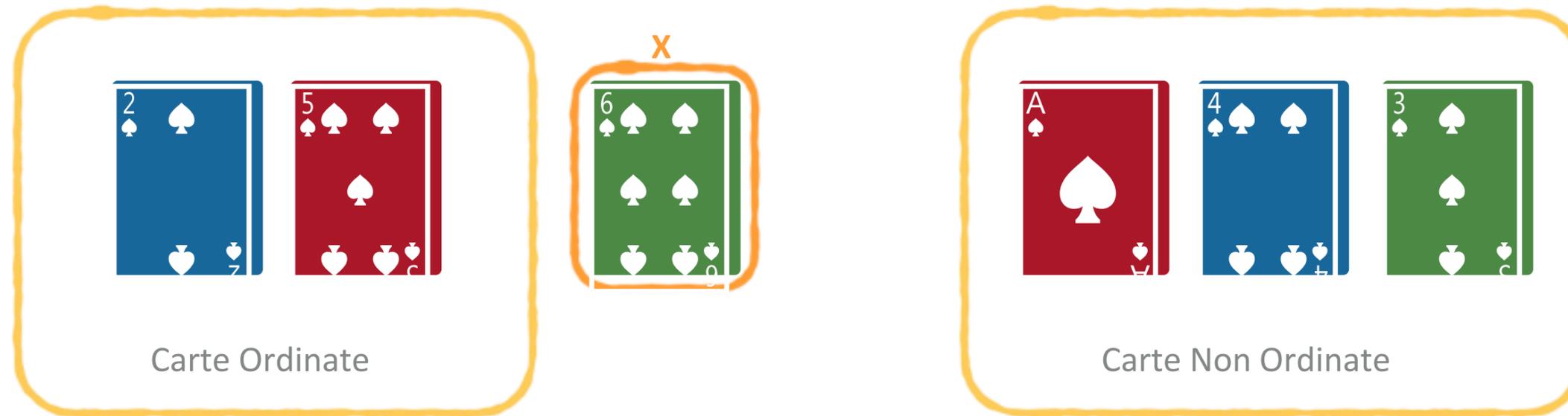


1. Inizialmente non si hanno carte ordinate
2. Finché ci rimangono carte non ordinate

IL TEST FALLISCE
E QUINDI PROSEGUIREMO
CON L'ISTRUZIONE NUMERO...

1. Prendi una carta dalla lista di carte non ordinata e chiamala X
2. Metti X in fondo alla lista di carte ordinate
- 3. Finché la carta che precede X ha valore maggiore del valore di X, scambia X con la carta che lo precede

INSERTION SORT: ESEMPIO



1. Inizialmente non si hanno carte ordinate

2. Finché ci rimangono carte non ordinate



1. Prendi una carta dalla lista di carte non ordinata e chiamala X

2. Metti X in fondo alla lista di carte ordinate

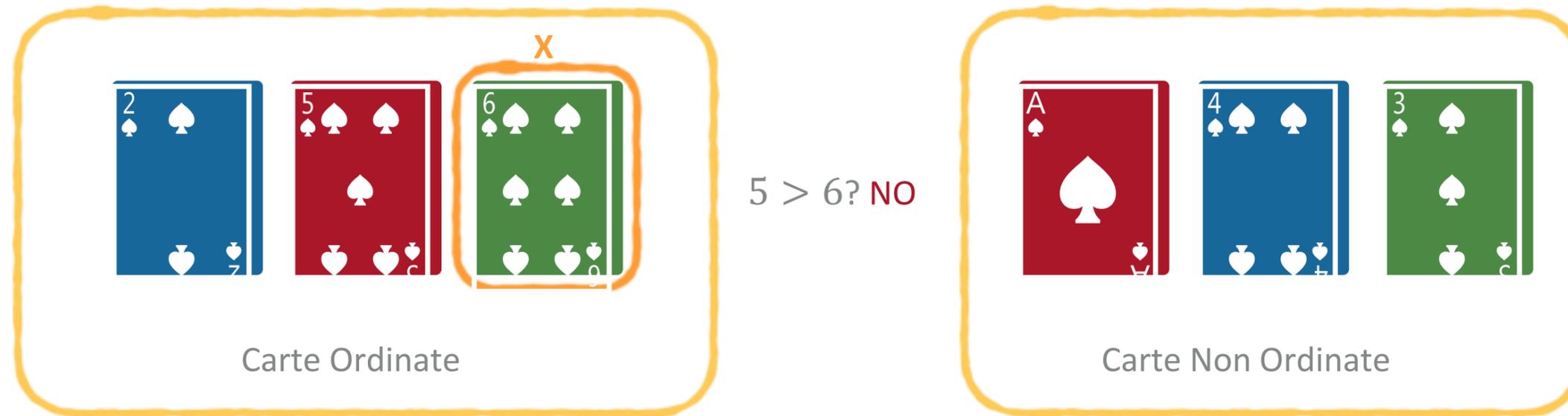
3. Finché la carta che precede X ha valore maggiore del valore di X, scambia X con la carta che lo precede

INSERTION SORT: ESEMPIO



1. Inizialmente non si hanno carte ordinate
2. Finché ci rimangono carte non ordinate
 1. Prendi una carta dalla lista di carte non ordinata e chiamala X
 - ➔ 2. Metti X in fondo alla lista di carte ordinate
 3. Finché la carta che precede X ha valore maggiore del valore di X, scambia X con la carta che lo precede

INSERTION SORT: ESEMPIO



1. Inizialmente non si hanno carte ordinate

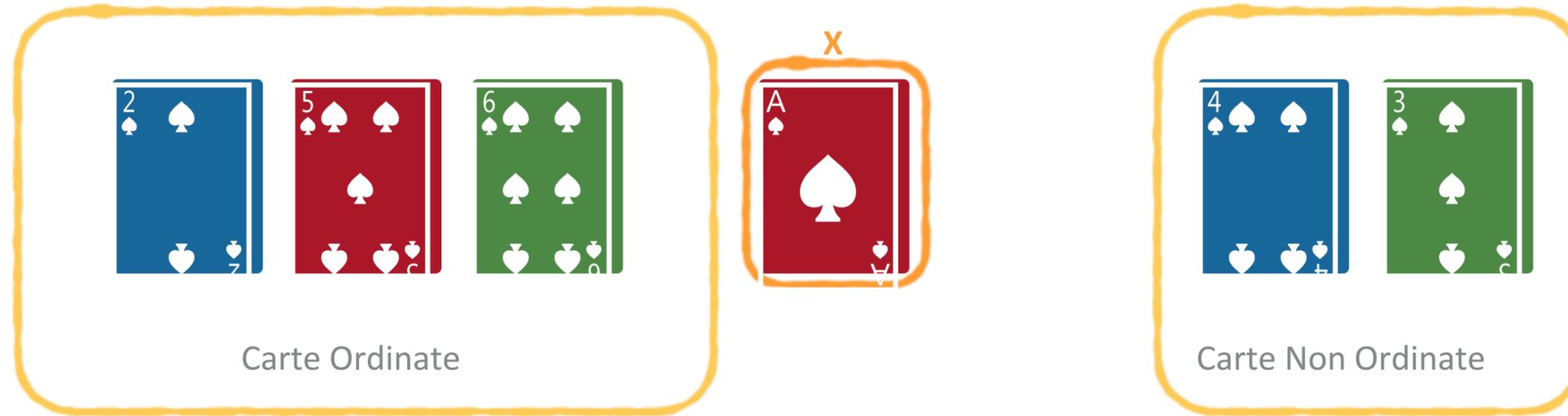
2. Finché ci rimangono carte non ordinate

1. Prendi una carta dalla lista di carte non ordinata e chiamala X

2. Metti X in fondo alla lista di carte ordinate

→ 3. Finché la carta che precede X ha valore maggiore del valore di X, scambia X con la carta che lo precede

INSERTION SORT: ESEMPIO



1. Inizialmente non si hanno carte ordinate

2. Finché ci rimangono carte non ordinate



1. Prendi una carta dalla lista di carte non ordinata e chiamala X

2. Metti X in fondo alla lista di carte ordinate

3. Finché la carta che precede X ha valore maggiore del valore di X, scambia X con la carta che lo precede

INSERTION SORT: ESEMPIO

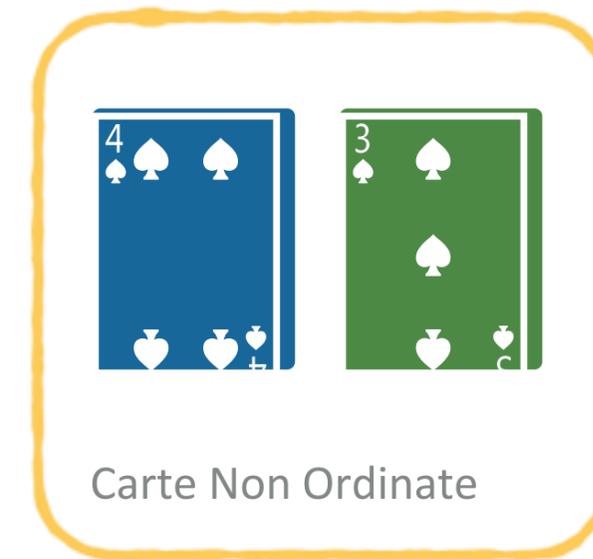


1. Inizialmente non si hanno carte ordinate
2. Finché ci rimangono carte non ordinate
 1. Prendi una carta dalla lista di carte non ordinata e chiamala X
 - ➔ 2. Metti X in fondo alla lista di carte ordinate
 3. Finché la carta che precede X ha valore maggiore del valore di X, scambia X con la carta che lo precede

INSERTION SORT: ESEMPIO



6 > 1? SI



QUI INVECE IL TEST HA SUCCESSO
QUINDI DOBBIAMO ESEGUIRE
L'OPERAZIONE DI SCAMBIO

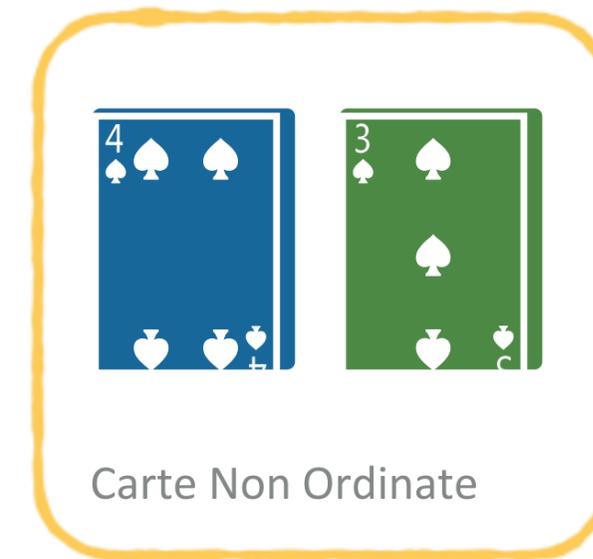
1. Inizialmente non si hanno carte ordinate
2. Finché ci rimangono carte non ordinate

1. Prendi una carta dalla lista di carte non ordinata e chiamala X
2. Metti X in fondo alla lista di carte ordinate
- 3. Finché la carta che precede X ha valore maggiore del valore di X, scambia X con la carta che lo precede

INSERTION SORT: ESEMPIO



5 > 1? SI

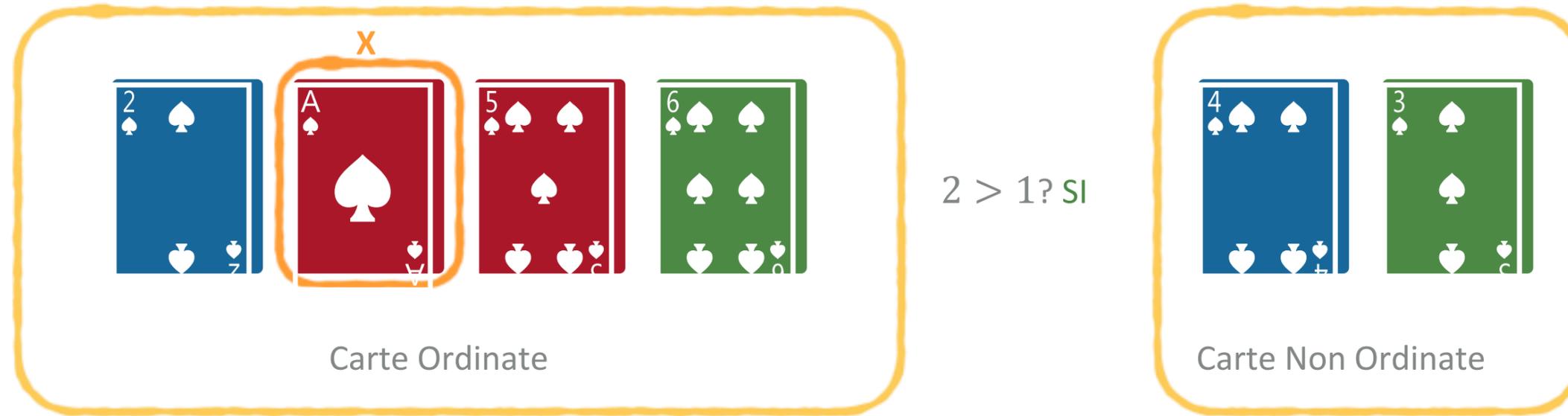


1. Inizialmente non si hanno carte ordinate
2. Finché ci rimangono carte non ordinate

**ORA CHE DOBBIAMO FARE?
CONTINUIAMO CON L'ISTRUZIONE 2.3
O TORNIAMO ALLA 2.1?**

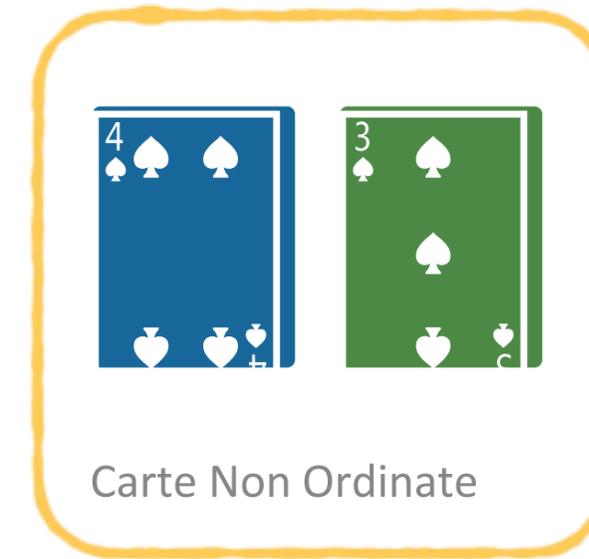
1. Prendi una carta dalla lista di carte non ordinata e chiamala X
2. Metti X in fondo alla lista di carte ordinate
- 3. Finché la carta che precede X ha valore maggiore del valore di X, scambia X con la carta che lo precede

INSERTION SORT: ESEMPIO



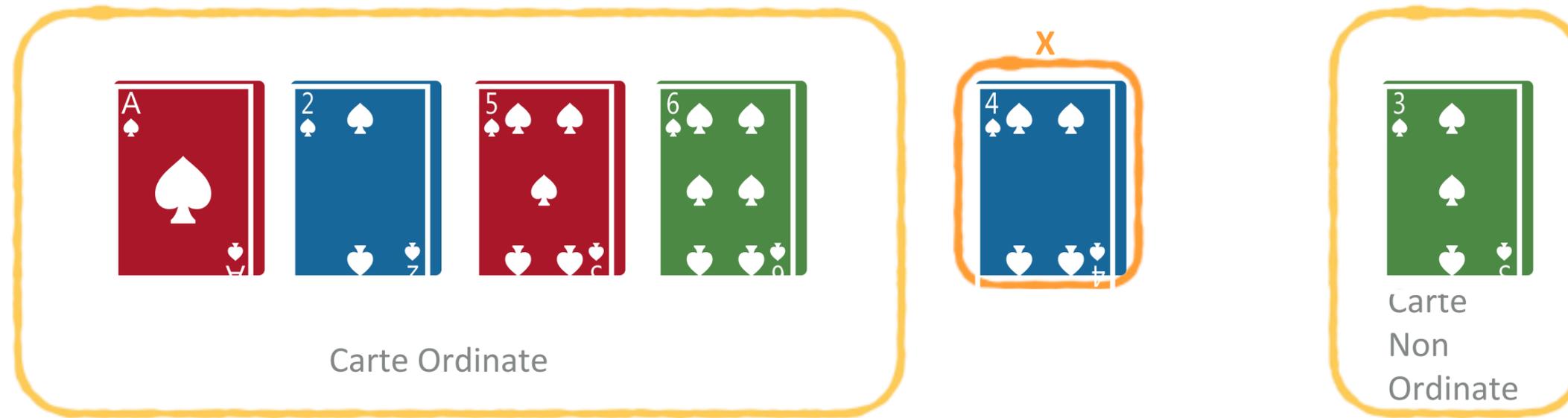
1. Inizialmente non si hanno carte ordinate
2. Finché ci rimangono carte non ordinate
 1. Prendi una carta dalla lista di carte non ordinata e chiamala X
 2. Metti X in fondo alla lista di carte ordinate
- 3. Finché la carta che precede X ha valore maggiore del valore di X, scambia X con la carta che lo precede

INSERTION SORT: ESEMPIO



1. Inizialmente non si hanno carte ordinate
2. Finché ci rimangono carte non ordinate
 1. Prendi una carta dalla lista di carte non ordinata e chiamala X
 2. Metti X in fondo alla lista di carte ordinate
- 3. Finché la carta che precede X ha valore maggiore del valore di X, scambia X con la carta che lo precede

INSERTION SORT: ESEMPIO



1. Inizialmente non si hanno carte ordinate

2. Finché ci rimangono carte non ordinate

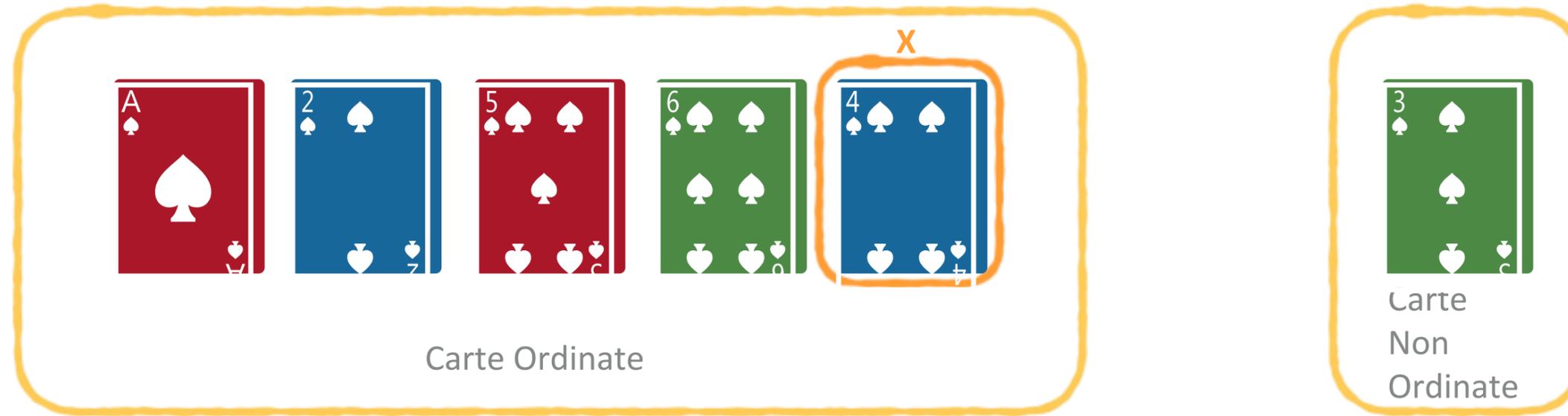


1. Prendi una carta dalla lista di carte non ordinata e chiamala X

2. Metti X in fondo alla lista di carte ordinate

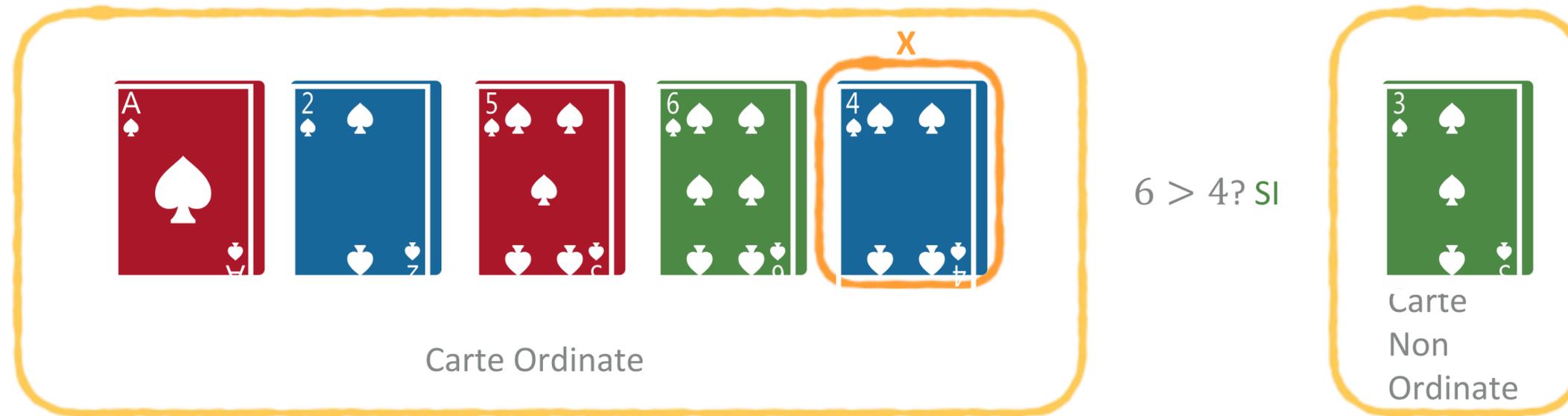
3. Finché la carta che precede X ha valore maggiore del valore di X, scambia X con la carta che lo precede

INSERTION SORT: ESEMPIO



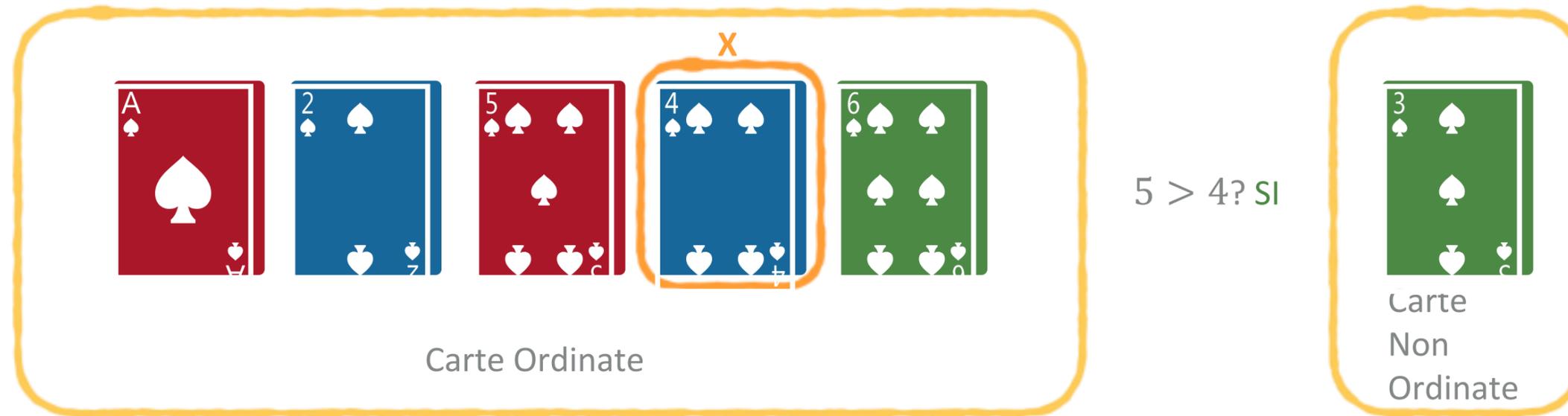
1. Inizialmente non si hanno carte ordinate
2. Finché ci rimangono carte non ordinate
 1. Prendi una carta dalla lista di carte non ordinata e chiamala X
 - 2. Metti X in fondo alla lista di carte ordinate
 3. Finché la carta che precede X ha valore maggiore del valore di X, scambia X con la carta che lo precede

INSERTION SORT: ESEMPIO



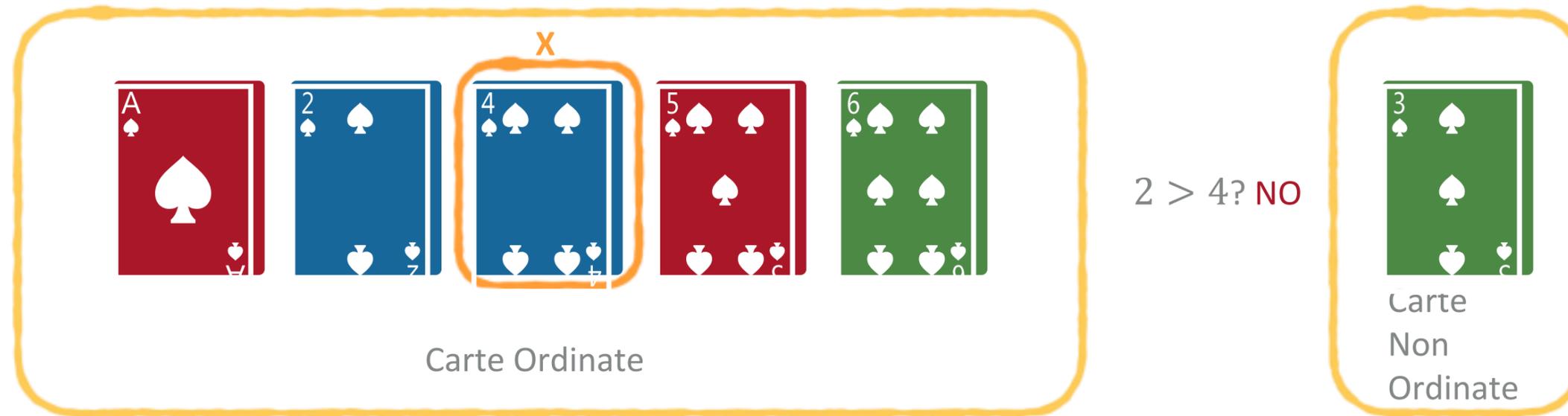
1. Inizialmente non si hanno carte ordinate
2. Finché ci rimangono carte non ordinate
 1. Prendi una carta dalla lista di carte non ordinata e chiamala X
 2. Metti X in fondo alla lista di carte ordinate
- 3. Finché la carta che precede X ha valore maggiore del valore di X, scambia X con la carta che lo precede

INSERTION SORT: ESEMPIO



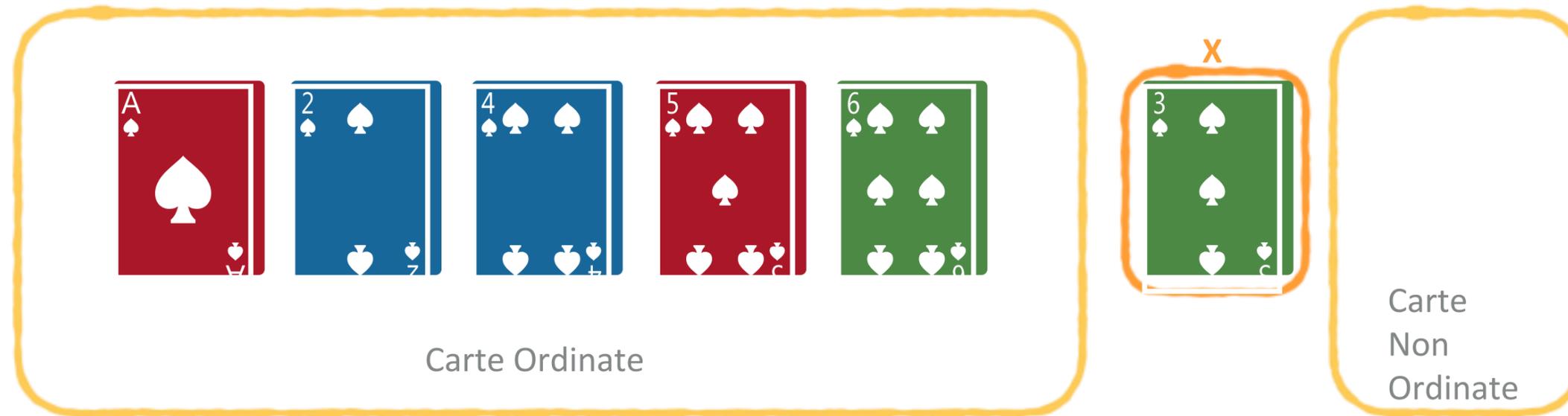
1. Inizialmente non si hanno carte ordinate
2. Finché ci rimangono carte non ordinate
 1. Prendi una carta dalla lista di carte non ordinata e chiamala X
 2. Metti X in fondo alla lista di carte ordinate
 - 3. Finché la carta che precede X ha valore maggiore del valore di X, scambia X con la carta che lo precede

INSERTION SORT: ESEMPIO



1. Inizialmente non si hanno carte ordinate
2. Finché ci rimangono carte non ordinate
 1. Prendi una carta dalla lista di carte non ordinata e chiamala X
 2. Metti X in fondo alla lista di carte ordinate
- 3. Finché la carta che precede X ha valore maggiore del valore di X, scambia X con la carta che lo precede

INSERTION SORT: ESEMPIO



1. Inizialmente non si hanno carte ordinate

2. Finché ci rimangono carte non ordinate



1. Prendi una carta dalla lista di carte non ordinata e chiamala X

2. Metti X in fondo alla lista di carte ordinate

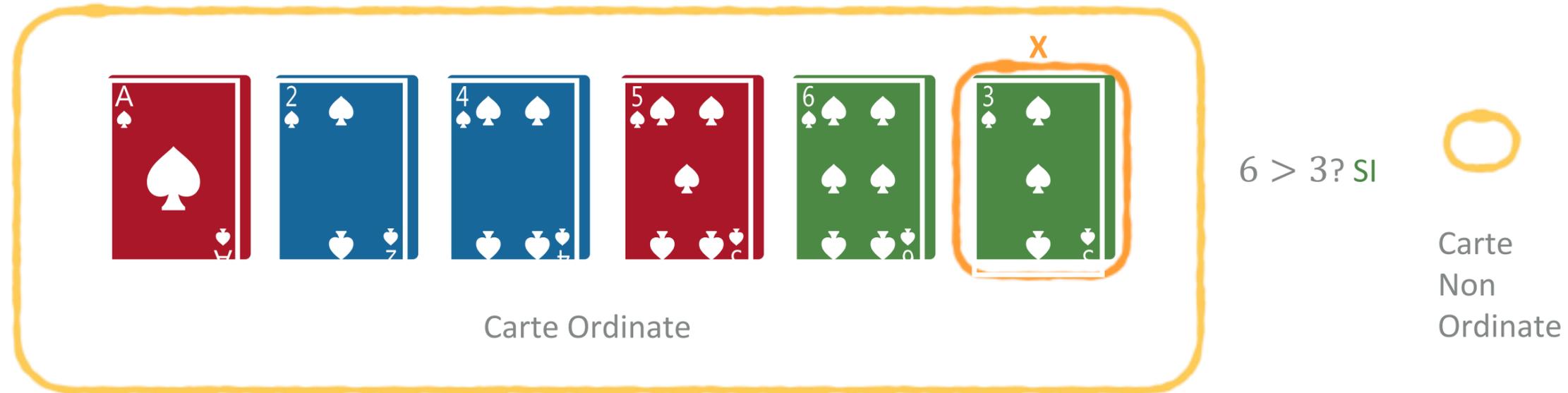
3. Finché la carta che precede X ha valore maggiore del valore di X, scambia X con la carta che lo precede

INSERTION SORT: ESEMPIO



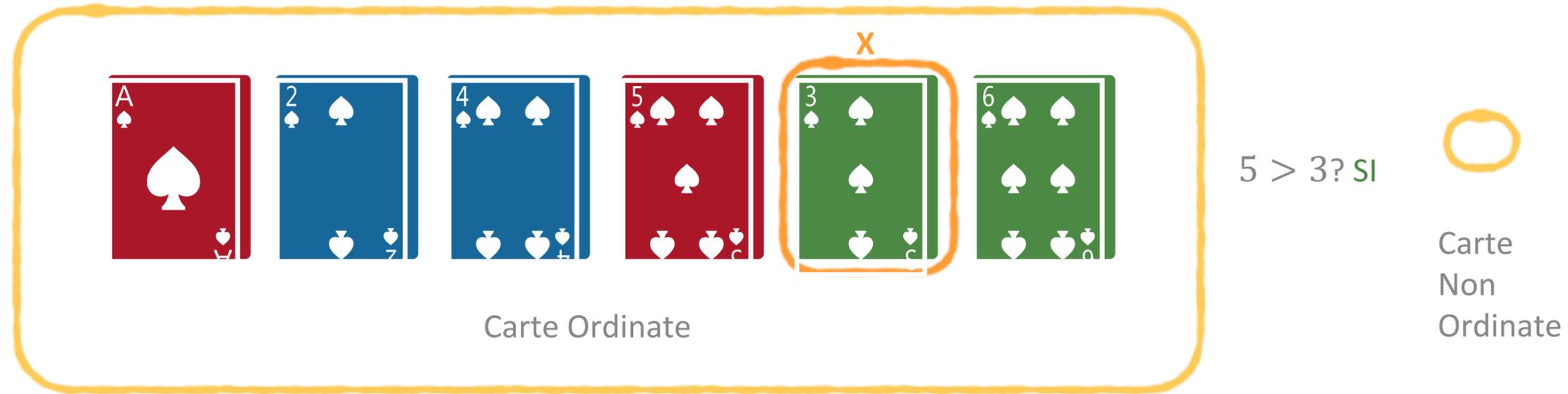
1. Inizialmente non si hanno carte ordinate
2. Finché ci rimangono carte non ordinate
 1. Prendi una carta dalla lista di carte non ordinata e chiamala X
 - 2. Metti X in fondo alla lista di carte ordinate
 3. Finché la carta che precede X ha valore maggiore del valore di X, scambia X con la carta che lo precede

INSERTION SORT: ESEMPIO



1. Inizialmente non si hanno carte ordinate
2. Finché ci rimangono carte non ordinate
 1. Prendi una carta dalla lista di carte non ordinata e chiamala X
 2. Metti X in fondo alla lista di carte ordinate
- 3. Finché la carta che precede X ha valore maggiore del valore di X, scambia X con la carta che lo precede

INSERTION SORT: ESEMPIO



1. Inizialmente non si hanno carte ordinate
2. Finché ci rimangono carte non ordinate
 1. Prendi una carta dalla lista di carte non ordinata e chiamala X
 2. Metti X in fondo alla lista di carte ordinate
- 3. Finché la carta che precede X ha valore maggiore del valore di X, scambia X con la carta che lo precede

INSERTION SORT: ESEMPIO



1. Inizialmente non si hanno carte ordinate
2. Finché ci rimangono carte non ordinate
 1. Prendi una carta dalla lista di carte non ordinata e chiamala X
 2. Metti X in fondo alla lista di carte ordinate
- 3. Finché la carta che precede X ha valore maggiore del valore di X, scambia X con la carta che lo precede

INSERTION SORT: ESEMPIO



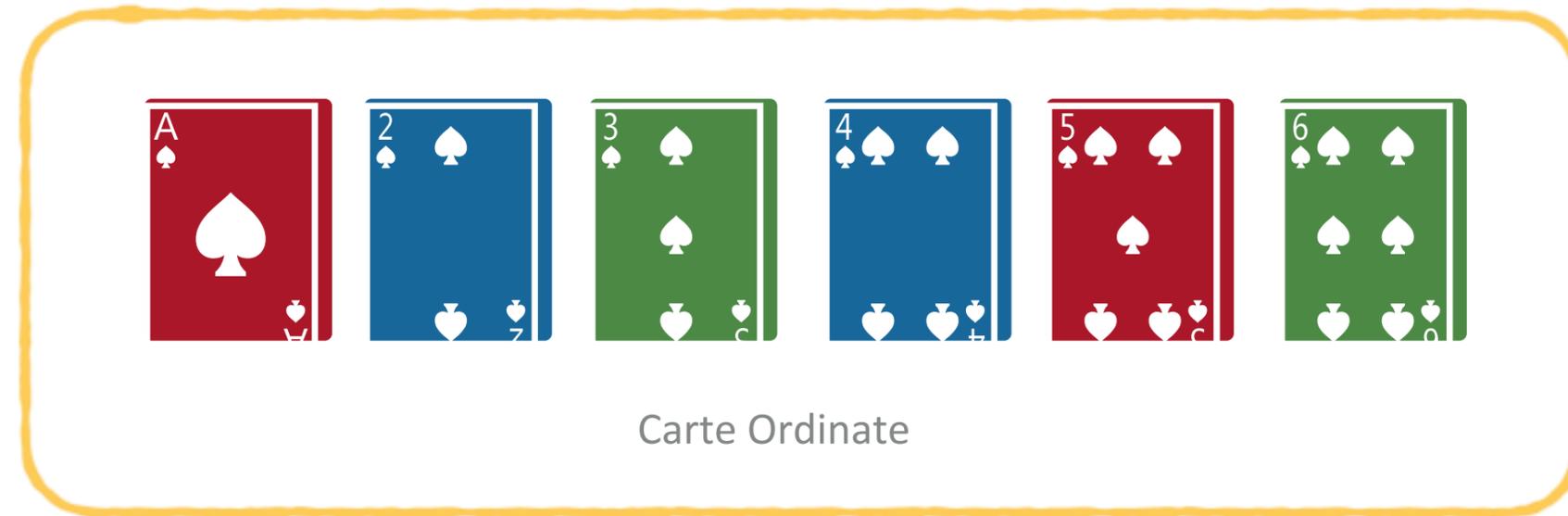
1. Inizialmente non si hanno carte ordinate

→ 2. Finché ci rimangono carte non ordinate

**ORA LA CONDIZIONE NON È PIÙ VERA
E ABBIAMO TERMINATO**

1. Prendi una carta dalla lista di carte non ordinate e chiamala X
2. Metti X in fondo alla lista di carte ordinate
3. Finché la carta che precede X ha valore maggiore del valore di X, scambia X con la carta che lo precede

INSERTION SORT: ESEMPIO



Siamo riusciti ad ottenere un mazzo di carte ordinato

Rimangono alcune domande:

- ▶ Correttezza: questo algoritmo funziona con ogni input?
- ▶ Efficienza: quanti passi ci mettiamo ad ordinare un mazzo di n carte al crescere di n ?

PRESTAZIONI

- ▶ **CORRETTEZZA:** l'algoritmo funziona, esiste senz'altro una soluzione per ogni possibile *istanza* del problema
- ▶ **EFFICIENZA:** quante comparazioni dobbiamo fare con n carte?
 - ▶ **Caso migliore:** circa n
(le estraiamo dall'insieme di partenza già ordinate)
 - ▶ **Caso peggiore:** al massimo $n * n$
(dobbiamo al massimo fare n confronti per ogni carta per posizionarla nell'ordine giusto)

PSEUDOCODICE

INSERTION-SORT (A, n)

```
for i = 2 to n
    key = A[i]      //inserisce A[i] nella sequenza ordinata A[1:i-1]
    j = i-1
    while j>0 && A[j]>key
        A[j+1] = A[j]
        j=j-1
    A[j+1] = key
```

Nota. Attenzione che nel libro c'è un typo (una «i» che invece deve essere «j»).

ORDINARE UN MAZZO DI CARTE



Possiamo pensare ad un algoritmo più efficiente?



Mergesort!

MERGESORT: ALGORITMO

1. Se il mazzo ha 1 carta è ordinato
2. Altrimenti spezza il mazzo in due parti $\lfloor n/2 \rfloor$ e $\lceil n/2 \rceil$
3. Applica questa procedura ad entrambe le parti
4. Unisci le due parti A e B seguendo la procedura MERGE, per unire due mazzi ordinati in un unico mazzo anch'esso ordinato

DOPO QUESTO PASSO
LE DUE PARTI SONO ORDINATE



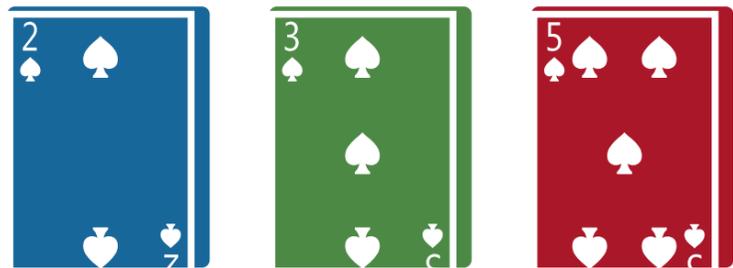
LA PROCEDURA DI MERGE VERRÀ
DETTAGLIATA NELLE PROSSIME SLIDE



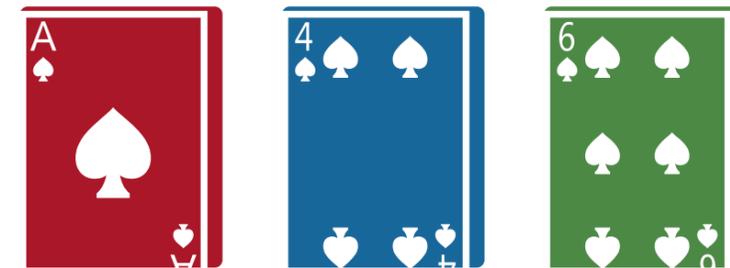
MERGESORT: PROCEDURA MERGE

Ipotesi: i due mazzi sono ordinati

Mazzo A



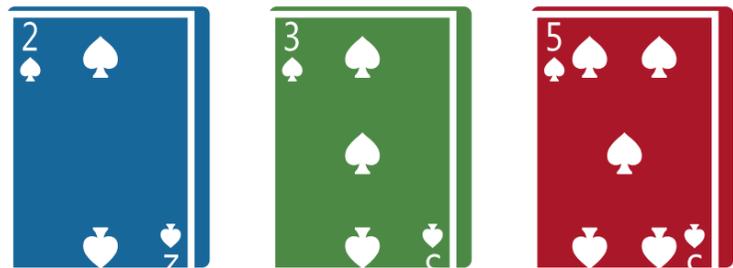
Mazzo B



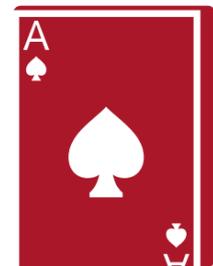
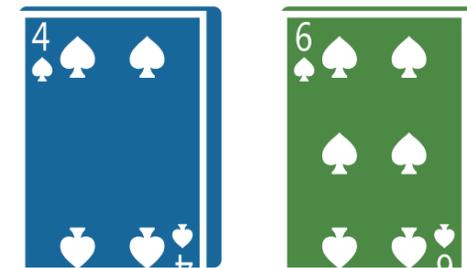
Mazzo unificato

MERGESORT: PROCEDURA MERGE

Mazzo *A*



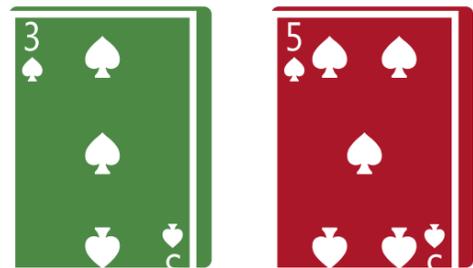
Mazzo *B*



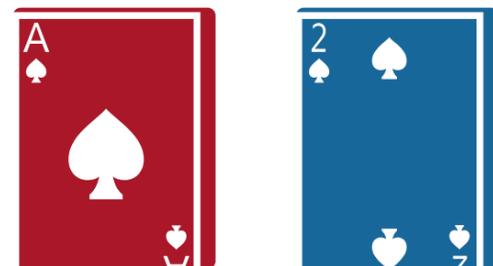
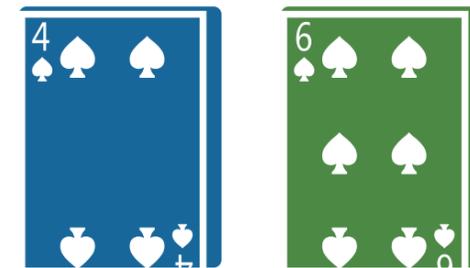
Mazzo unificato

MERGESORT: PROCEDURA MERGE

Mazzo *A*



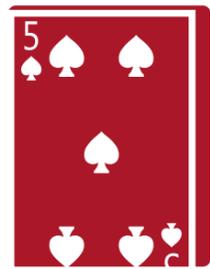
Mazzo *B*



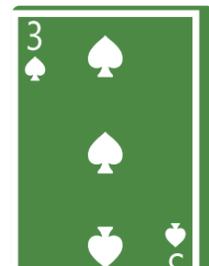
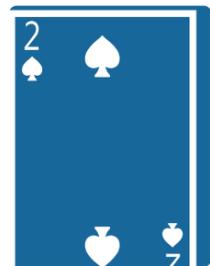
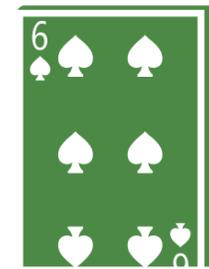
Mazzo unificato

MERGESORT: PROCEDURA MERGE

Mazzo *A*



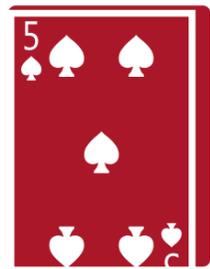
Mazzo *B*



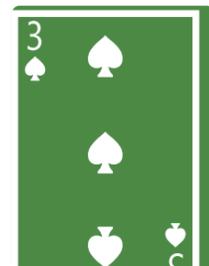
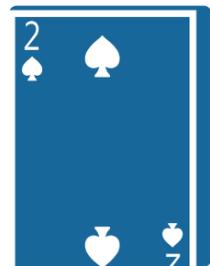
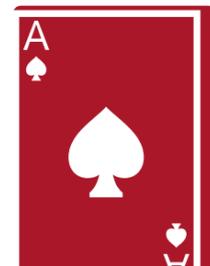
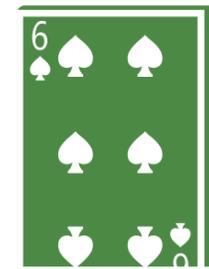
Mazzo unificato

MERGESORT: PROCEDURA MERGE

Mazzo *A*



Mazzo *B*

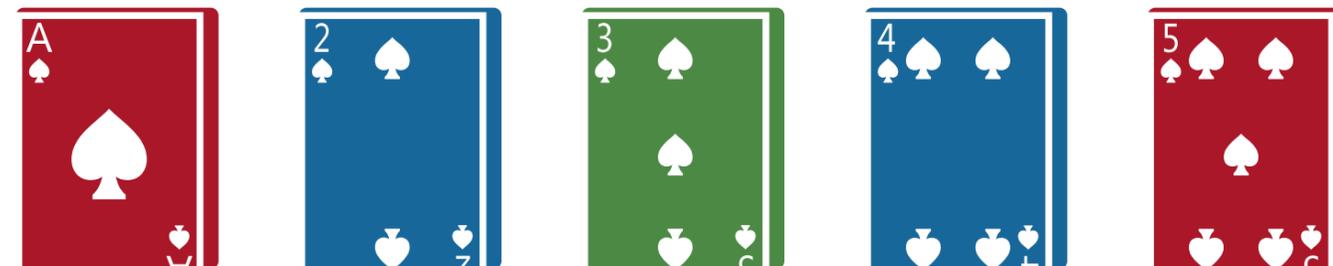
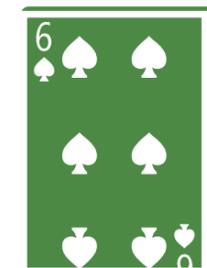


Mazzo unificato

MERGESORT: PROCEDURA MERGE

Mazzo *A*

Mazzo *B*

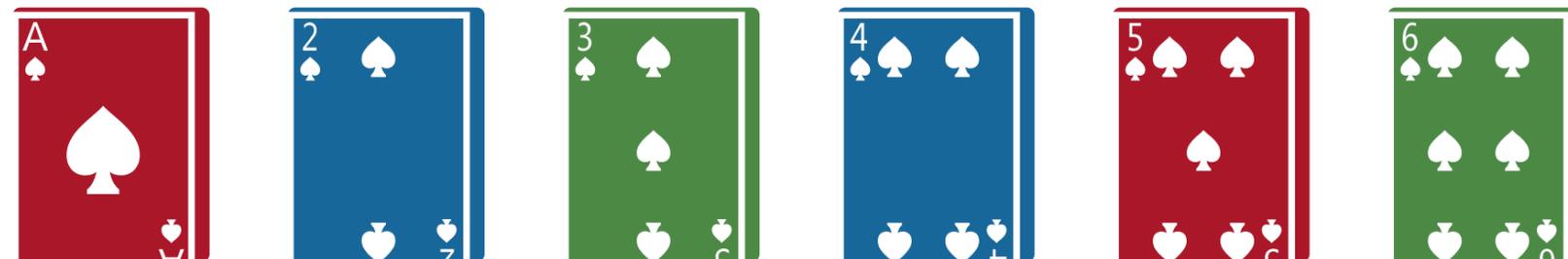


Mazzo unificato

MERGESORT: PROCEDURA MERGE

Mazzo *A*

Mazzo *B*



Mazzo unificato

MERGESORT: ESEMPIO



SPEZZARE IL MAZZO

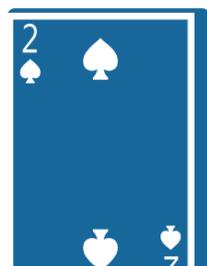
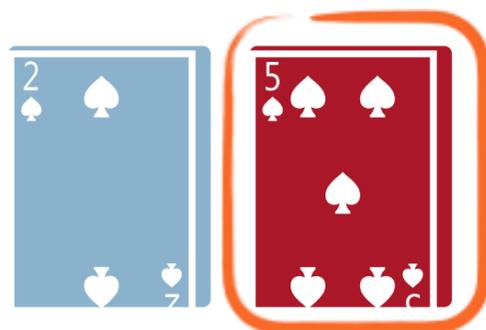
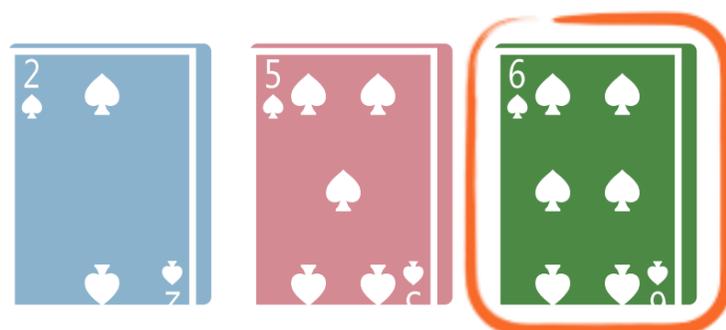
MERGESORT: ESEMPIO



MERGESORT: ESEMPIO

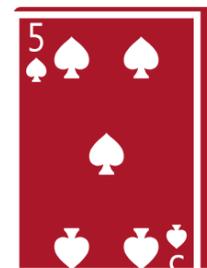
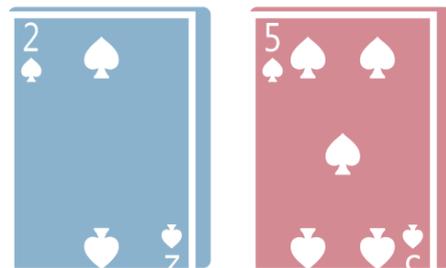
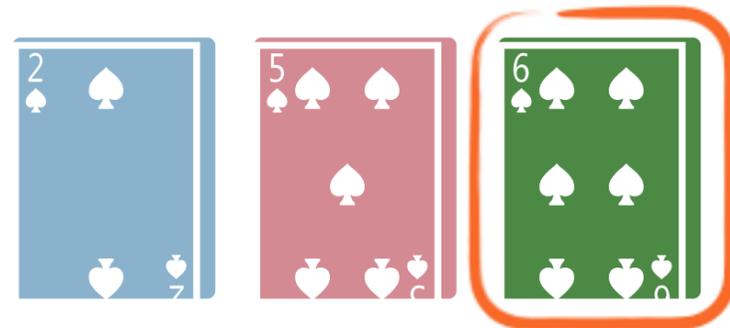


MERGESORT: ESEMPIO



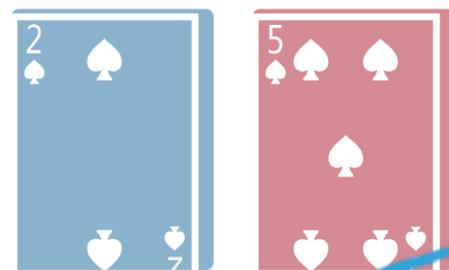
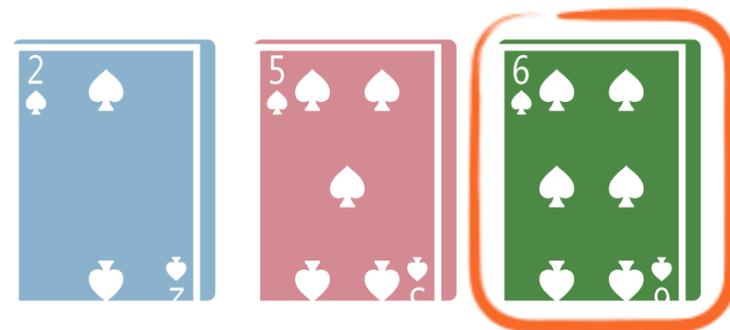
QUESTO MAZZO È COMPOSTO
DA UNA CARTA SOLA
E QUINDI È ORDINATO

MERGESORT: ESEMPIO

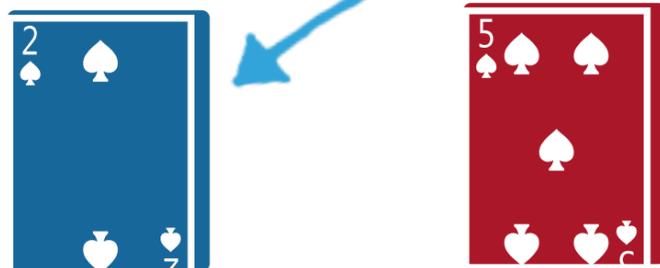


QUESTO MAZZO È COMPOSTO
DA UNA CARTA SOLA
E QUINDI È ORDINATO

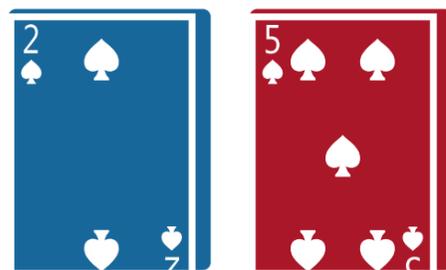
MERGESORT: ESEMPIO



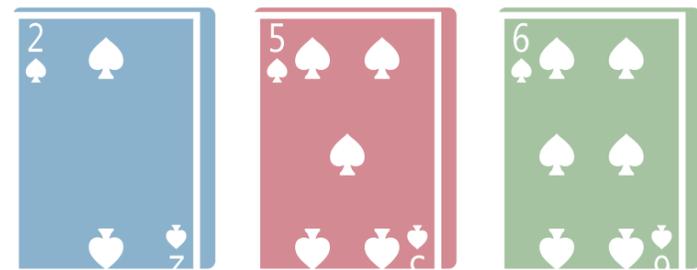
UNIFICHIAMO QUESTI DUE MAZZI
CON LA PROCEDURA DI MERGE



MERGESORT: ESEMPIO

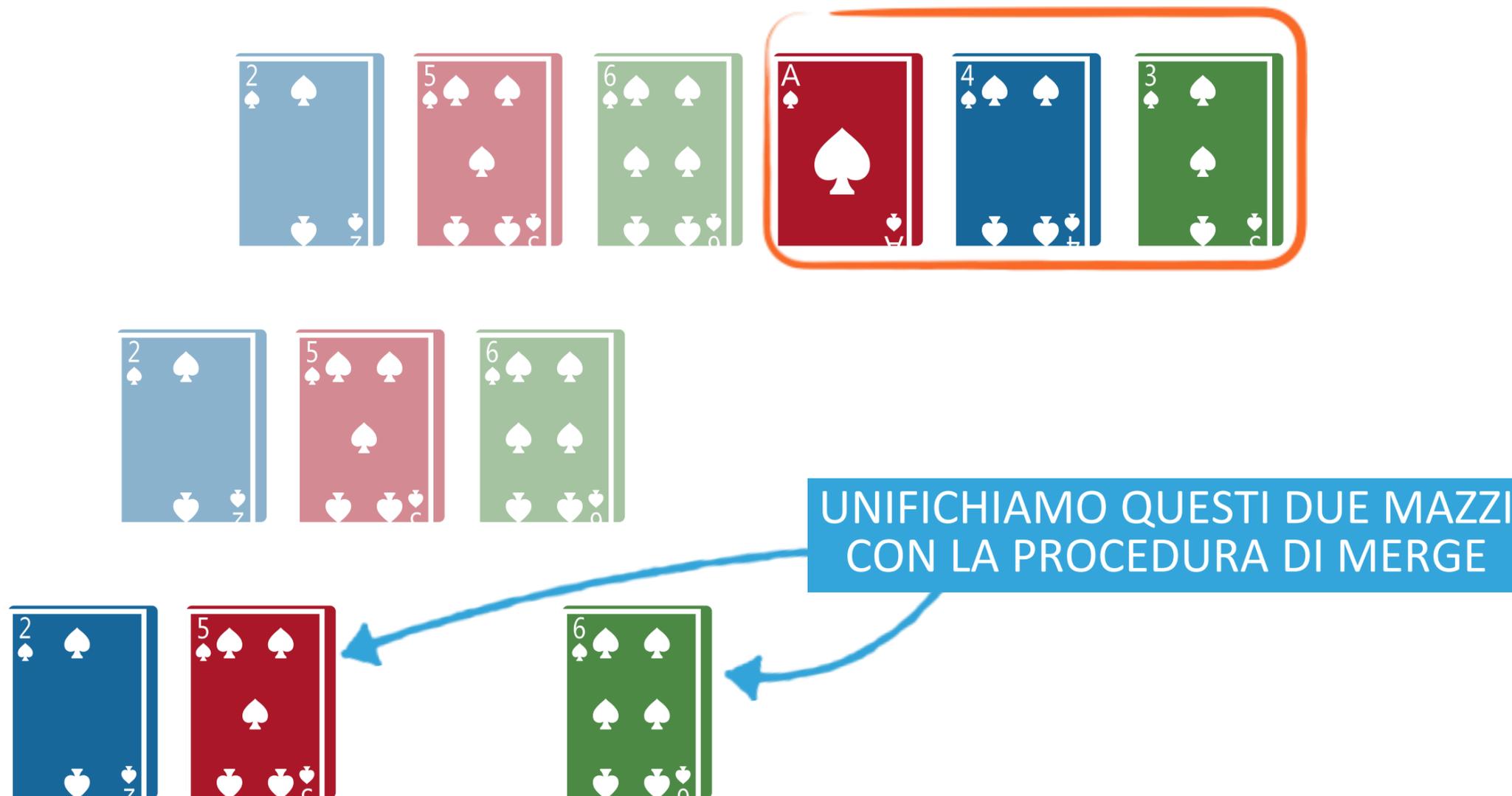


MERGESORT: ESEMPIO

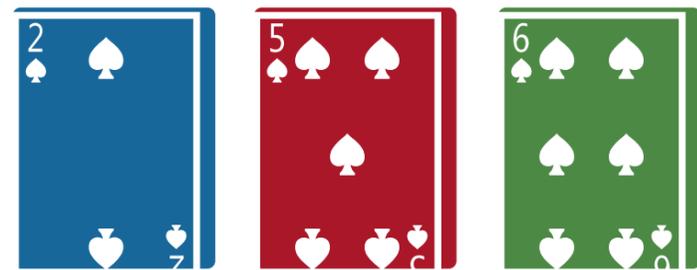


QUESTO MAZZO È COMPOSTO
DA UNA CARTA SOLA
E QUINDI È ORDINATO

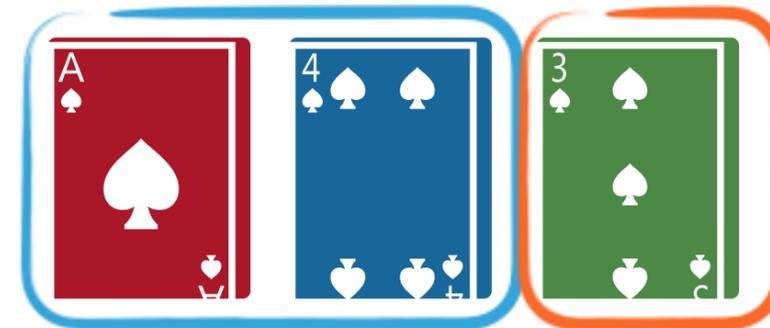
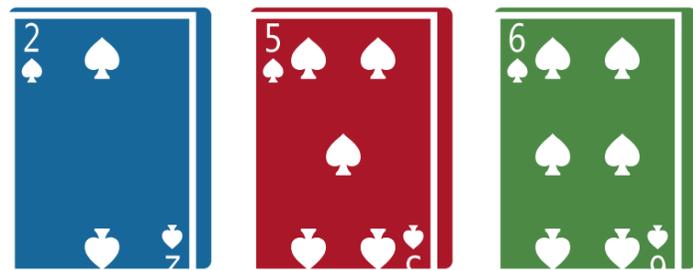
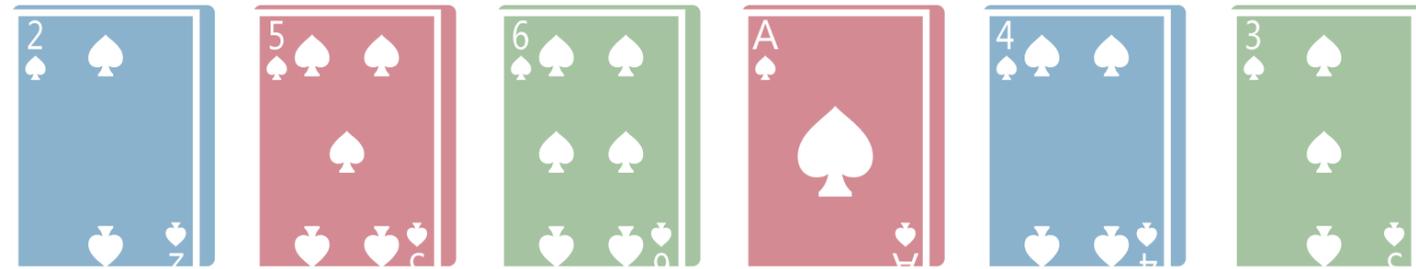
MERGESORT: ESEMPIO



MERGESORT: ESEMPIO

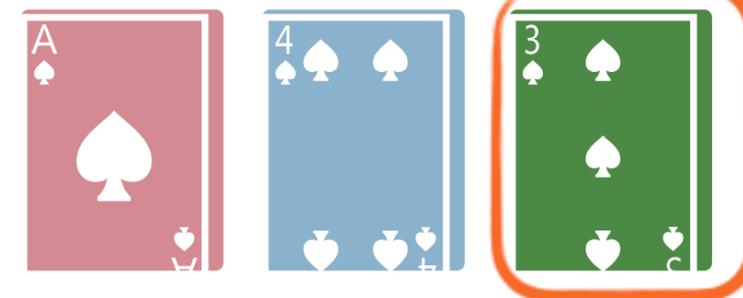
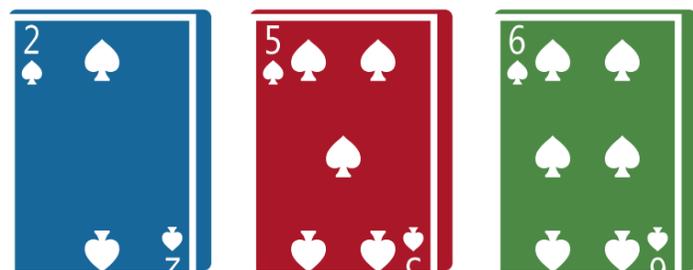
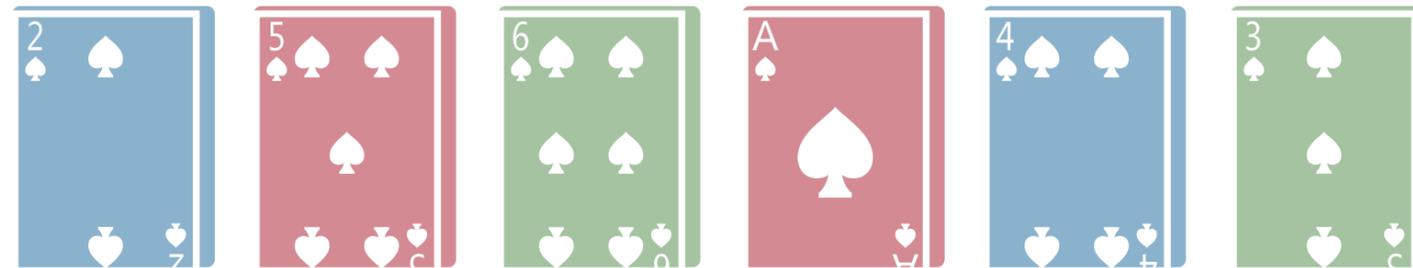


MERGESORT: ESEMPIO



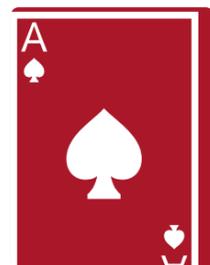
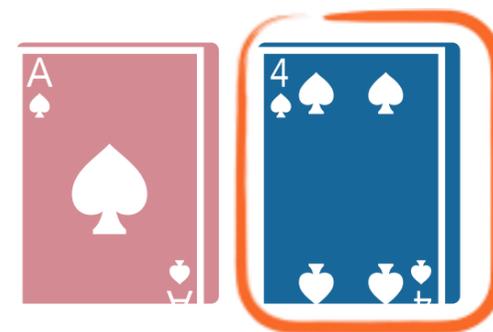
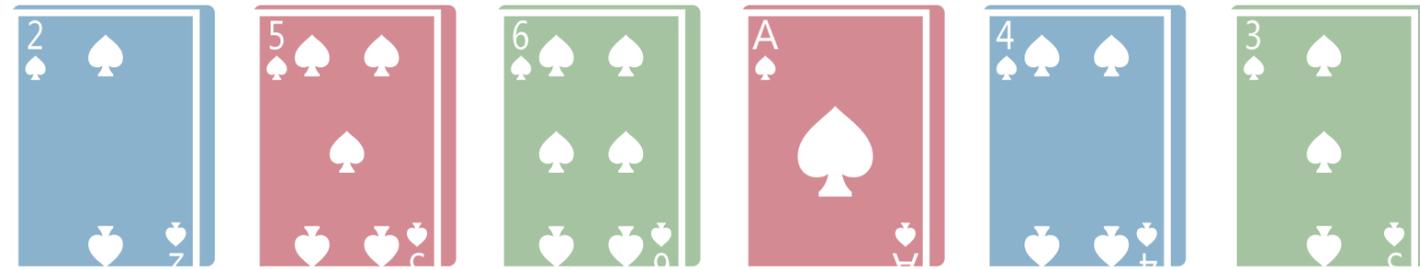
SPEZZARE IL MAZZO

MERGESORT: ESEMPIO

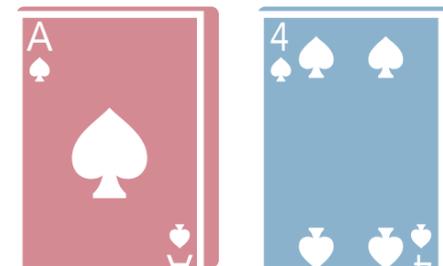
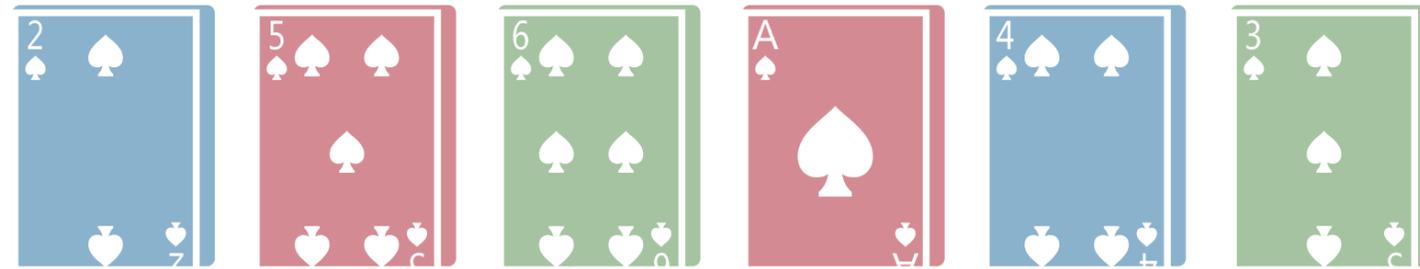


SPEZZARE IL MAZZO

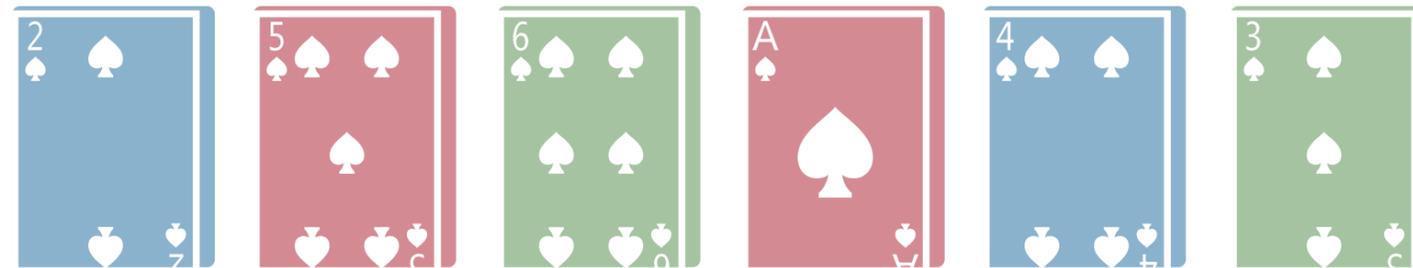
MERGESORT: ESEMPIO



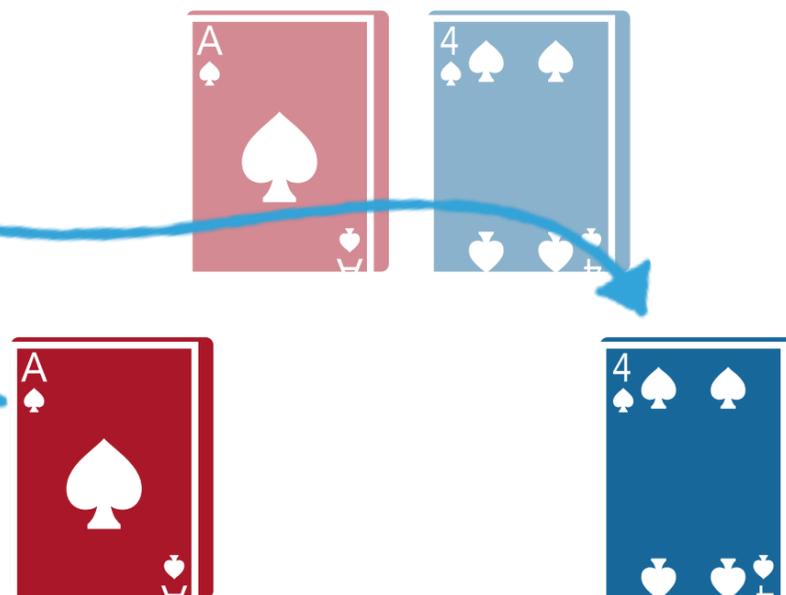
MERGESORT: ESEMPIO



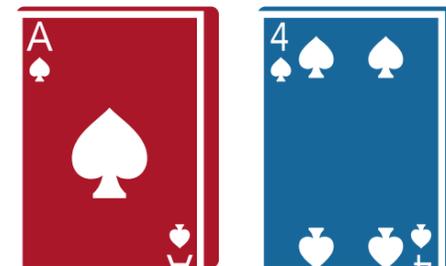
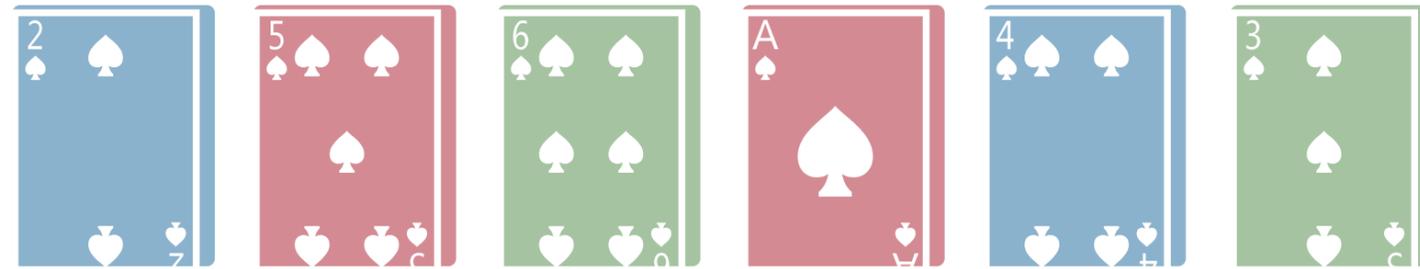
MERGESORT: ESEMPIO



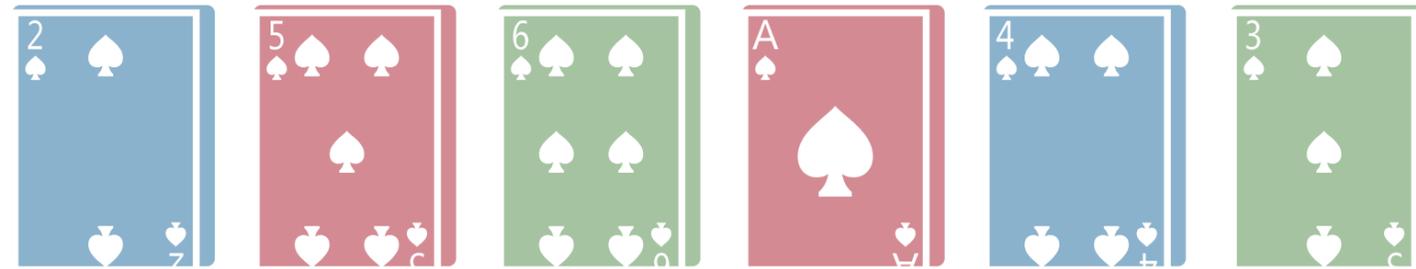
UNIFICHIAMO QUESTI DUE MAZZI
CON LA PROCEDURA DI MERGE



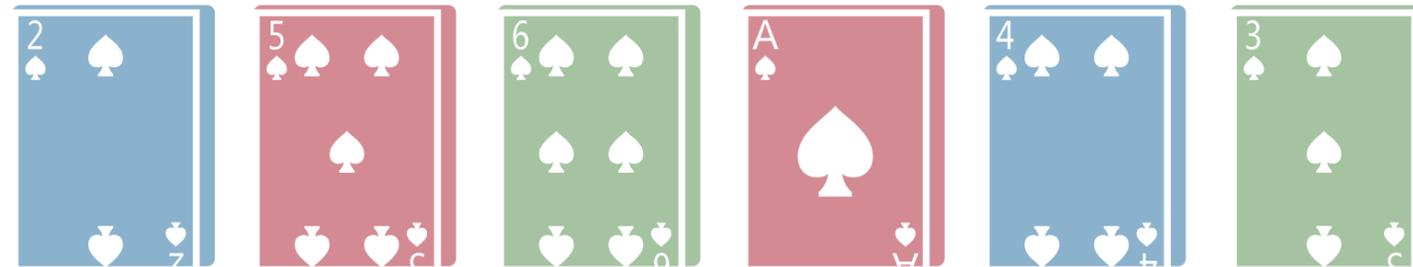
MERGESORT: ESEMPIO



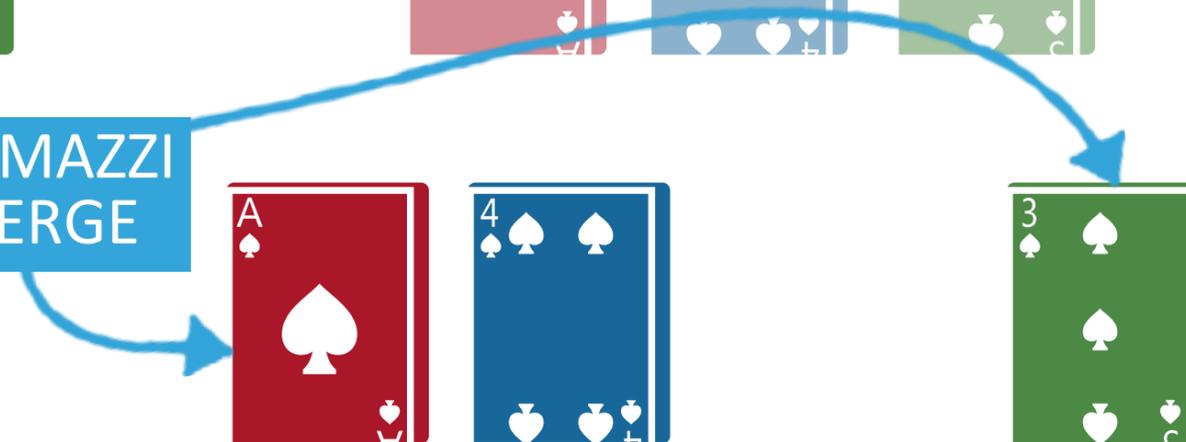
MERGESORT: ESEMPIO



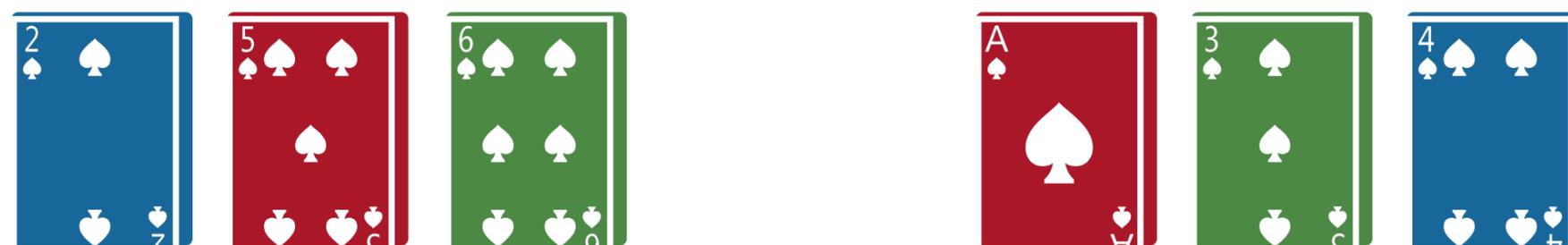
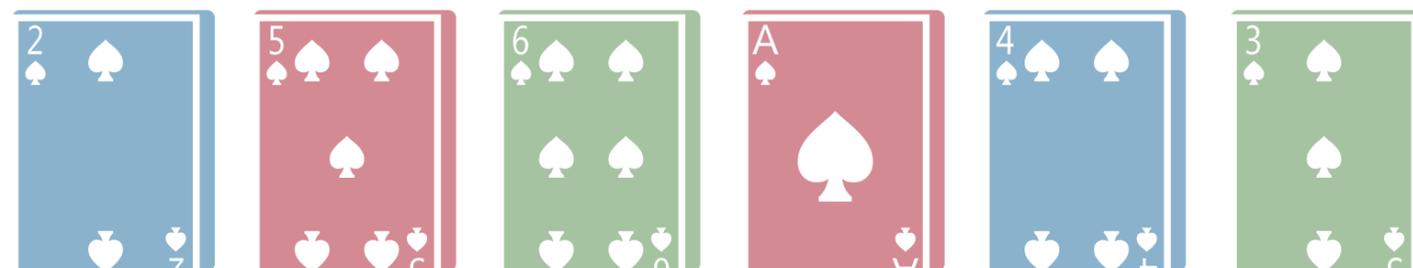
MERGESORT: ESEMPIO



UNIFICHIAMO QUESTI DUE MAZZI
CON LA PROCEDURA DI MERGE

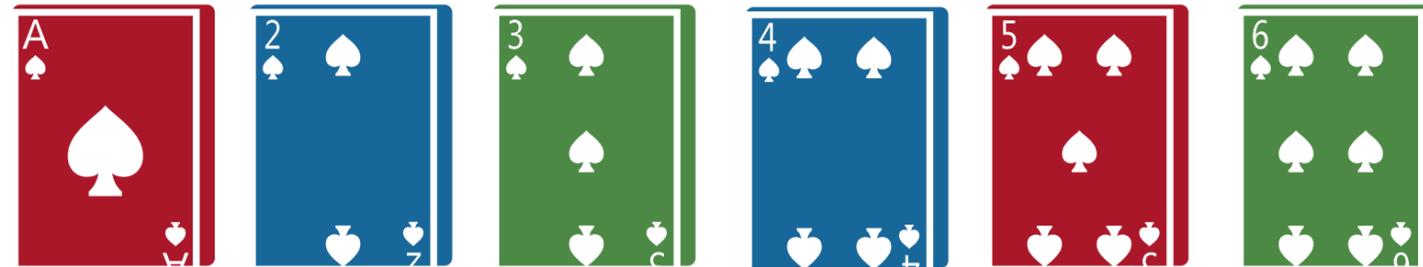


MERGESORT: ESEMPIO



UNIFICHIAMO QUESTI DUE MAZZI
CON LA PROCEDURA DI MERGE

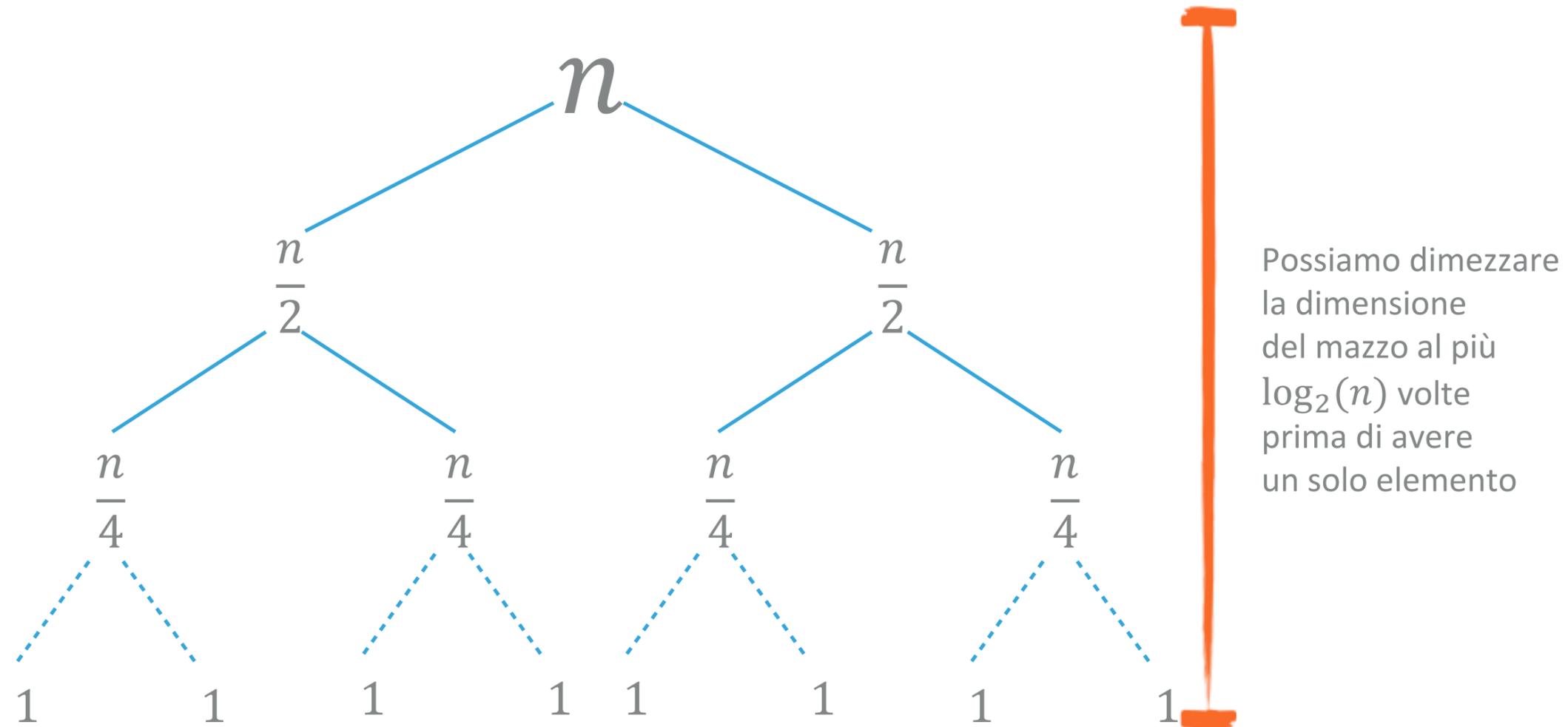
MERGESORT: ESEMPIO



Abbiamo ottenuto il mazzo ordinato!

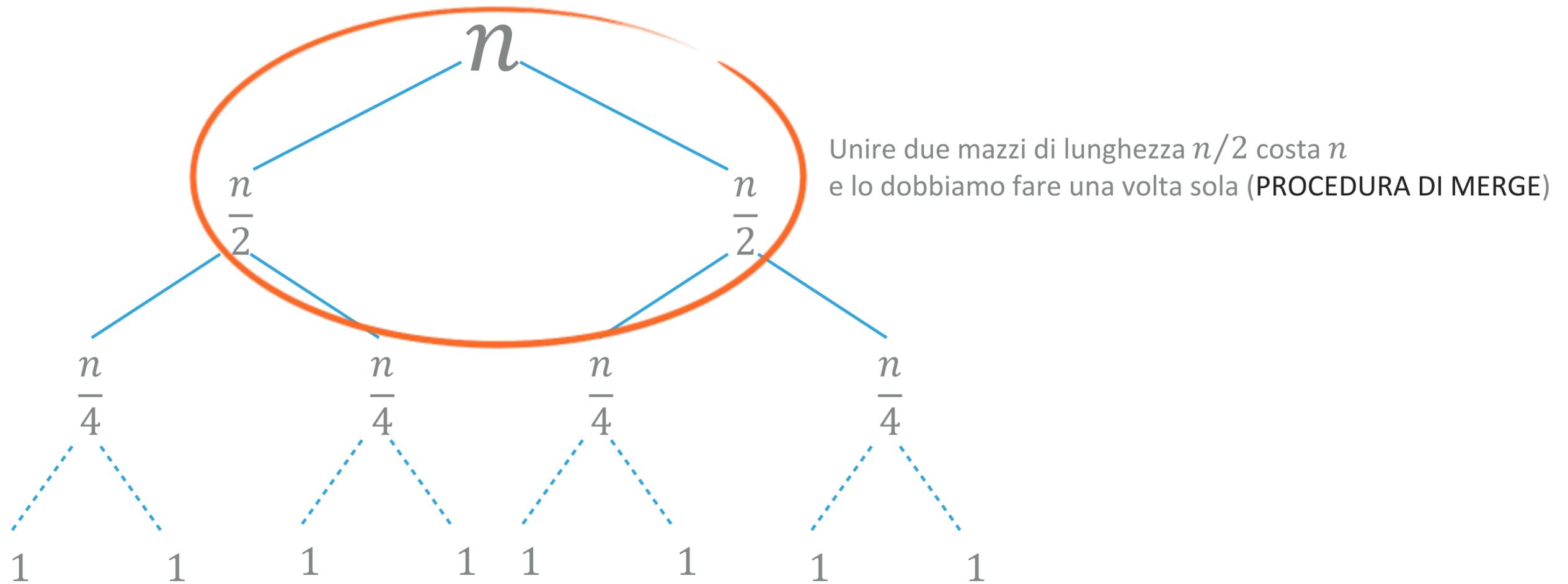
Ma quale è il costo in numero di comparazioni?

CALCOLO DEL COSTO

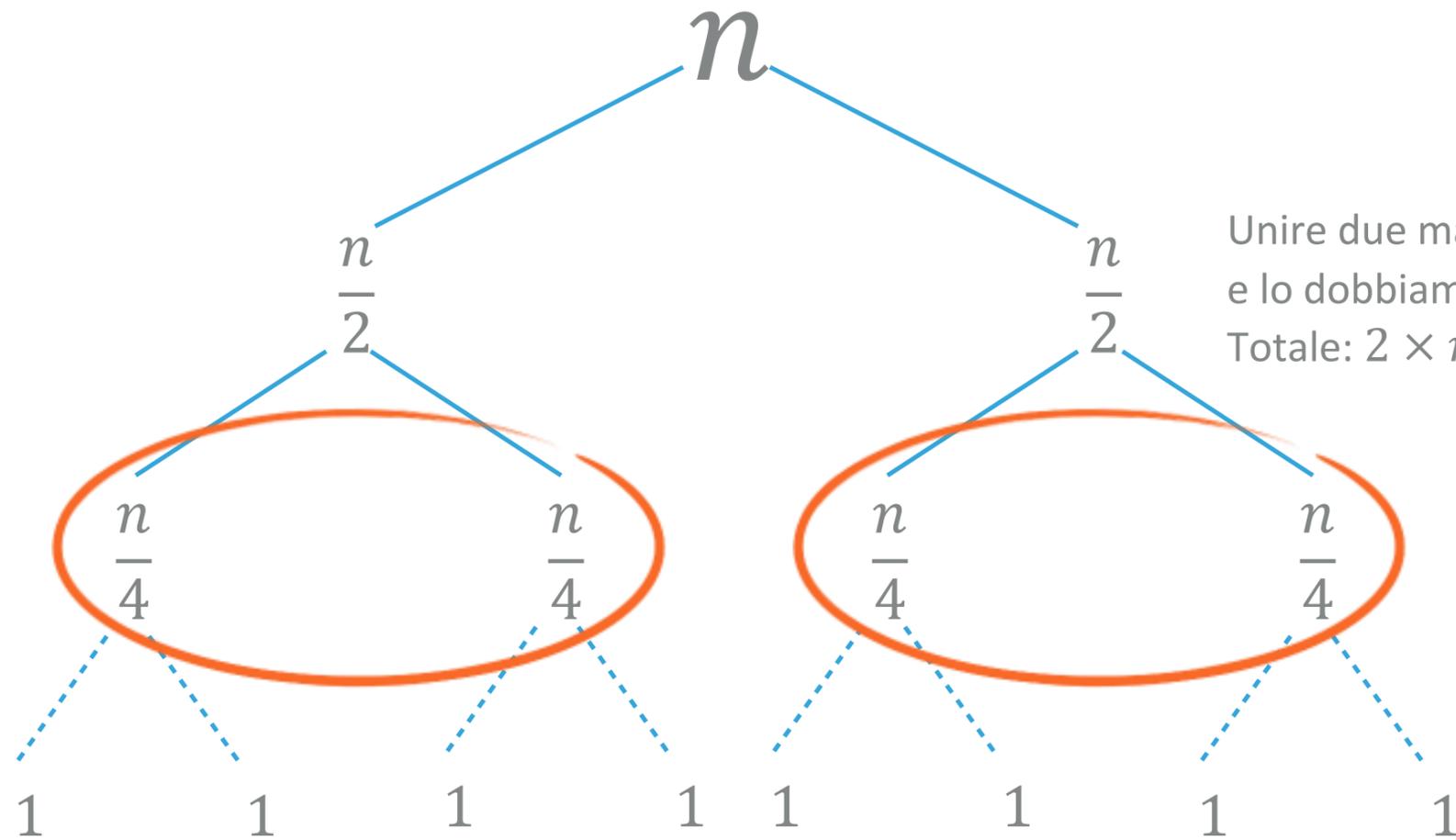


Nota. Questo particolare albero si chiama «albero di ricorsione».

CALCOLO DEL COSTO

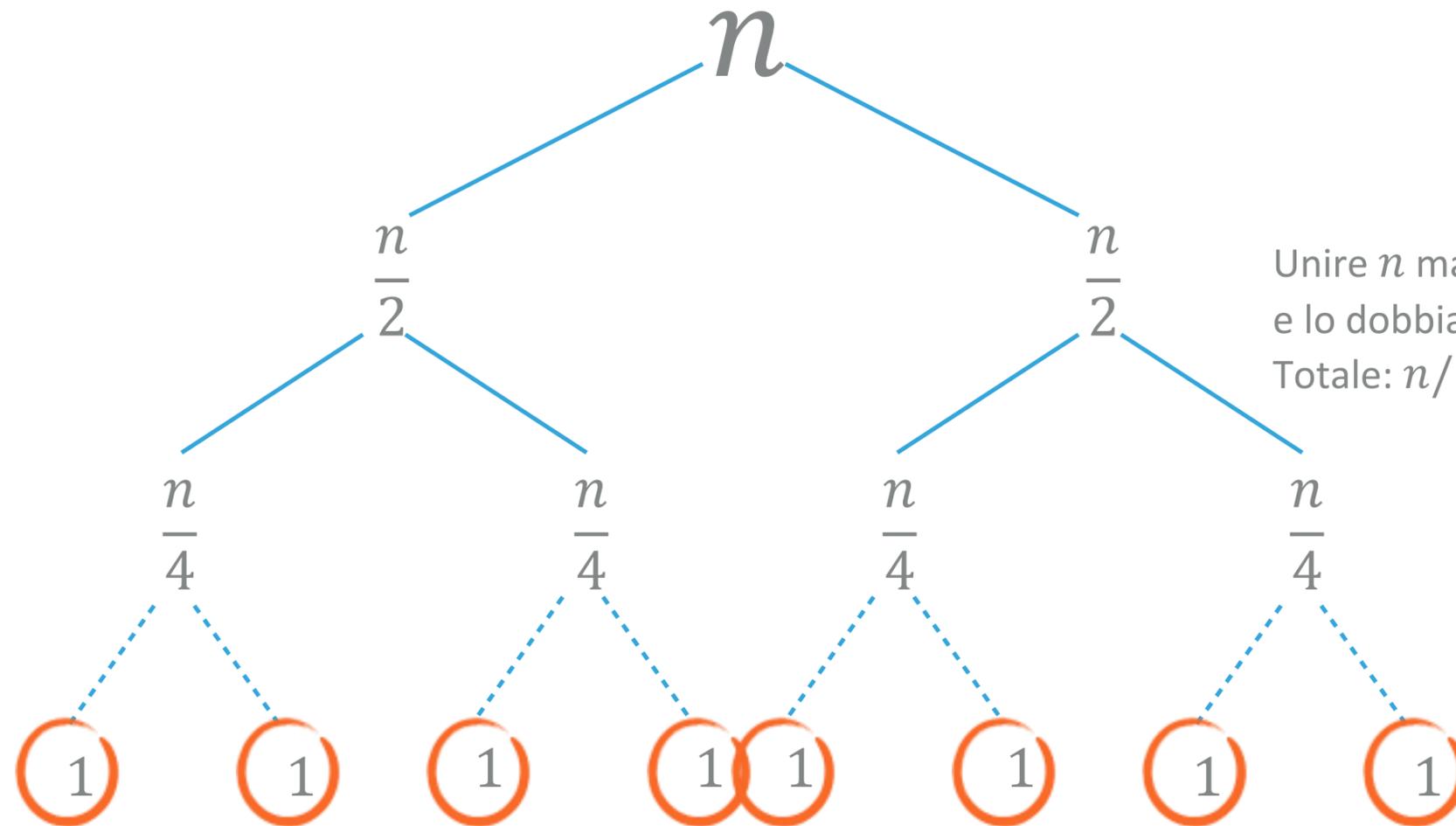


CALCOLO DEL COSTO



Unire due mazzi di lunghezza $n/4$ costa $n/2$
e lo dobbiamo fare 2 volte.
Totale: $2 \times n/2 = n$

CALCOLO DEL COSTO



Unire n mazzi di lunghezza 1 costa 2
e lo dobbiamo fare $n/2$ volte.
Totale: $n/2 \times 2 = n$

PSEUDOCODICE: PROCEDURA MERGE (1/2)

MERGE (A, p, q, r)

$n_L = q - p - 1$ //lunghezza della sottosequenza $L = A[p:q]$

$n_R = r - q$ //lunghezza della sottosequenza $R = A[q+1:r]$

//si creano due nuovi array: $L[0:n_L-1]$ ed $R[0:n_R-1]$

for $i=0$ to $n_L - 1$

$L[i] = A[p+i]$ //copia $A[p:q]$ in $L[0:n_L-1]$

for $j=0$ to $n_R - 1$

$R[j] = A[q+1+j]$ //copia $A[q+1:r]$ in $R[0:n_R-1]$

PSEUDOCODICE: PROCEDURA MERGE (2/2)

```
i=0      //indice che scorre su L
j=0      //indice che scorre su R
k=0      //indice che scorre su A
while i ≤ nL & j ≤ nR //finchè le due sottosequenze non sono vuote
    if L[i] ≤ R[j]
        A[k] = L[i]
        i = i+1
    else
        A[k] = R[j]
        j = j+1
    k = k+1
while i ≤ nL //finisco di copiare il resto di L in A
    A[k] = L[i]
    i = i+1
    k = k+1
while j ≤ nR //finisco di copiare il resto di R in A
    A[k] = R[j]
    j = j+1
    k = k+1
```

Nota. C'è un typo sul libro (j=1 invece che j=0).

PSEUDOCODICE: ALGORITMO MERGESORT COMPLETO

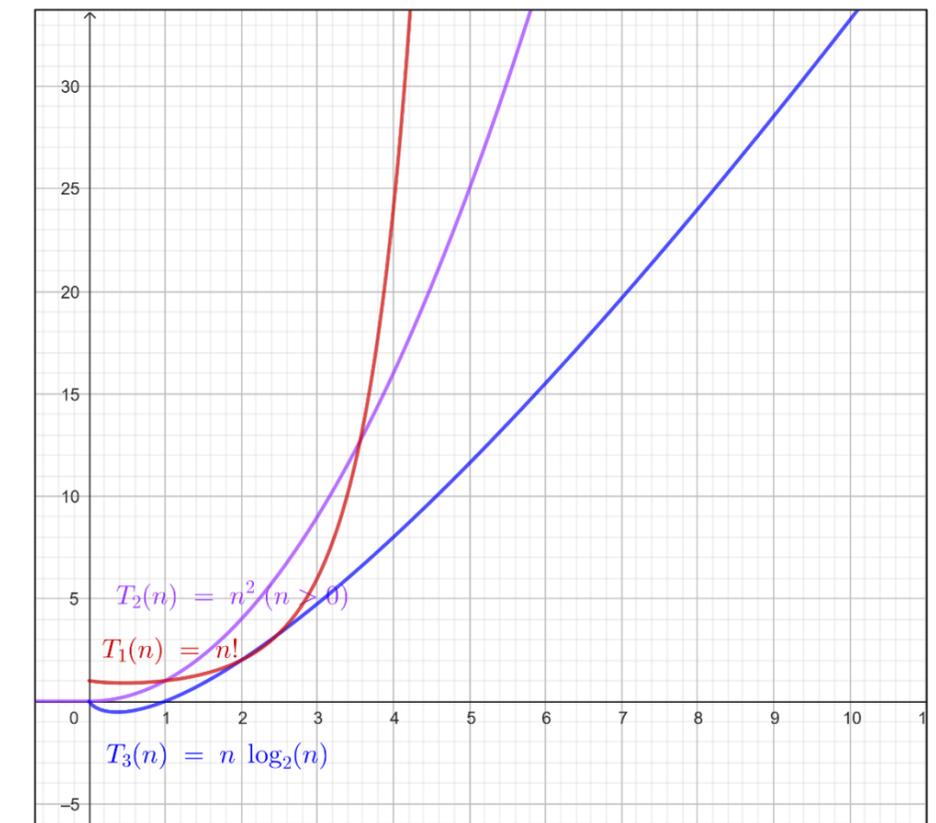
```
MERGE-SORT (A, p, r)
if p ≥ r //0 o 1 solo elemento (caso base)
    return
q = ⌊(p+r)/2⌋ //calcolo punto medio di A[p:r]
MERGE-SORT (A, p, q) //ordina ricorsivamente A[p:q]
MERGE-SORT (A, q+1, r) //ordina ricorsivamente A[q+1:r]
MERGE (A, p, q, r) //fonde A[p:q] con A[q+1:r] per ottenere A[p:r] ordinato
```

CALCOLO DEL COSTO

- ▶ Ogni “livello” ci costa al più n comparazioni.
- ▶ Abbiamo $\log_2(n)$ livelli.
- ▶ Il costo totale in termini di comparazioni è **$n \log_2 n$** .
- ▶ Abbiamo migliorato ancora il tempo necessario ad ordinare un mazzo di carte:

Algoritmo	$T(n)$
Bogosort	$n!$
Insertion sort	$n * n = n^2$
Mergesort	$n * \log_2 n$

bogosort insertion sort mergesort





**UNIVERSITÀ
DEGLI STUDI
DI TRIESTE**

Prossima lezione: 15 maggio, h.9:00, aula 4C