

INFORMAZIONI SUL CORSO
COSA È UN ALGORITMO
ESEMPI DI ALGORITMI

ALGORITMI E STRUTTURE DATI

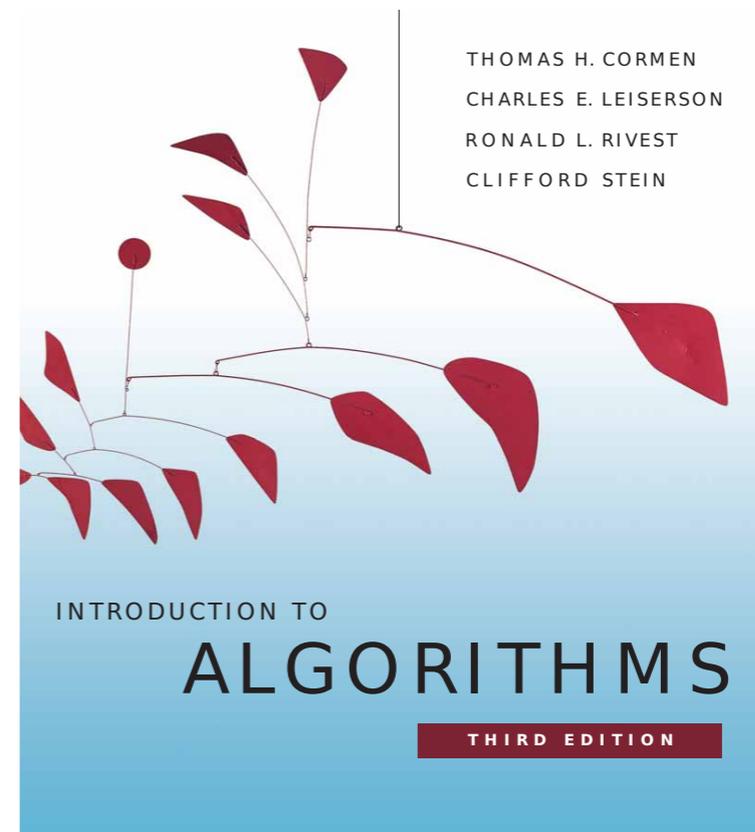
STRUTTURA DEL CORSO

- ▶ Introduzione alla progettazione di algoritmi (complessità, ricorsione, strutture dati)
- ▶ Ordinamento e selezione
- ▶ Strutture dati per l'indicizzazione (alberi binari di ricerca, hash)
- ▶ Strutture relazionali (grafi e algoritmi su grafi)

LIBRI E MATERIALE

- ▶ Cormen, Leiserson, Rivest, Stein
Introduction to Algorithms (Introduzione agli algoritmi)
3rd edition.
MIT Press (McGraw Hill per l'edizione italiana)

(È un libro grosso, con tanti argomenti oltre a quelli che vedremo).



SOFTWARE E MATERIALE DEL CORSO

- ▶ Il materiale del corso (slide ed esempi di codice) sarà disponibile online sul sito e-learning del corso
- ▶ Come software si utilizzerà Python versione 3.6 o successive
(disponibile su <https://www.python.org>)
 - ▶ Solitamente preinstallato su macOS e Linux
- ▶ È possibile usare un ambiente Python 3 online senza installazione su repl.it (<https://repl.it>)

ESAME E HOMEWORK

- ▶ L'esame avrà una parte scritta ed una orale, obbligatorie.
- ▶ **Scritto:** risoluzione di problemi tipo quelli visti durante il corso.
- ▶ **Orale:** domande sulla teoria e risoluzione di problemi, vale ± 5 punti.
- ▶ **Provette:** ci saranno due provette intermedie. Chi le supera entrambe può saltare lo scritto (vale la media delle provette come voto dello scritto).
- ▶ **Homework:** verranno dati settimanalmente alcuni esercizi per casa, sia problemi da risolvere che esercizi implementativi. Chi li consegna in tempo, in funzione del voto preso, avrà dei punti bonus allo scritto (massimo 3 punti, ma il passaggio da 29 a 30 costa due punti).
- ▶ La lode si può prendere solo con l'orale, di norma con un voto sullo scritto di almeno 27.

LET'S START!

Cos'è un algoritmo?

COSA È UN ALGORITMO? (DEFINIZIONE INFORMALE)

Una procedura computazionale ben definita che prende un valore o insieme di valori come **input** e produce un valore o insieme di valori come **output**

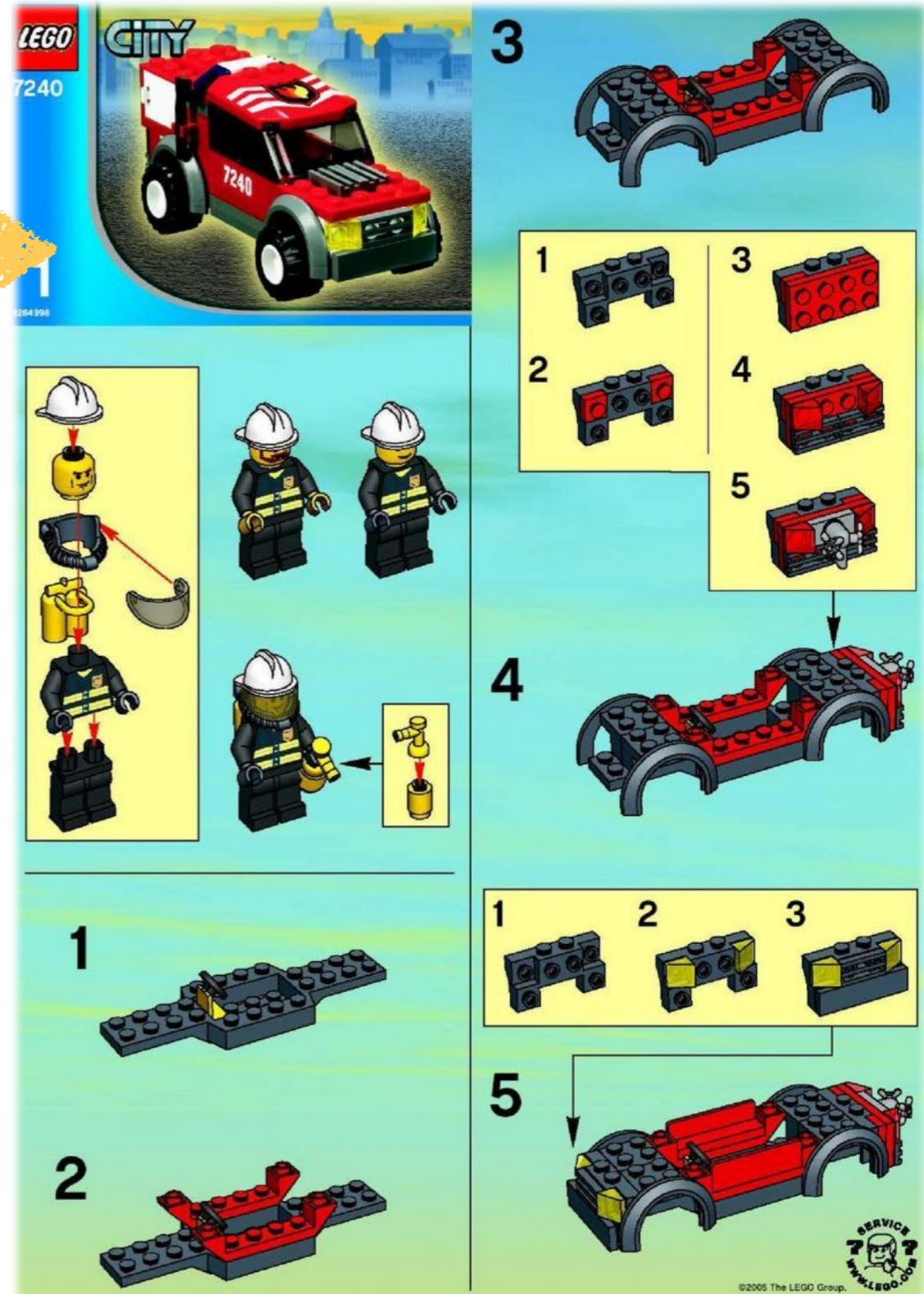
Un algoritmo è quindi un insieme di operazioni che trasformano un input in un output

ALCUNI ESEMPI COMUNI

Input: un insieme ben definito di pezzi di lego

Output:

Le istruzioni contengono un insieme di passi non ambigui che trasformano l'input nell'output



ALCUNI ESEMPI COMUNI

Input: un insieme ben definito di pezzi di ingredienti

Output: tartufi al tiramisù

Anche in questo caso le istruzioni contengono un insieme di passi non ambigui (più o meno) che trasformano l'input nell'output

Preparazione

COME FARE I TARTUFI AL TIRAMISÙ



Per preparare i tartufini al tiramisù iniziamo ad inserire i Pavesini nel mixer [1] e a tritarli, fino ad ottenere una polvere fine. [2]



A parte lavoriamo il mascarpone con lo zucchero, aiutandoci con le fruste elettriche. [3]
Aggiungiamo la polvere di Pavesini ed amalgamiamo con una spatola a mano. [4]

ORDINARE UN MAZZO DI CARTE



Abbiamo un mazzo di carte che è stato mescolato

Vogliamo disporre le carte in ordine crescente

Siamo in grado di dare un **algoritmo**?

Iniziamo a formalizzare il problema

ORDINARE UN MAZZO DI CARTE



Dati in ingresso (**input**)

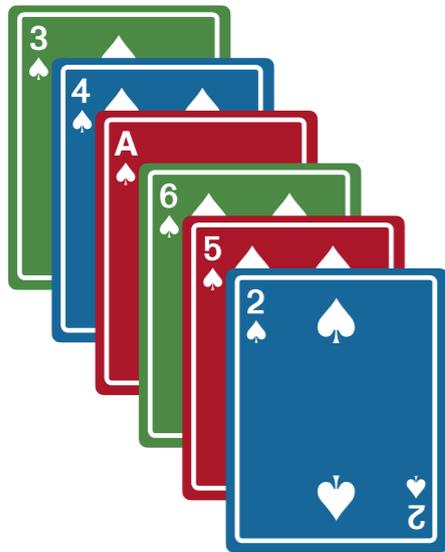
Un mazzo di carte non necessariamente ordinate



Dati in uscita (**output**)

Una permutazione del mazzo di carte fornito in input
in cui tutte le carte sono ordinate in ordine crescente

ORDINARE UN MAZZO DI CARTE



Dati in ingresso (**input**)

Un mazzo di carte non necessariamente ordinate

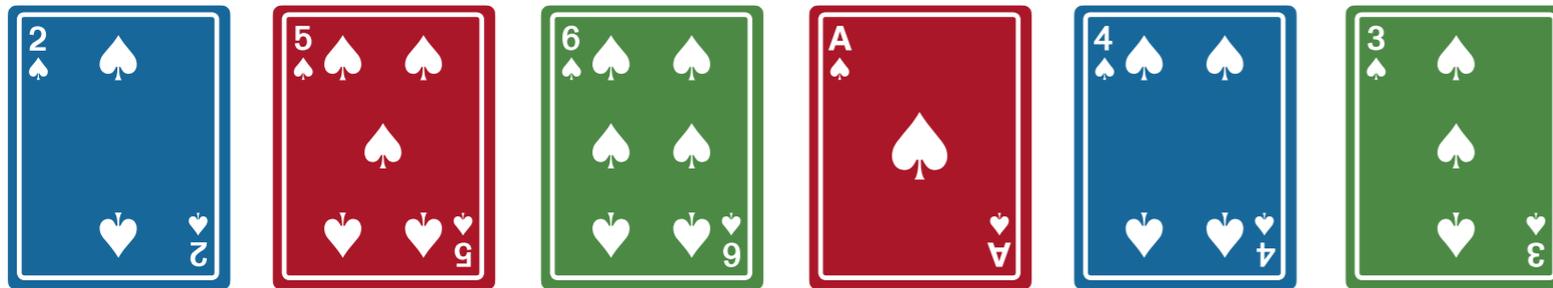
**CI SERVE UNA PROCEDURA MECCANICA
CHE DALL'INPUT PRODUCA L'OUTPUT**



Dati in uscita (**output**)

Una permutazione del mazzo di carte fornito in input
in cui tutte le carte sono ordinate in ordine crescente

IDEE?



IDEA: INSERIMENTO DI UNA CARTA ALLA VOLTA



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

OPZIONE 2: INSERIMENTO DI UNA CARTA ALLA VOLTA



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

OPZIONE 2: INSERIMENTO DI UNA CARTA ALLA VOLTA



1. Inizialmente non si hanno carte ordinate

→ 2. Finché ci rimangono carte non ordinate

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

2.2. Metti X in fondo alla lista di carte ordinate

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

OPZIONE 2: INSERIMENTO DI UNA CARTA ALLA VOLTA



1. Inizialmente non si hanno carte ordinate

→ 2. Finché ci rimangono carte non ordinate

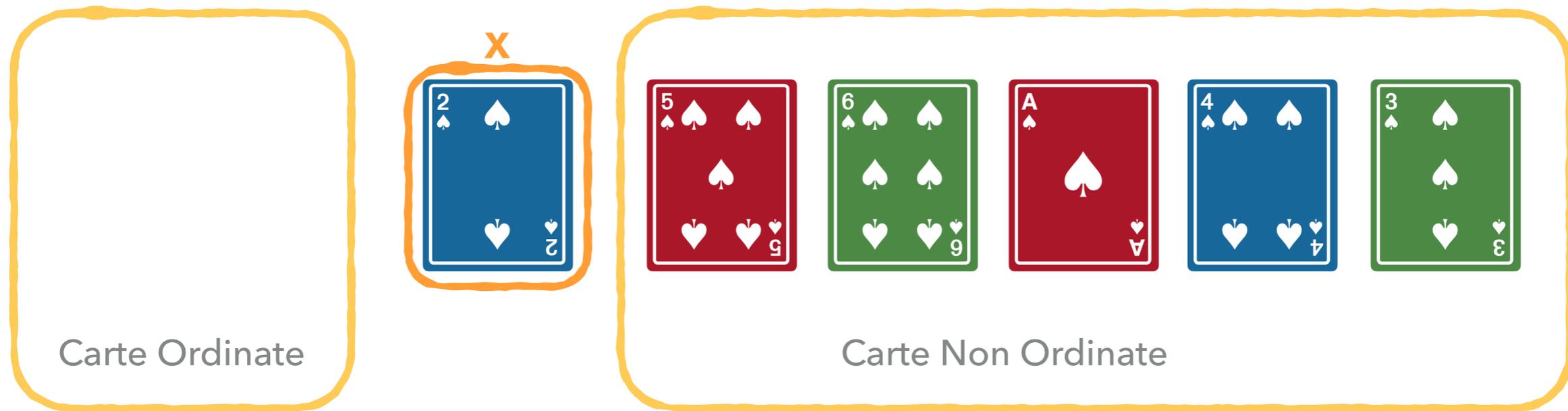
**CONTINUEREMO AD ESEGUIRE
QUESTE ISTRUZIONI
FINCHÉ LA CONDIZIONE
NON SARÀ SODDISFATTA**

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

2.2. Metti X in fondo alla lista di carte ordinate

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

OPZIONE 2: INSERIMENTO DI UNA CARTA ALLA VOLTA



1. Inizialmente non si hanno carte ordinate

2. Finché ci rimangono carte non ordinate

→ 2.1. Prendi una carta dalla lista di carte non ordinata e chiamala X

2.2. Metti X in fondo alla lista di carte ordinate

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

OPZIONE 2: INSERIMENTO DI UNA CARTA ALLA VOLTA



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

OPZIONE 2: INSERIMENTO DI UNA CARTA ALLA VOLTA



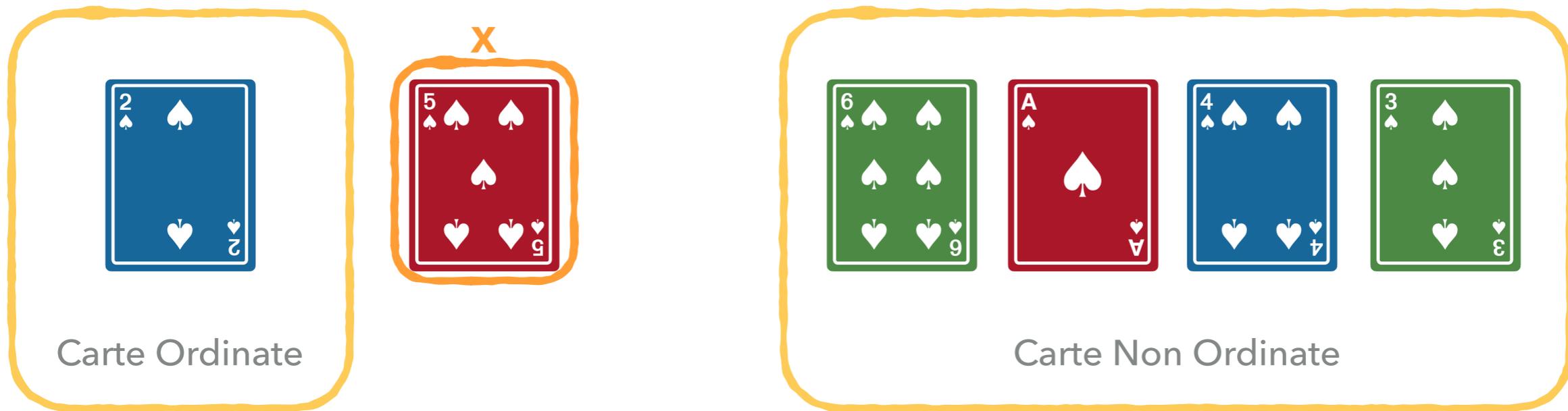
1. Inizialmente non si hanno carte ordinate
2. Finché ci rimangono carte non ordinate

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

- 2.1. Prendi una carta dalla lista di carte non ordinata e chiamala X
- 2.2. Metti X in fondo alla lista di carte ordinate

→ 2.3. Finché la carta che precede X ha valore maggiore del valore di X, scambia X con la carta che lo precede

OPZIONE 2: INSERIMENTO DI UNA CARTA ALLA VOLTA



1. Inizialmente non si hanno carte ordinate

2. Finché ci rimangono carte non ordinate

→ 2.1. Prendi una carta dalla lista di carte non ordinata e chiamala X

2.2. Metti X in fondo alla lista di carte ordinate

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

OPZIONE 2: INSERIMENTO DI UNA CARTA ALLA VOLTA



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

OPZIONE 2: INSERIMENTO DI UNA CARTA ALLA VOLTA



$2 > 5?$ NO



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

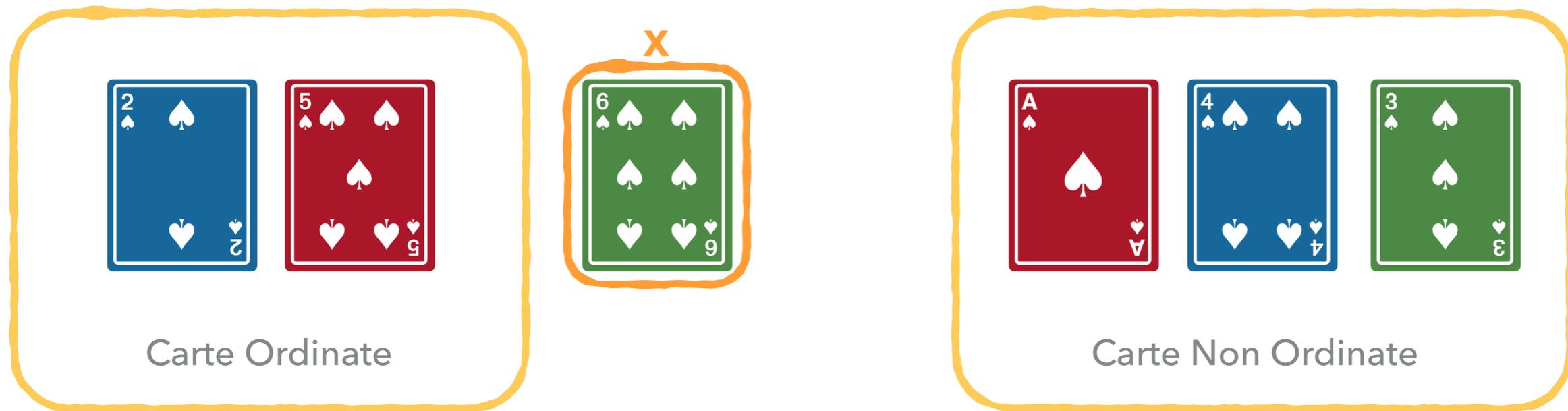
**IL TEST FALLISCE
E QUINDI PROSEGUIREMO
CON L'ISTRUZIONE NUMERO...**

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

2.2. Metti X in fondo alla lista di carte ordinate

→ 2.3. Finché la carta che precede X ha valore maggiore del valore di X, scambia X con la carta che lo precede

OPZIONE 2: INSERIMENTO DI UNA CARTA ALLA VOLTA



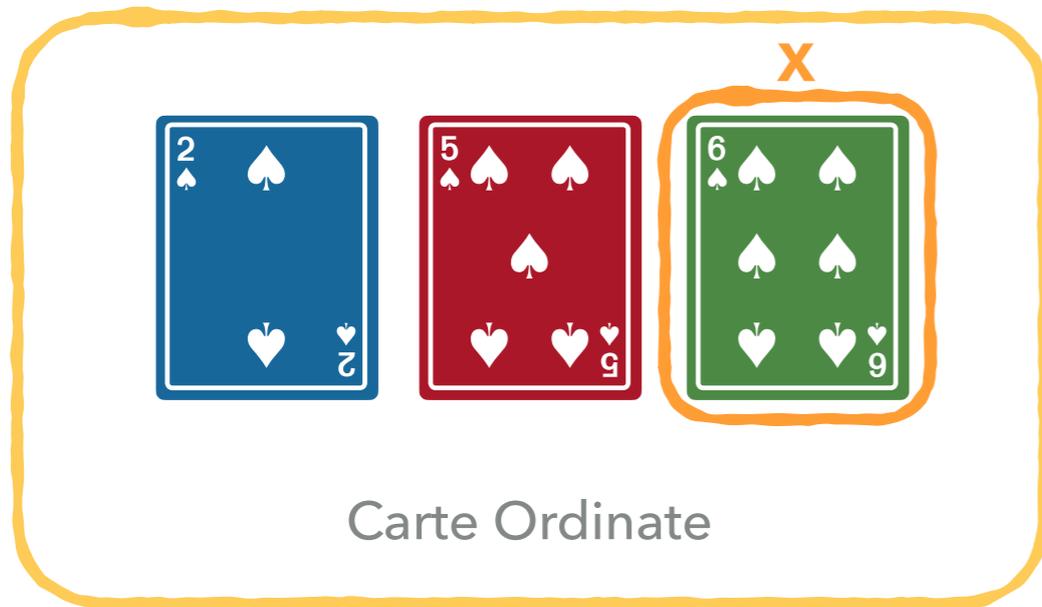
1. Inizialmente non si hanno carte ordinate
2. Finché ci rimangono carte non ordinate

→ 2.1. Prendi una carta dalla lista di carte non ordinata e chiamala X

2.2. Metti X in fondo alla lista di carte ordinate

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

OPZIONE 2: INSERIMENTO DI UNA CARTA ALLA VOLTA



1. Inizialmente non si hanno carte ordinate

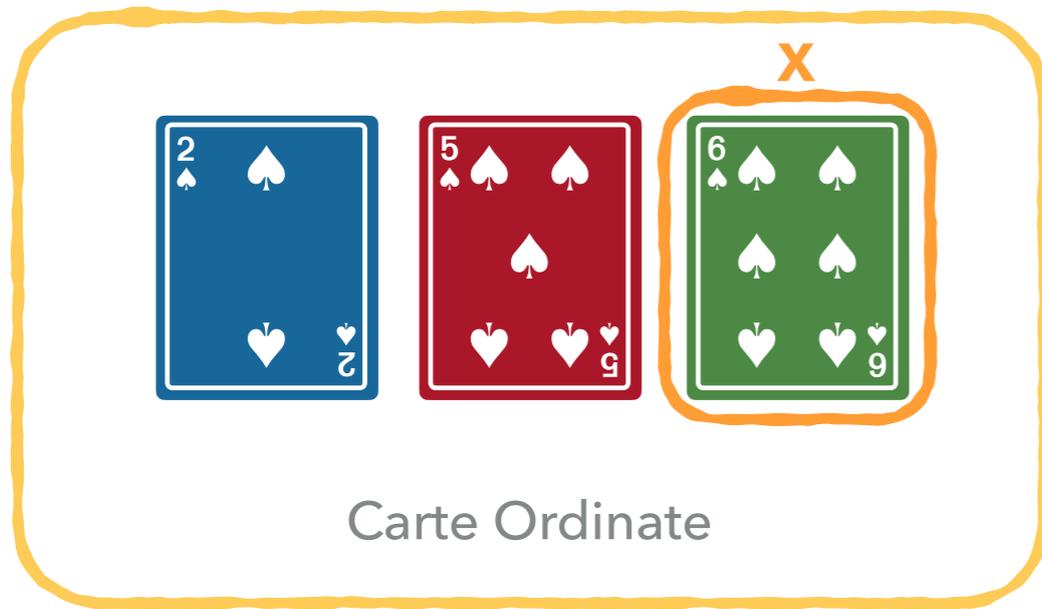
2. Finché ci rimangono carte non ordinate

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

→ 2.2. Metti X in fondo alla lista di carte ordinate

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

OPZIONE 2: INSERIMENTO DI UNA CARTA ALLA VOLTA



5 > 6? NO



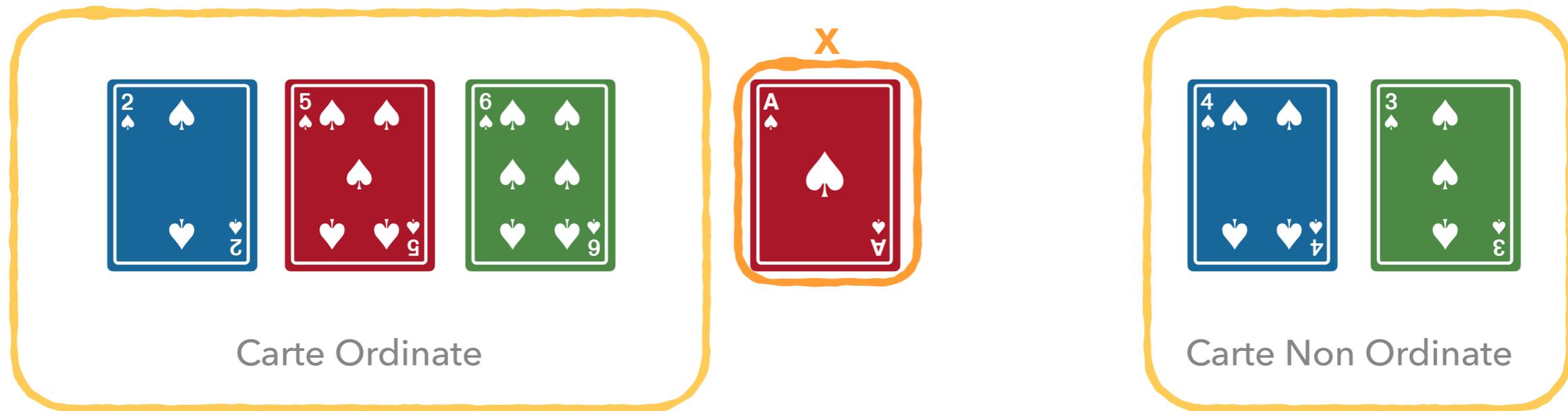
1. Inizialmente non si hanno carte ordinate
2. Finché ci rimangono carte non ordinate

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

2.2. Metti X in fondo alla lista di carte ordinate

→ 2.3. Finché la carta che precede X ha valore maggiore del valore di X, scambia X con la carta che lo precede

OPZIONE 2: INSERIMENTO DI UNA CARTA ALLA VOLTA



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

→ 2.1. Prendi una carta dalla lista di carte non ordinata e chiamala X

2.2. Metti X in fondo alla lista di carte ordinate

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

OPZIONE 2: INSERIMENTO DI UNA CARTA ALLA VOLTA



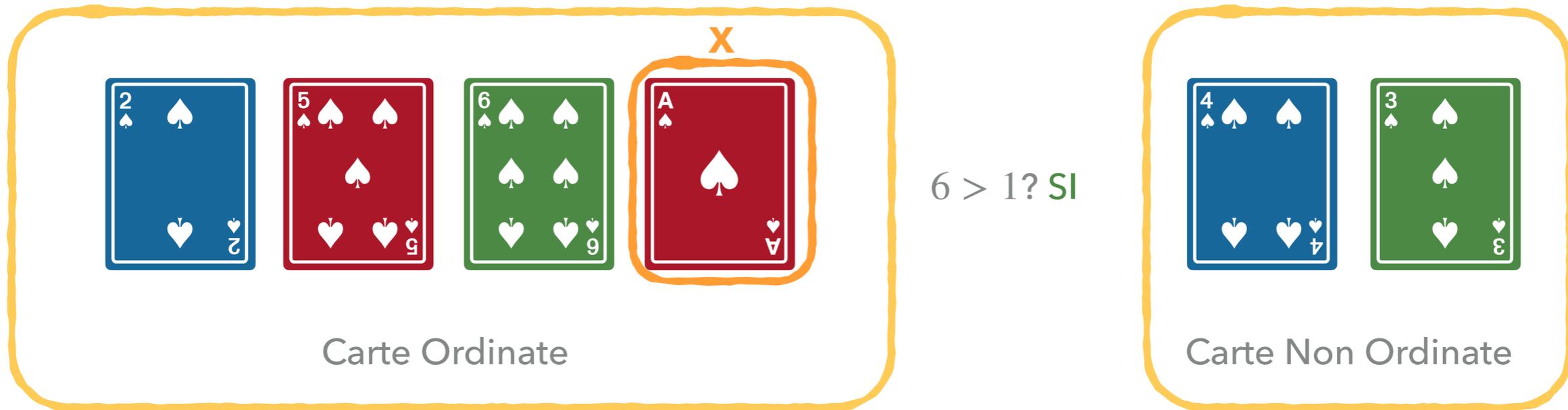
1. Inizialmente non si hanno carte ordinate
2. Finché ci rimangono carte non ordinate

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

→ 2.2. Metti X in fondo alla lista di carte ordinate

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

OPZIONE 2: INSERIMENTO DI UNA CARTA ALLA VOLTA



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

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

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

2.2. Metti X in fondo alla lista di carte ordinate

→ 2.3. Finché la carta che precede X ha valore maggiore del valore di X,
scambia X con la carta che lo precede

OPZIONE 2: INSERIMENTO DI UNA CARTA ALLA VOLTA



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

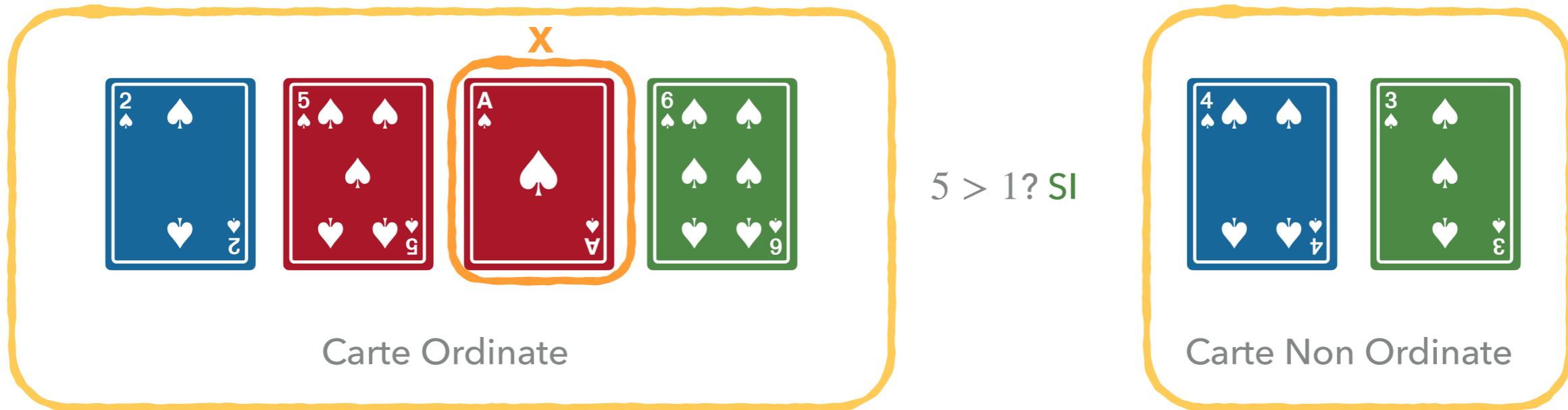
**ORA CHE DOBBIAMO FARE?
CONTINUIAMO CON L'ISTRUZIONE 2.3
O TORNIAMO ALLA 2.1?**

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

2.2. Metti X in fondo alla lista di carte ordinate

→ 2.3. Finché la carta che precede X ha valore maggiore del valore di X, scambia X con la carta che lo precede

OPZIONE 2: INSERIMENTO DI UNA CARTA ALLA VOLTA



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

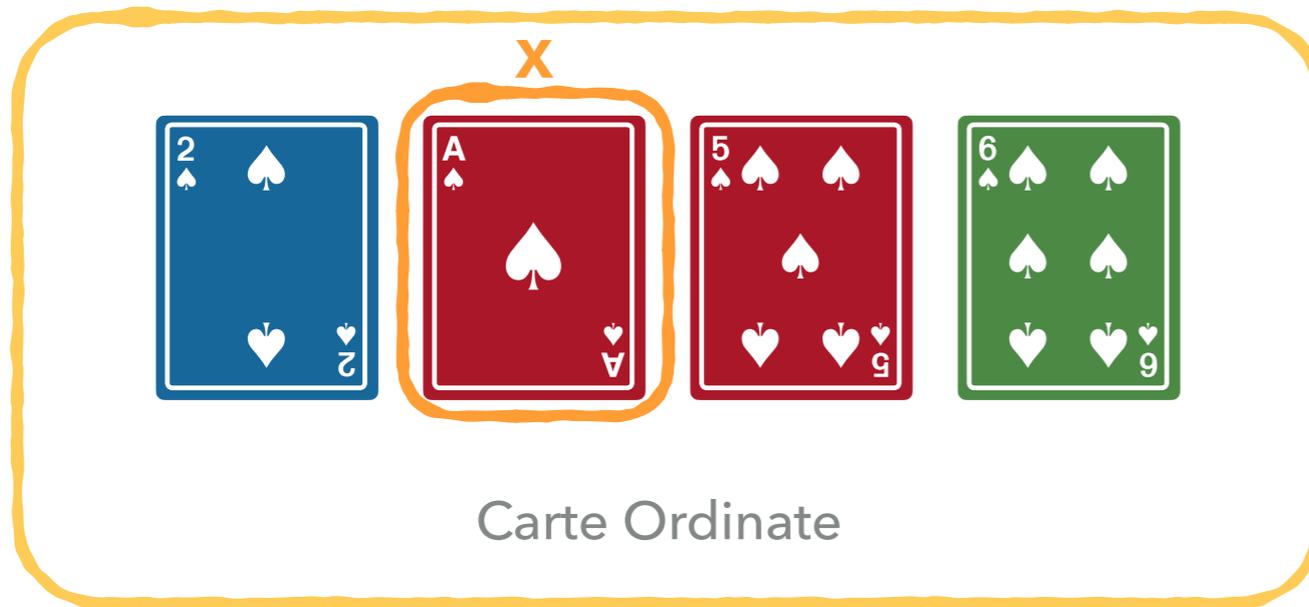
**ORA CHE DOBBIAMO FARE?
CONTINUIAMO CON L'ISTRUZIONE 2.3
O TORNIAMO ALLA 2.1?**

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

2.2. Metti X in fondo alla lista di carte ordinate

→ 2.3. Finché la carta che precede X ha valore maggiore del valore di X, scambia X con la carta che lo precede

OPZIONE 2: INSERIMENTO DI UNA CARTA ALLA VOLTA



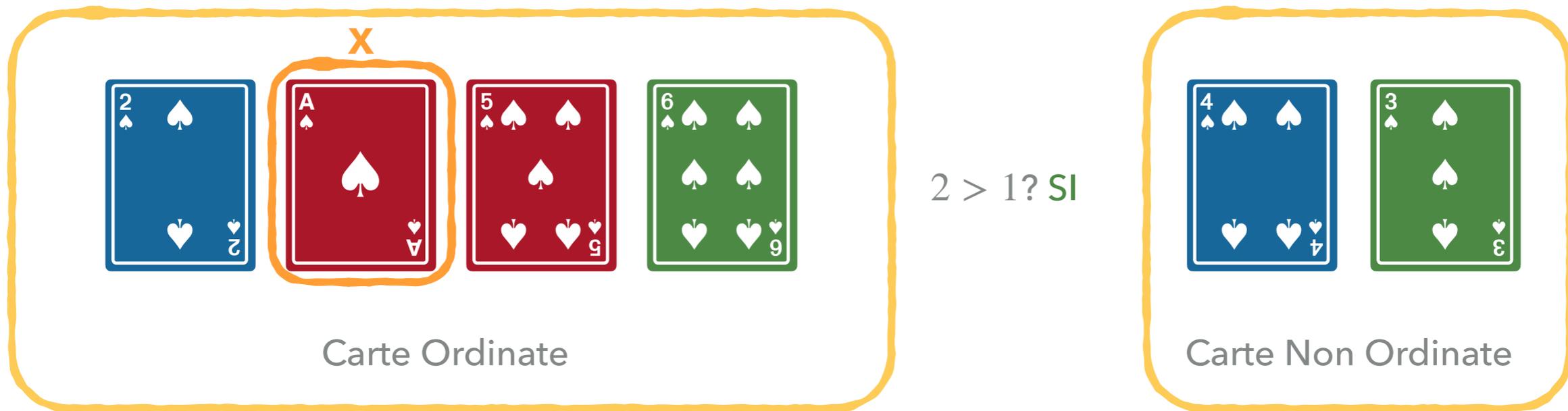
1. Inizialmente non si hanno carte ordinate
2. Finché ci rimangono carte non ordinate

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

2.2. Metti X in fondo alla lista di carte ordinate

→ 2.3. Finché la carta che precede X ha valore maggiore del valore di X, scambia X con la carta che lo precede

OPZIONE 2: INSERIMENTO DI UNA CARTA ALLA VOLTA



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

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

2.2. Metti X in fondo alla lista di carte ordinate

→ 2.3. Finché la carta che precede X ha valore maggiore del valore di X, scambia X con la carta che lo precede

OPZIONE 2: INSERIMENTO DI UNA CARTA ALLA VOLTA



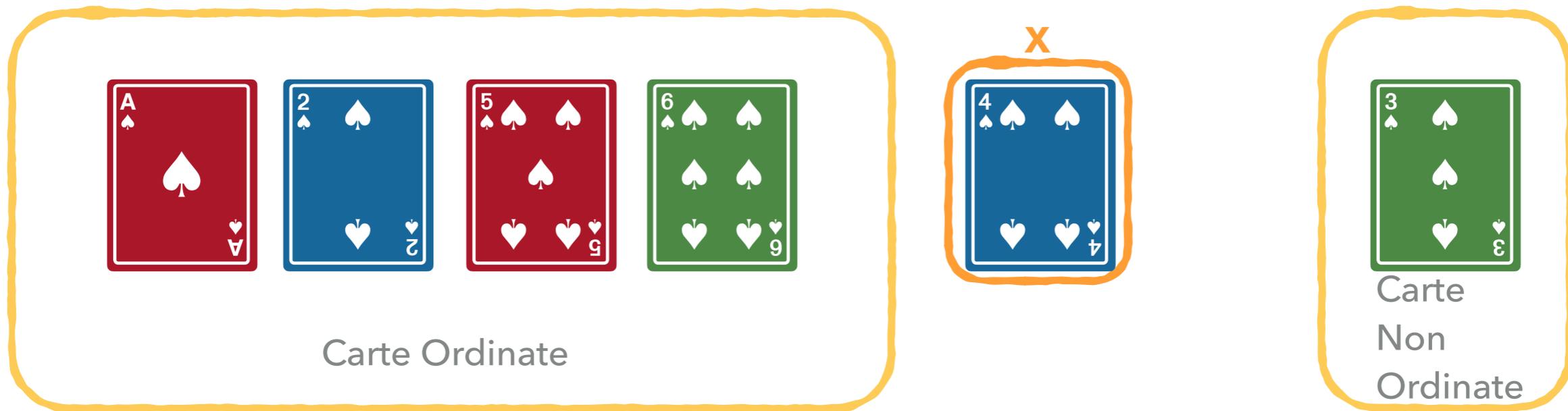
1. Inizialmente non si hanno carte ordinate
2. Finché ci rimangono carte non ordinate

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

2.2. Metti X in fondo alla lista di carte ordinate

→ 2.3. Finché la carta che precede X ha valore maggiore del valore di X, scambia X con la carta che lo precede

OPZIONE 2: INSERIMENTO DI UNA CARTA ALLA VOLTA



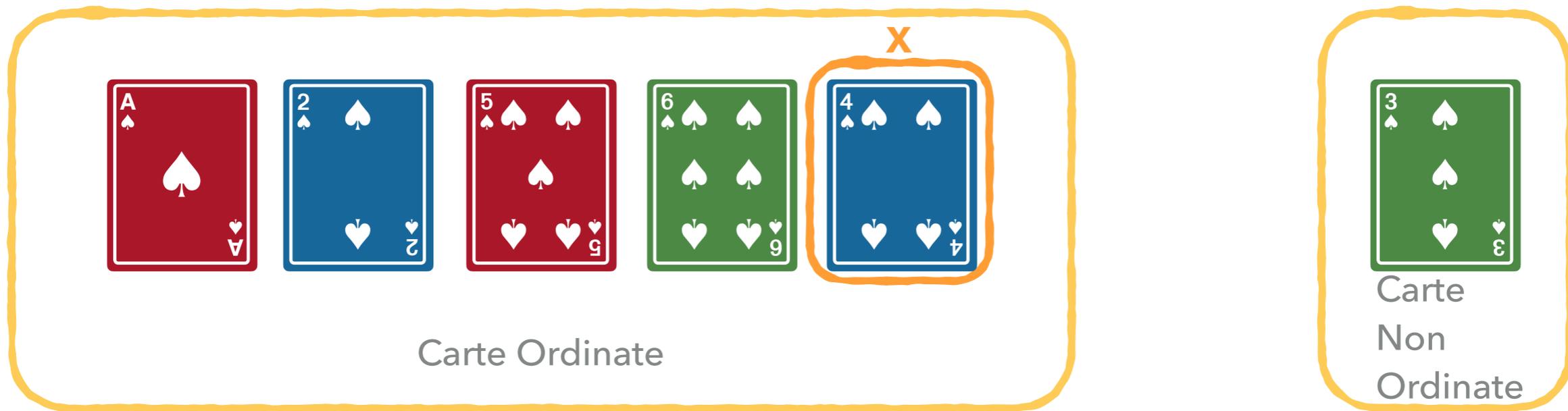
1. Inizialmente non si hanno carte ordinate
2. Finché ci rimangono carte non ordinate

→ 2.1. Prendi una carta dalla lista di carte non ordinata e chiamala X

2.2. Metti X in fondo alla lista di carte ordinate

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

OPZIONE 2: INSERIMENTO DI UNA CARTA ALLA VOLTA



1. Inizialmente non si hanno carte ordinate

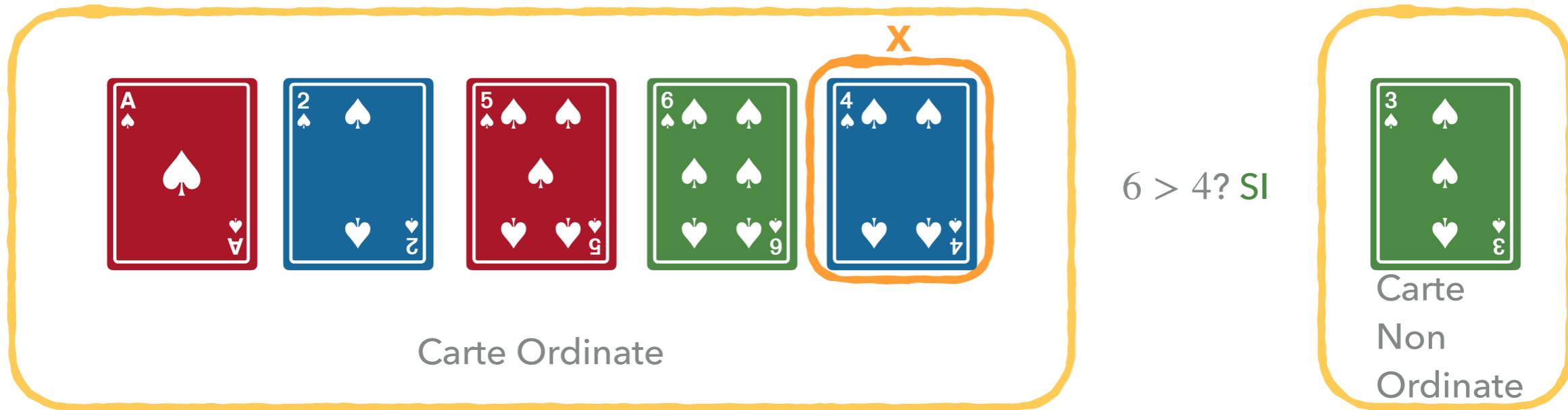
2. Finché ci rimangono carte non ordinate

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

→ 2.2. Metti X in fondo alla lista di carte ordinate

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

OPZIONE 2: INSERIMENTO DI UNA CARTA ALLA VOLTA



1. Inizialmente non si hanno carte ordinate

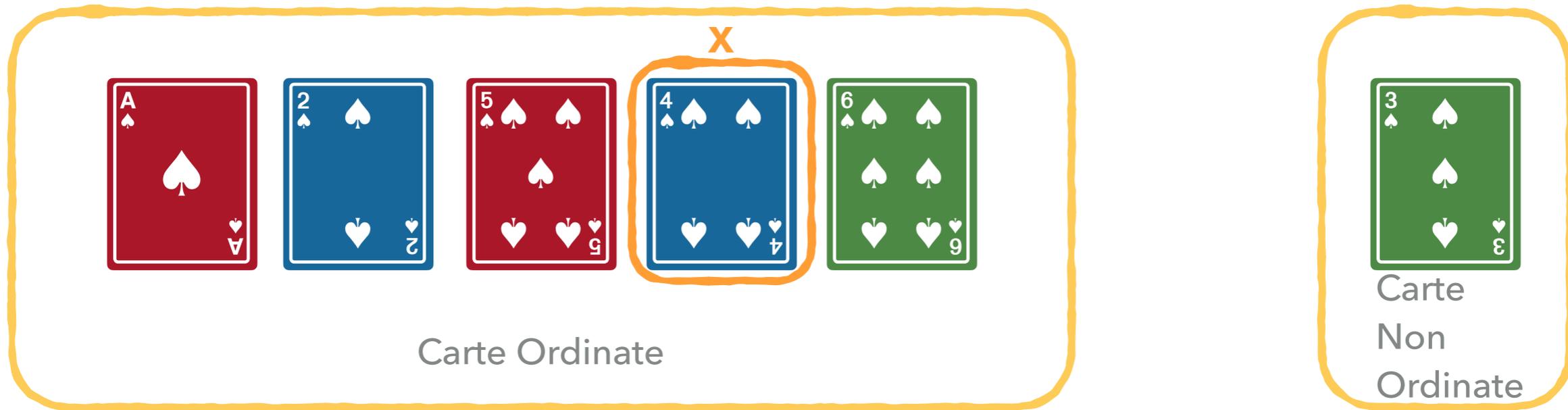
2. Finché ci rimangono carte non ordinate

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

2.2. Metti X in fondo alla lista di carte ordinate

→ 2.3. Finché la carta che precede X ha valore maggiore del valore di X, scambia X con la carta che lo precede

OPZIONE 2: INSERIMENTO DI UNA CARTA ALLA VOLTA



1. Inizialmente non si hanno carte ordinate

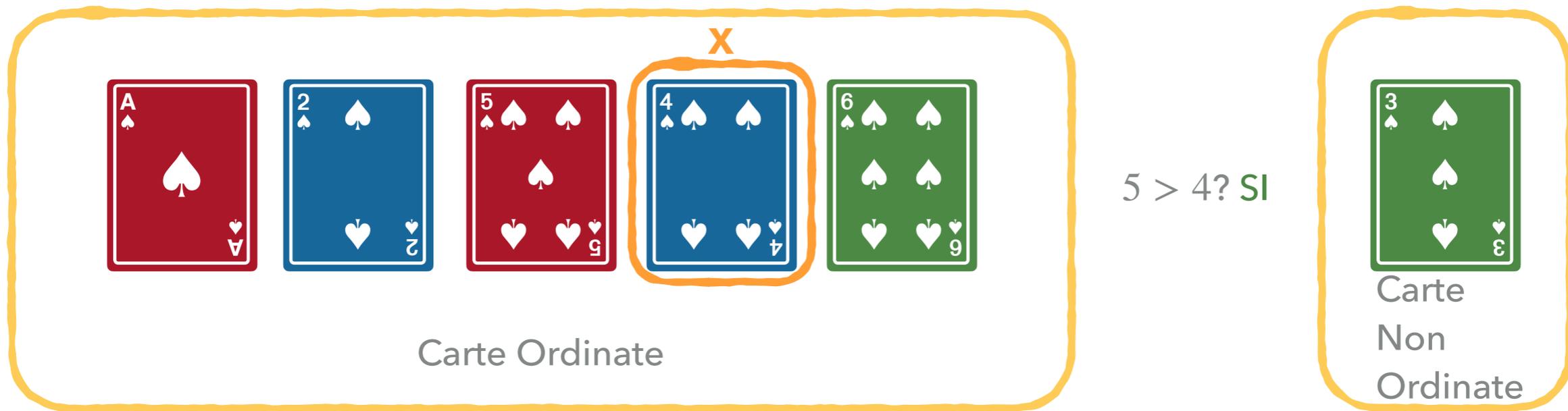
2. Finché ci rimangono carte non ordinate

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

2.2. Metti X in fondo alla lista di carte ordinate

→ 2.3. Finché la carta che precede X ha valore maggiore del valore di X, scambia X con la carta che lo precede

OPZIONE 2: INSERIMENTO DI UNA CARTA ALLA VOLTA



1. Inizialmente non si hanno carte ordinate

2. Finché ci rimangono carte non ordinate

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

2.2. Metti X in fondo alla lista di carte ordinate

→ 2.3. Finché la carta che precede X ha valore maggiore del valore di X, scambia X con la carta che lo precede

OPZIONE 2: INSERIMENTO DI UNA CARTA ALLA VOLTA



1. Inizialmente non si hanno carte ordinate

2. Finché ci rimangono carte non ordinate

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

2.2. Metti X in fondo alla lista di carte ordinate

→ 2.3. Finché la carta che precede X ha valore maggiore del valore di X, scambia X con la carta che lo precede

OPZIONE 2: INSERIMENTO DI UNA CARTA ALLA VOLTA



1. Inizialmente non si hanno carte ordinate

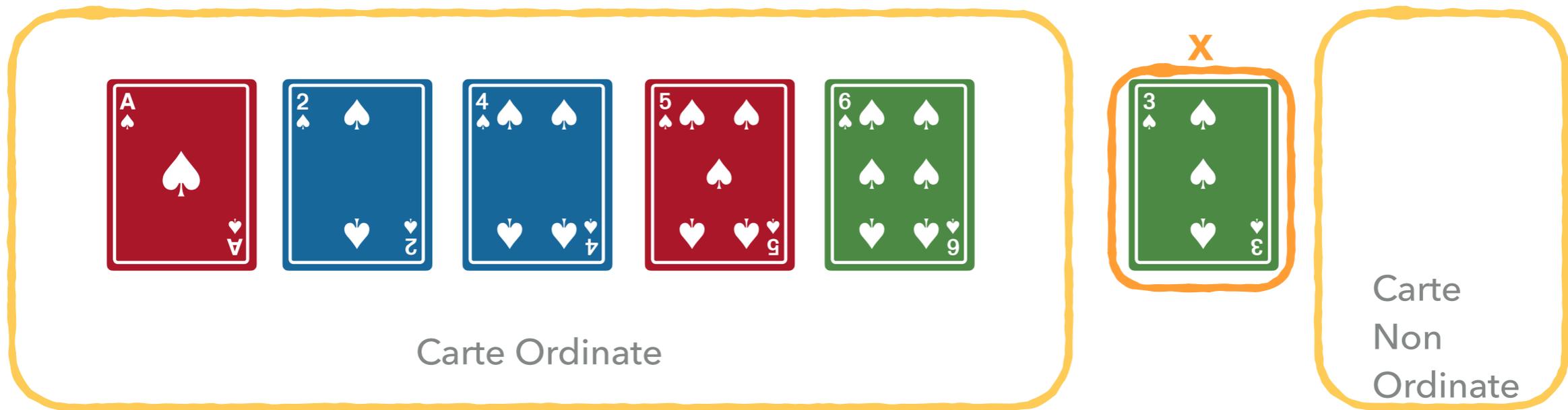
2. Finché ci rimangono carte non ordinate

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

2.2. Metti X in fondo alla lista di carte ordinate

→ 2.3. Finché la carta che precede X ha valore maggiore del valore di X, scambia X con la carta che lo precede

OPZIONE 2: INSERIMENTO DI UNA CARTA ALLA VOLTA



1. Inizialmente non si hanno carte ordinate

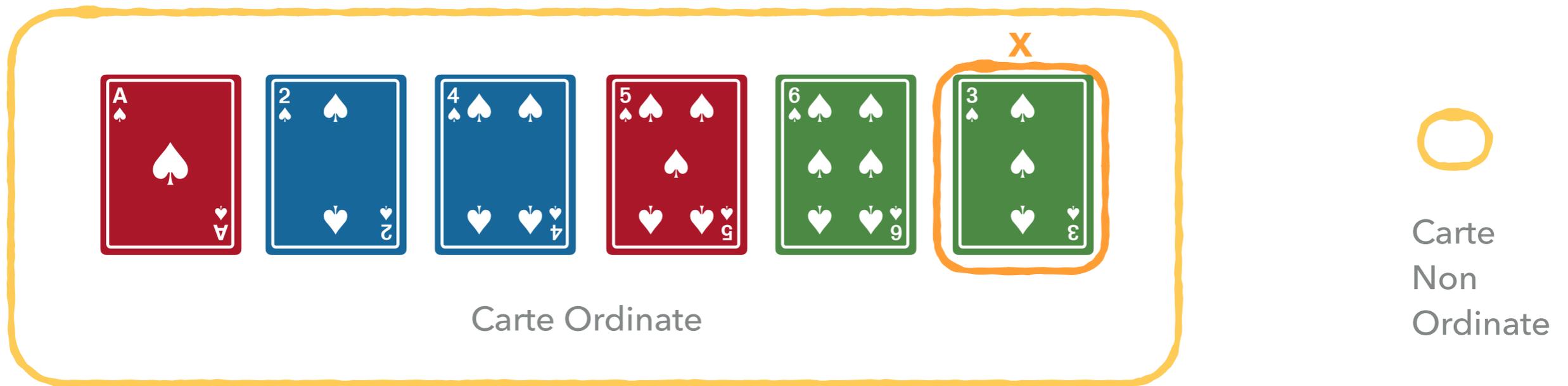
2. Finché ci rimangono carte non ordinate

→ 2.1. Prendi una carta dalla lista di carte non ordinata e chiamala X

2.2. Metti X in fondo alla lista di carte ordinate

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

OPZIONE 2: INSERIMENTO DI UNA CARTA ALLA VOLTA



1. Inizialmente non si hanno carte ordinate

2. Finché ci rimangono carte non ordinate

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

→ 2.2. Metti X in fondo alla lista di carte ordinate

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

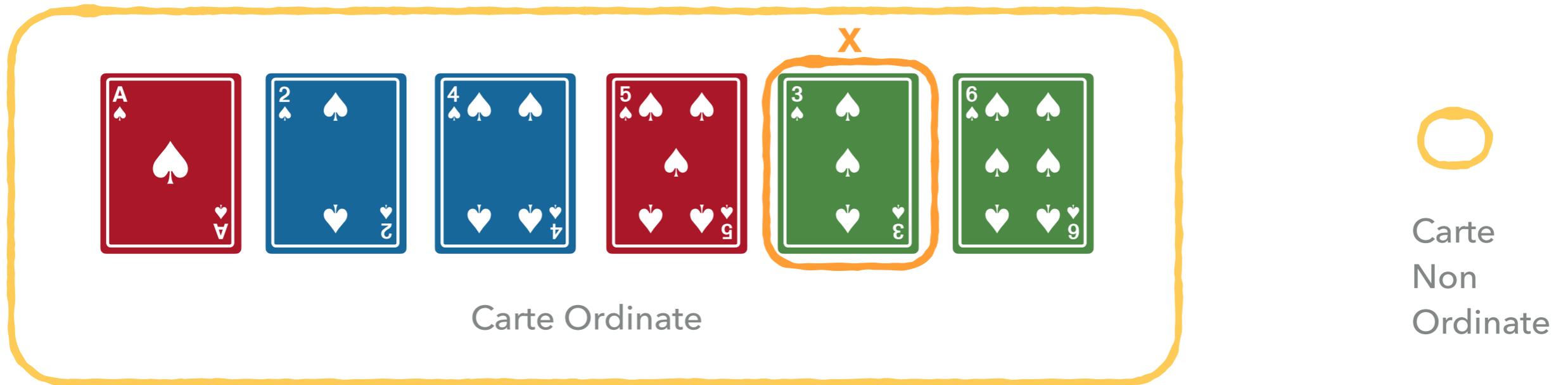
OPZIONE 2: INSERIMENTO DI UNA CARTA ALLA VOLTA



1. Inizialmente non si hanno carte ordinate
2. Finché ci rimangono carte non ordinate
 - 2.1. Prendi una carta dalla lista di carte non ordinata e chiamala X
 - 2.2. Metti X in fondo alla lista di carte ordinate

→ 2.3. Finché la carta che precede X ha valore maggiore del valore di X, scambia X con la carta che lo precede

OPZIONE 2: INSERIMENTO DI UNA CARTA ALLA VOLTA



1. Inizialmente non si hanno carte ordinate

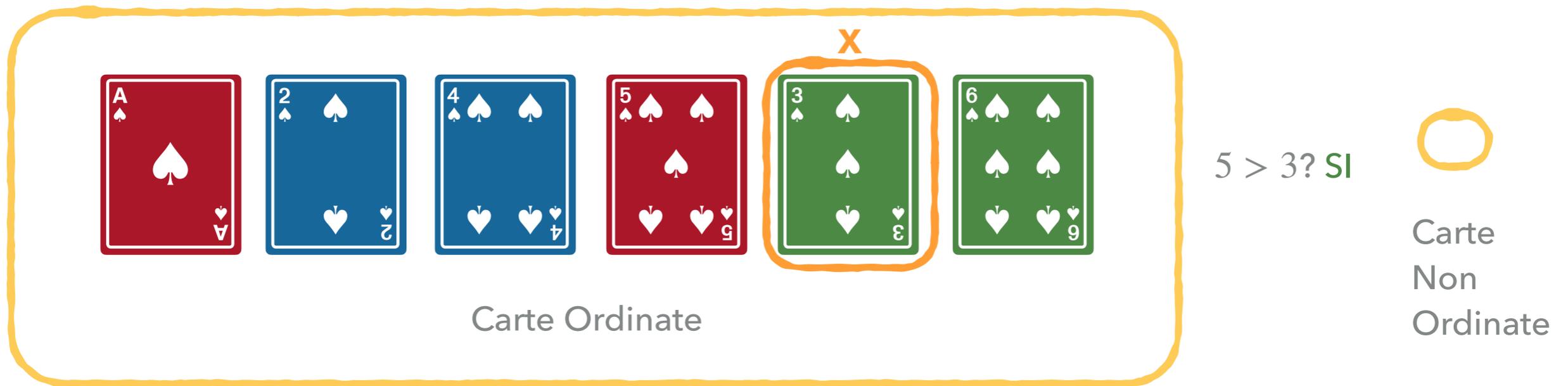
2. Finché ci rimangono carte non ordinate

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

2.2. Metti X in fondo alla lista di carte ordinate

→ 2.3. Finché la carta che precede X ha valore maggiore del valore di X, scambia X con la carta che lo precede

OPZIONE 2: INSERIMENTO DI UNA CARTA ALLA VOLTA



1. Inizialmente non si hanno carte ordinate

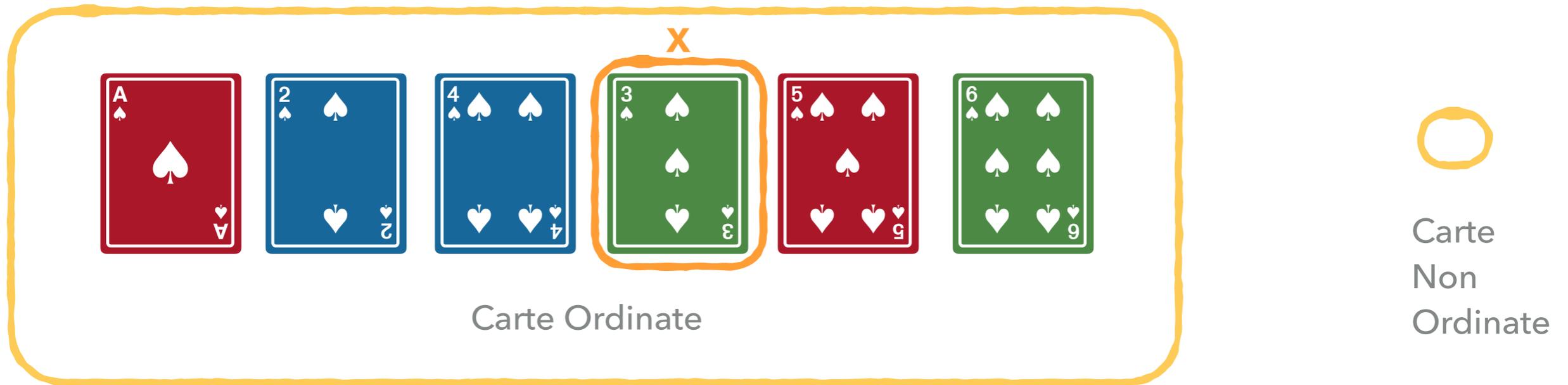
2. Finché ci rimangono carte non ordinate

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

2.2. Metti X in fondo alla lista di carte ordinate

→ 2.3. Finché la carta che precede X ha valore maggiore del valore di X, scambia X con la carta che lo precede

OPZIONE 2: INSERIMENTO DI UNA CARTA ALLA VOLTA



1. Inizialmente non si hanno carte ordinate

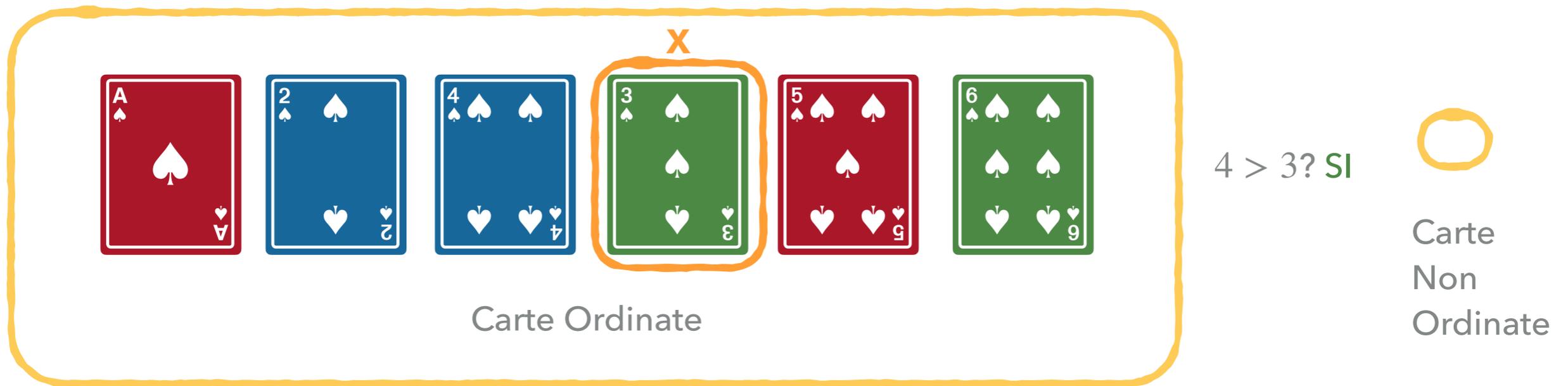
2. Finché ci rimangono carte non ordinate

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

2.2. Metti X in fondo alla lista di carte ordinate

→ 2.3. Finché la carta che precede X ha valore maggiore del valore di X, scambia X con la carta che lo precede

OPZIONE 2: INSERIMENTO DI UNA CARTA ALLA VOLTA



1. Inizialmente non si hanno carte ordinate

2. Finché ci rimangono carte non ordinate

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

2.2. Metti X in fondo alla lista di carte ordinate

→ 2.3. Finché la carta che precede X ha valore maggiore del valore di X, scambia X con la carta che lo precede

OPZIONE 2: INSERIMENTO DI UNA CARTA ALLA VOLTA



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

OPZIONE 2: INSERIMENTO DI UNA CARTA ALLA VOLTA



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

OPZIONE 2: INSERIMENTO DI UNA CARTA ALLA VOLTA



1. Inizialmente non si hanno carte ordinate

→ 2. Finché ci rimangono carte non ordinate

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

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

2.2. Metti X in fondo alla lista di carte ordinate

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

OPZIONE 2: INSERIMENTO DI UNA CARTA ALLA VOLTA



Siamo riusciti ad ottenere un mazzo di carte ordinate

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 ?

OPZIONE 2: INSERIMENTO DI UNA CARTA ALLA VOLTA

- ▶ Quante comparazioni dobbiamo fare nel caso peggiore?
- ▶ Inserire una carta nella lista delle carte ordinate richiede non più di n comparazioni
- ▶ Dobbiamo inserire n carte nella lista delle carte ordinate per ordinarle tutte
- ▶ Quindi il numero di comparazioni è limitato superiormente da n^2

EFFICIENCY MATTERS: MASSIMO COMUNE DIVISORE

Input: due numeri interi m ed n , $m > n$

Output: il più grande intero più piccolo di m ed n che li divide entrambi.

Idea: provo tutti i possibili divisori, dal più grande al più piccolo....

EFFICIENCY MATTERS: MASSIMO COMUNE DIVISORE

Input: due numeri interi m ed n , $m > n$

Output: il più grande intero più piccolo di m ed n che li divide entrambi.

Idea: provo tutti i possibili divisori, dal più grande al più piccolo....

```
def MCD(m,n):  
    mcd = 0  
    for i=1 to floor(sqrt(n)):  
        if i | n:  
            if n//i | m:  
                return n//i  
            else if i | m:  
                mcd = i  
    return mcd
```

EFFICIENCY MATTERS: MASSIMO COMUNE DIVISORE

Input: due numeri interi m ed n , $m > n$

Output: il più grande intero più piccolo di m ed n che li divide entrambi.

Idea: provo tutti i possibili divisori, dal più grande al più piccolo....

```
def MCD(m,n):  
    mcd = 0  
    for i=1 to floor(sqrt(n)):  
        if i | n:  
            if n//i | m:  
                return n//i  
            else if i | m:  
                mcd = i  
    return mcd
```

Complessità

l'algoritmo fa nel caso peggiore \sqrt{n} iterazioni del ciclo for.

Ci piace questa complessità?

Proviamo con $n = 2^{10}$?

EFFICIENCY MATTERS: MASSIMO COMUNE DIVISORE

Input: due numeri interi m ed n , $m > n$

Output: il più grande intero più piccolo di m ed n che li divide entrambi.

Idea: provo tutti i possibili divisori, dal più grande al più piccolo....

```
def MCD(m,n):  
    mcd = 0  
    for i=1 to floor(sqrt(n)):  
        if i | n:  
            if n//i | m:  
                return n//i  
            else if i | m:  
                mcd = i  
    return mcd
```

Complessità

l'algoritmo fa nel caso peggiore \sqrt{n} iterazioni del ciclo for.

Ci piace questa complessità?

Proviamo con $n = 2^{10}$?

e con 2^{100} ?

EFFICIENCY MATTERS: MASSIMO COMUNE DIVISORE

Input: due numeri interi m ed n , $m > n$

Output: il più grande intero p più piccolo di m ed n che li divide entrambi.

EFFICIENCY MATTERS: MASSIMO COMUNE DIVISORE

Input: due numeri interi m ed n , $m > n$

Output: il più grande intero p più piccolo di m ed n che li divide entrambi.

Idea: ci penso un poco...

partiamo da $m = qn + r$:

se $r = 0$, allora $n \mid m$, e ho finito

se $r \neq 0$, siccome $p \mid m$ e $p \mid n$, allora $p \mid r$

quindi $\text{MCD}(m, n) = \text{MCD}(n, r)$

EFFICIENCY MATTERS: MASSIMO COMUNE DIVISORE

Input: due numeri interi m ed n , $m > n$

Output: il più grande intero p più piccolo di m ed n che li divide entrambi.

Idea: ci penso un poco...

partiamo da $m = qn + r$:

se $r = 0$, allora $n \mid m$, e ho finito

se $r \neq 0$, siccome $p \mid m$ e $p \mid n$, allora $p \mid r$

quindi $\text{MCD}(m, n) = \text{MCD}(n, r)$

def **Euclide**(m, n):

 while $n \neq 0$:

$m, n = n, m \bmod n$

 return m

EFFICIENCY MATTERS: MASSIMO COMUNE DIVISORE

Input: due numeri interi m ed n , $m > n$

Output: il più grande intero p più piccolo di m ed n che li divide entrambi.

Idea: ci penso un poco...

partiamo da $m = qn + r$:

se $r = 0$, allora $n \mid m$, e ho finito

se $r \neq 0$, siccome $p \mid m$ e $p \mid n$, allora $p \mid r$

quindi $\text{MCD}(m, n) = \text{MCD}(n, r)$

def **Euclide**(m, n):

 while $n \neq 0$:

$m, n = n, m \bmod n$

 return m

Complessità:

Partiamo da $m = qn + r$

$m > n$, ergo $q \geq 1$ e $n > r$

Segue $m = qn + r \geq qr + r \geq r + r = 2r$

Quindi $r \leq m/2$.

Ergo in due passi di Euclide, m dimezza.

L'algoritmo termina dopo non più di $2 \log_2 m$ passi.

EFFICIENCY MATTERS: MASSIMO COMUNE DIVISORE

Input: due numeri interi m ed n , $m > n$

Output: il più grande intero p più piccolo di m ed n che li divide entrambi.

Idea: ci penso un poco...

partiamo da $m = qn + r$:

se $r = 0$, allora $n \mid m$, e ho finito

se $r \neq 0$, siccome $p \mid m$ e $p \mid n$, allora $p \mid r$

quindi $\text{MCD}(m, n) = \text{MCD}(n, r)$

def **Euclide**(m, n):

 while $n \neq 0$:

$m, n = n, m \bmod n$

 return m

Complessità:

Partiamo da $m = qn + r$

$m > n$, ergo $q \geq 1$ e $n > r$

Segue $m = qn + r \geq qr + r \geq r + r = 2r$

Quindi $r \leq m/2$.

Ergo in due passi di Euclide, m dimezza.

L'algoritmo termina dopo non più di $2 \log_2 m$ passi.

Compariamo la complessità quando $m = 2^{100}$ e $n = 2^{100}-1$