SQLServerFast.com Execution Plan Video Training

Block 3: Combining data

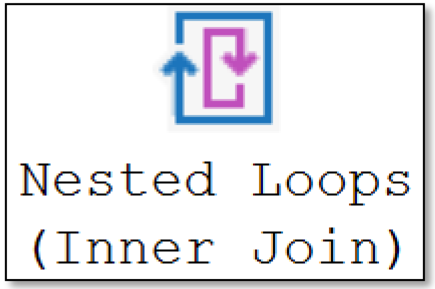
Level: Advanced

Chapter 1: Nested Loops (advanced)

Nested Loops (advanced)

Supported logical operations

- Inner Join
- Left Outer Join
- Left Semi Join
- Left Semi Join (probed)
- Left Anti Semi Join

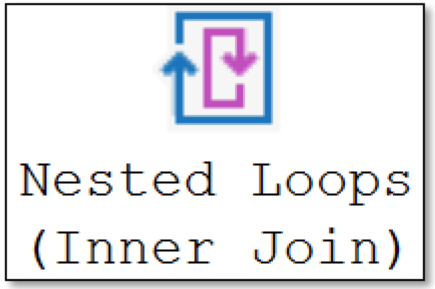


Nested Loops (advanced)

Supported logical operations

- Inner Join ← Algorithm shown in basic level
- Left Outer Join
- Left Semi Join
- Left Semi Join (probed)
- Left Anti Semi Join

} ← Algorithm shown here



Nested Loops
(Inner Join)

Nested Loops (advanced)

Effect of a join type change

FruitNum	FruitName
1	Apple
3	Cherry
2	Lime
3	Melon
2	Orange

ColorNum	ColorName	ColorCode
4	Blue	#0000FF
2	Orange	#FFA500
1	Red	#FF0000
2	Yellow	#FFFF00

Inner Join results

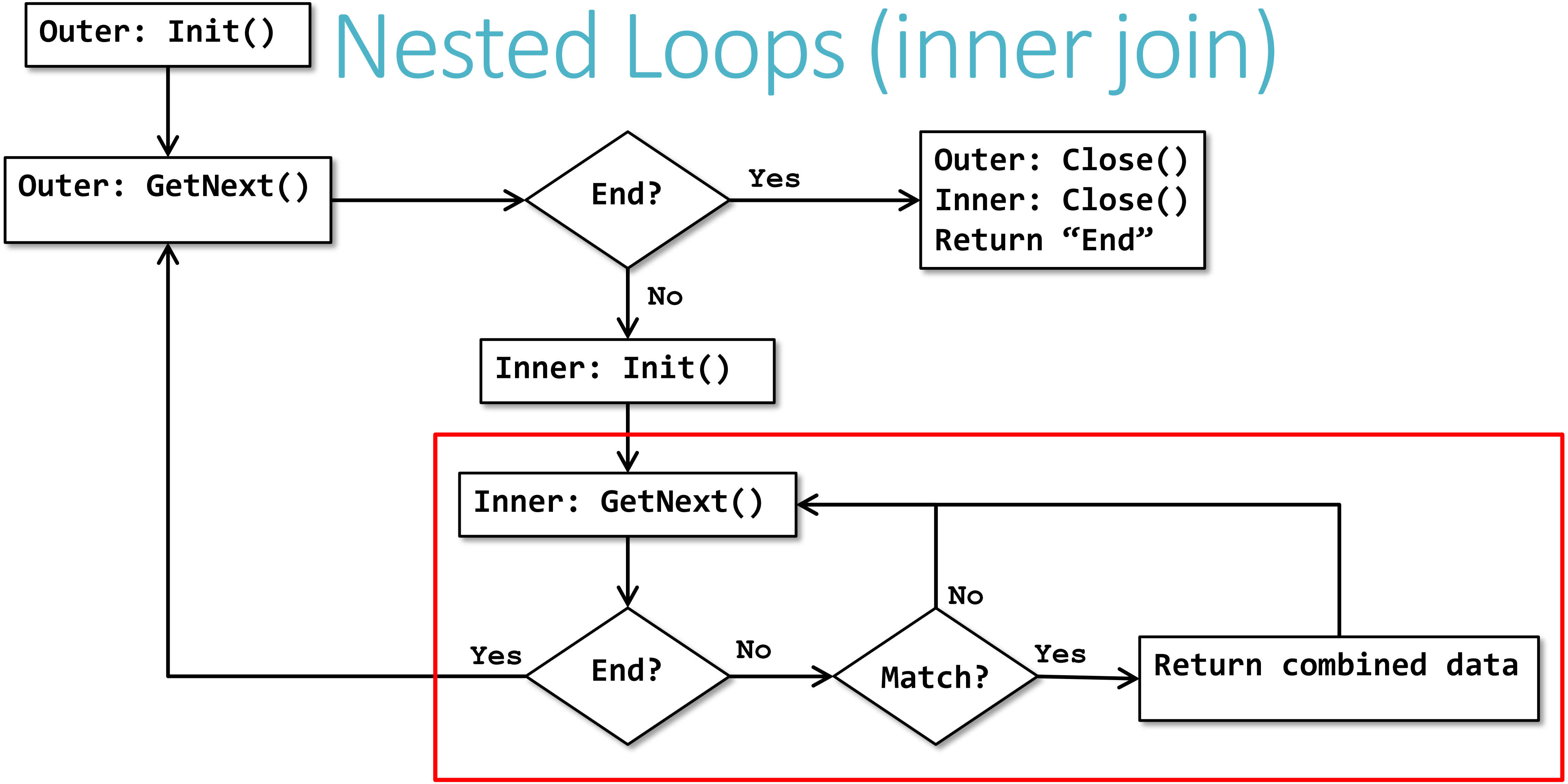
FruitNum	FruitName	ColorName
1	Apple	Red
2	Lime	Orange
2	Lime	Yellow
2	Orange	Orange
2	Orange	Yellow

Left Outer Join results

FruitNum	FruitName	ColorName
1	Apple	Red
2	Lime	Orange
2	Lime	Yellow
2	Orange	Orange
2	Orange	Yellow
3	Cherry	*(null)*
3	Melon	*(null)*

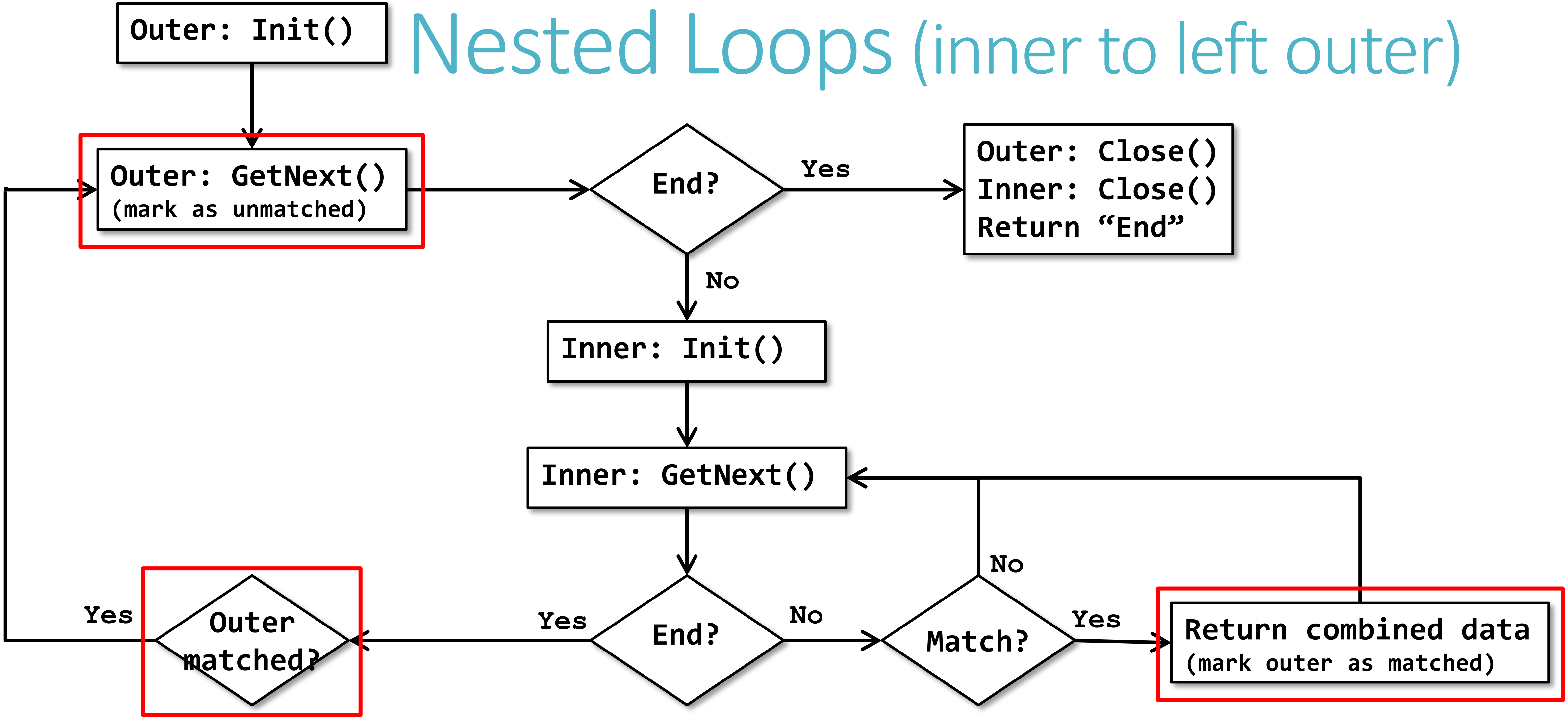
```
Nested Loops
(Inner Join)
```

Nested Loops (inner join)



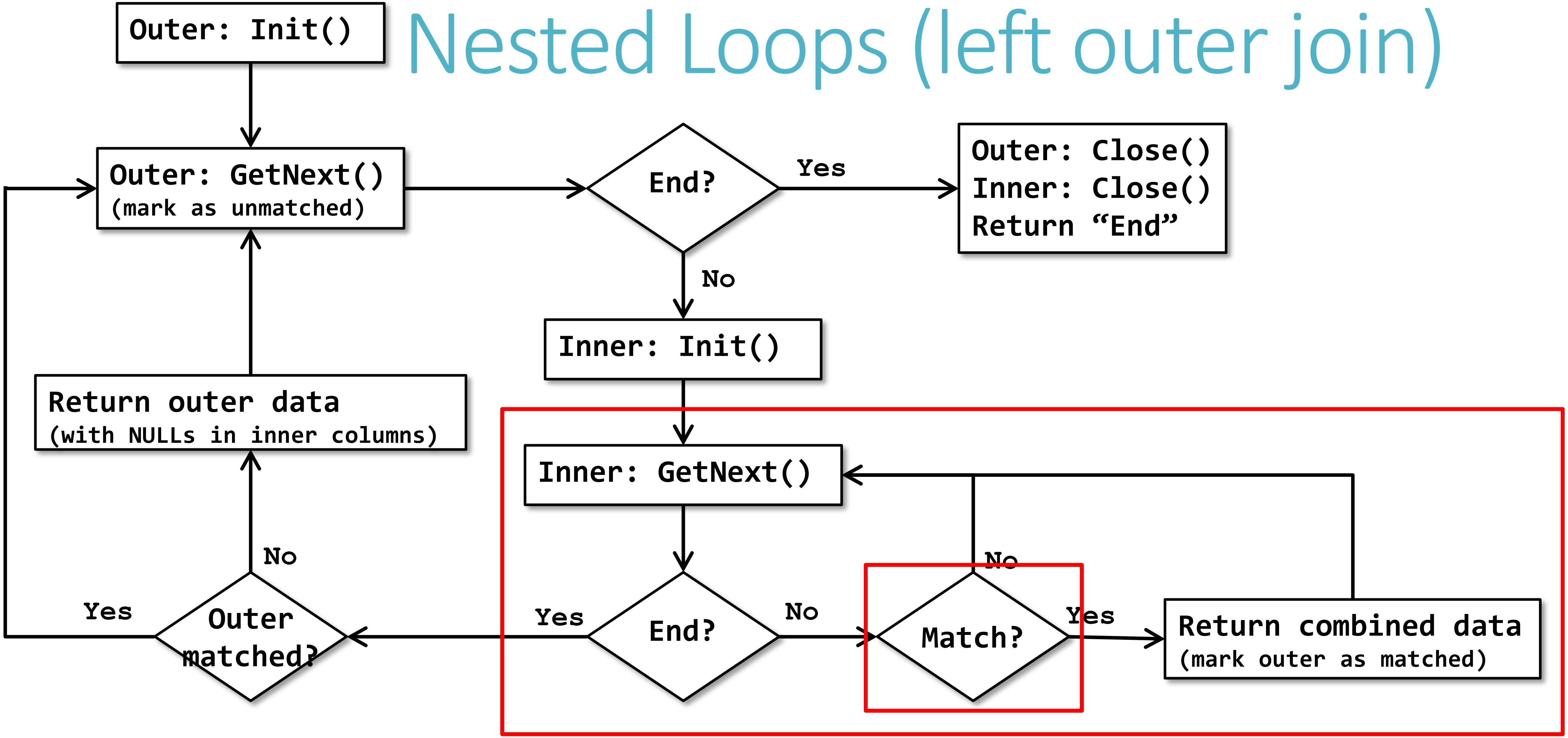
SQLServerFast.com execution plan training, block 3, basic level, chapter 2: Nested Loops - (c) Hugo Kornelis

Nested Loops (inner to left outer)



SQLServerFast.com execution plan training, block 3, basic level, chapter 2: Nested Loops - (c) Hugo Kornelis

Nested Loops (left outer join)



SQLServerFast.com execution plan training, block 3, basic level, chapter 2: Nested Loops - (c) Hugo Kornelis

Nested Loops (advanced)

Effect of a join type change

FruitNum	FruitName
1	Apple
3	Cherry
2	Lime
3	Melon
2	Orange

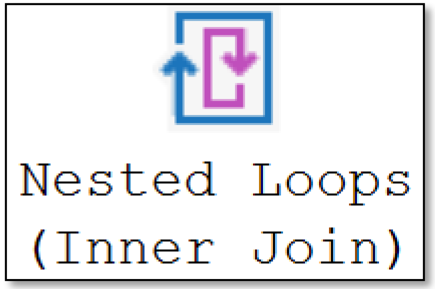
ColorNum	ColorName	ColorCode
4	Blue	#0000FF
2	Orange	#FFA500
1	Red	#FF0000
2	Yellow	#FFFF00

Inner Join results

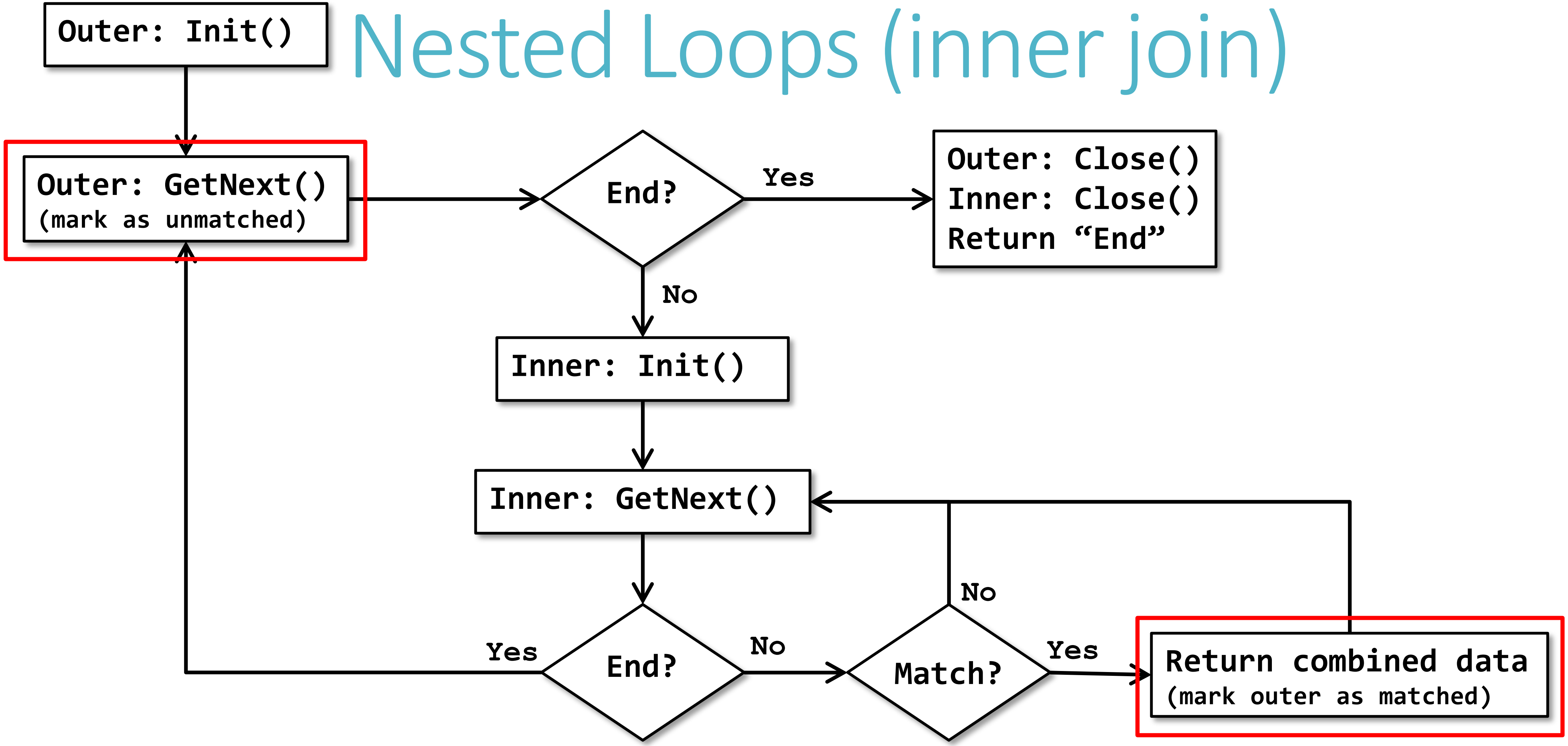
FruitNum	FruitName	ColorName
1	Apple	Red
2	Lime	Orange
2	Lime	Yellow
2	Orange	Orange
2	Orange	Yellow

Left Semi Join results

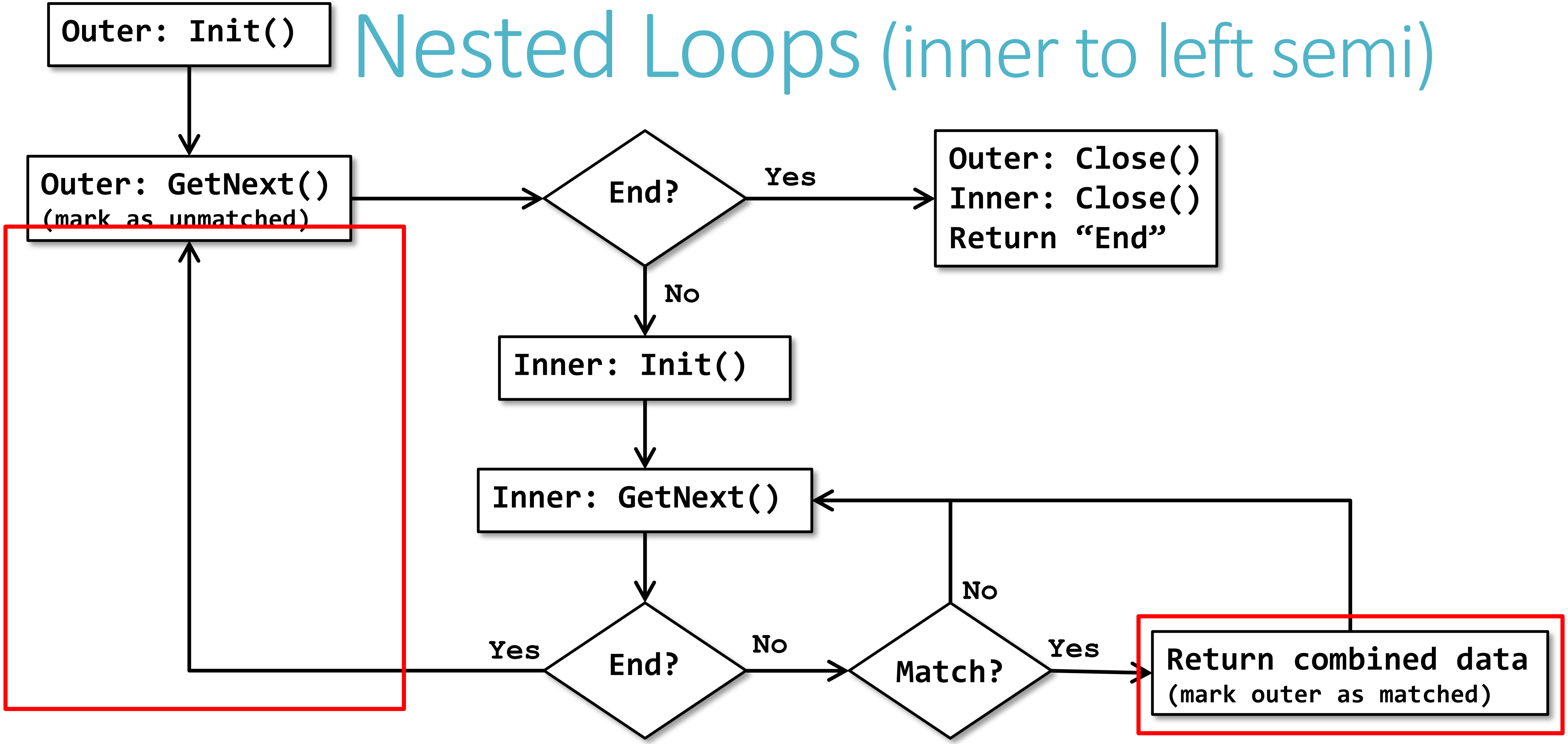
FruitNum	FruitName
1	Apple
2	Lime
2	Orange


Nested Loops
(Inner Join)

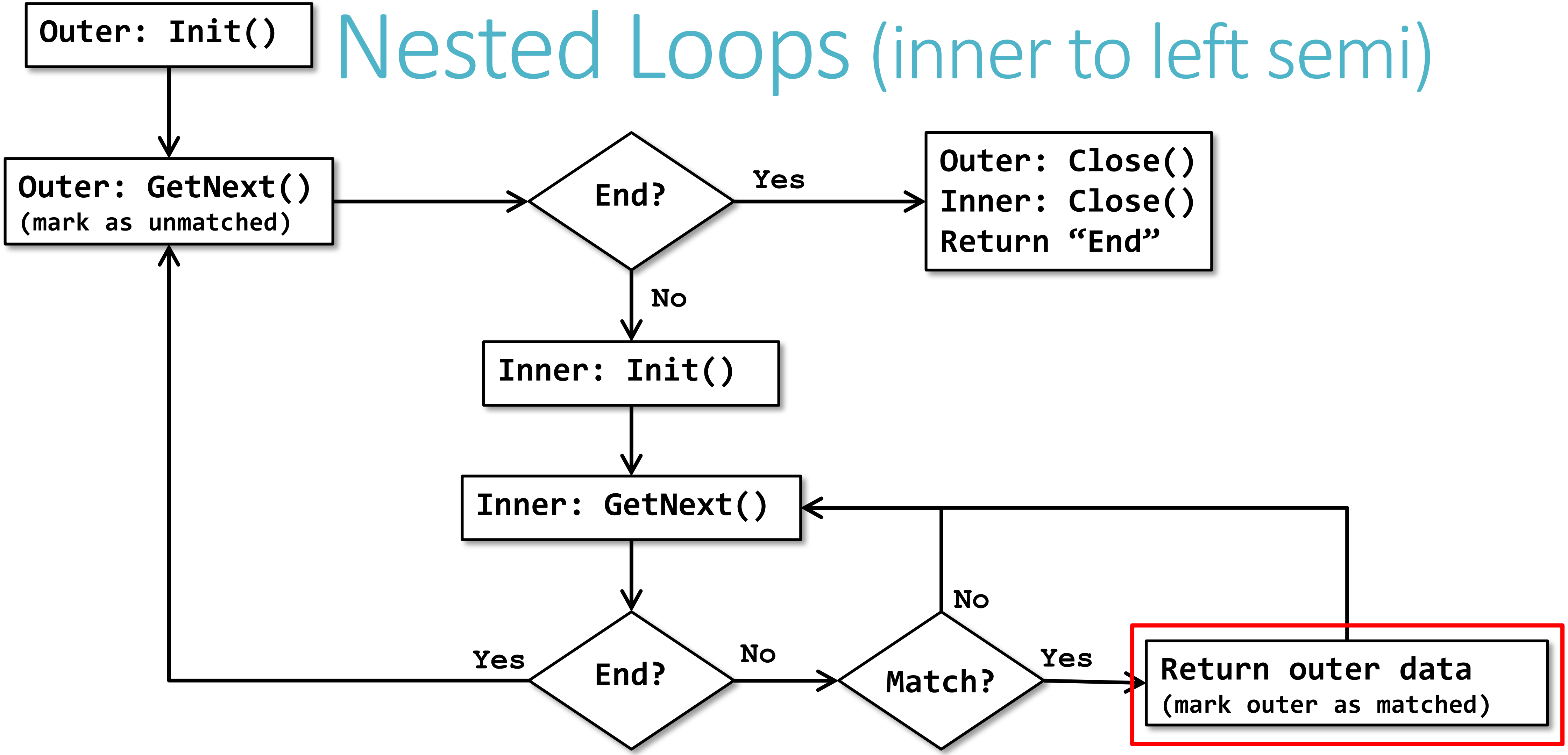
Nested Loops (inner join)



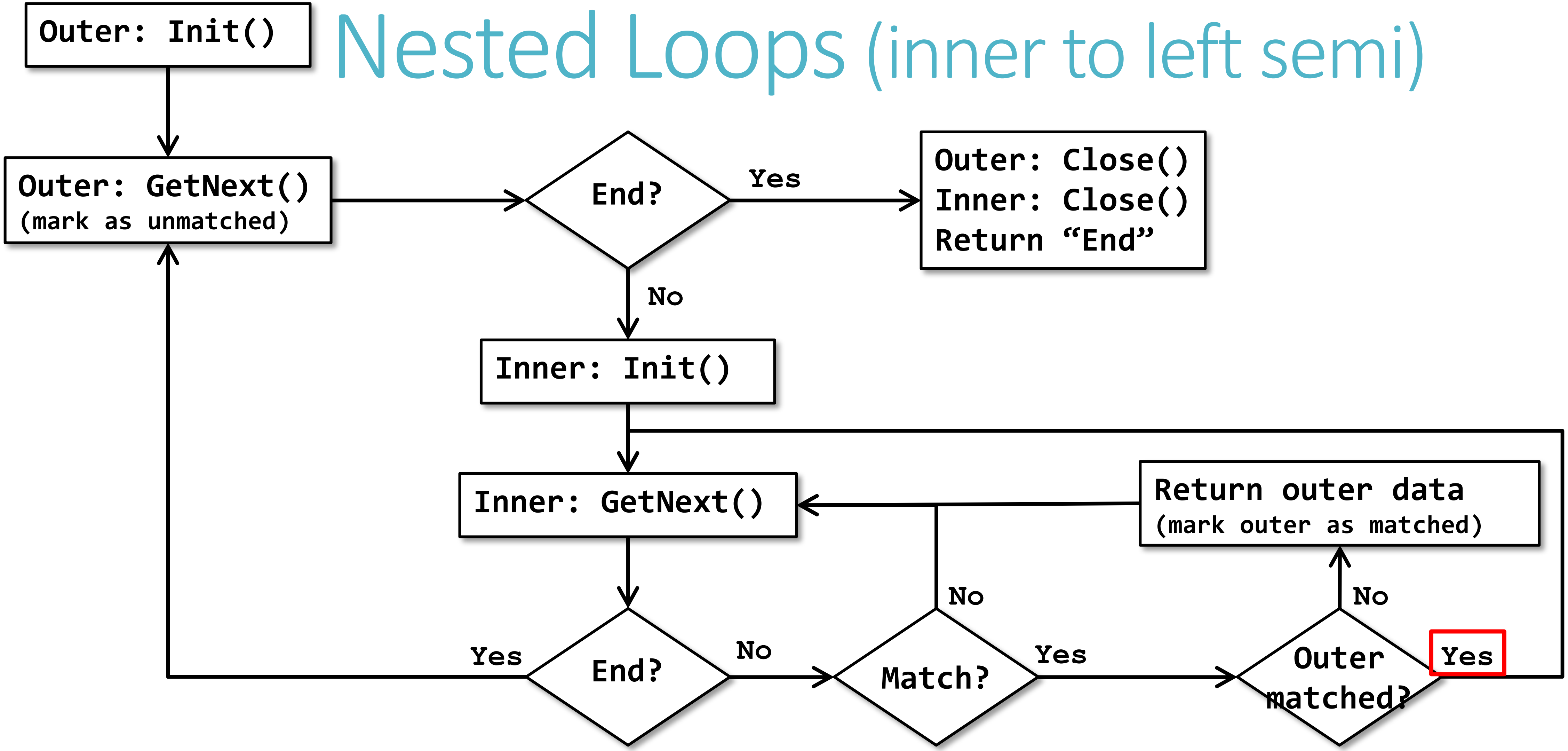
Nested Loops (inner to left semi)



Nested Loops (inner to left semi)

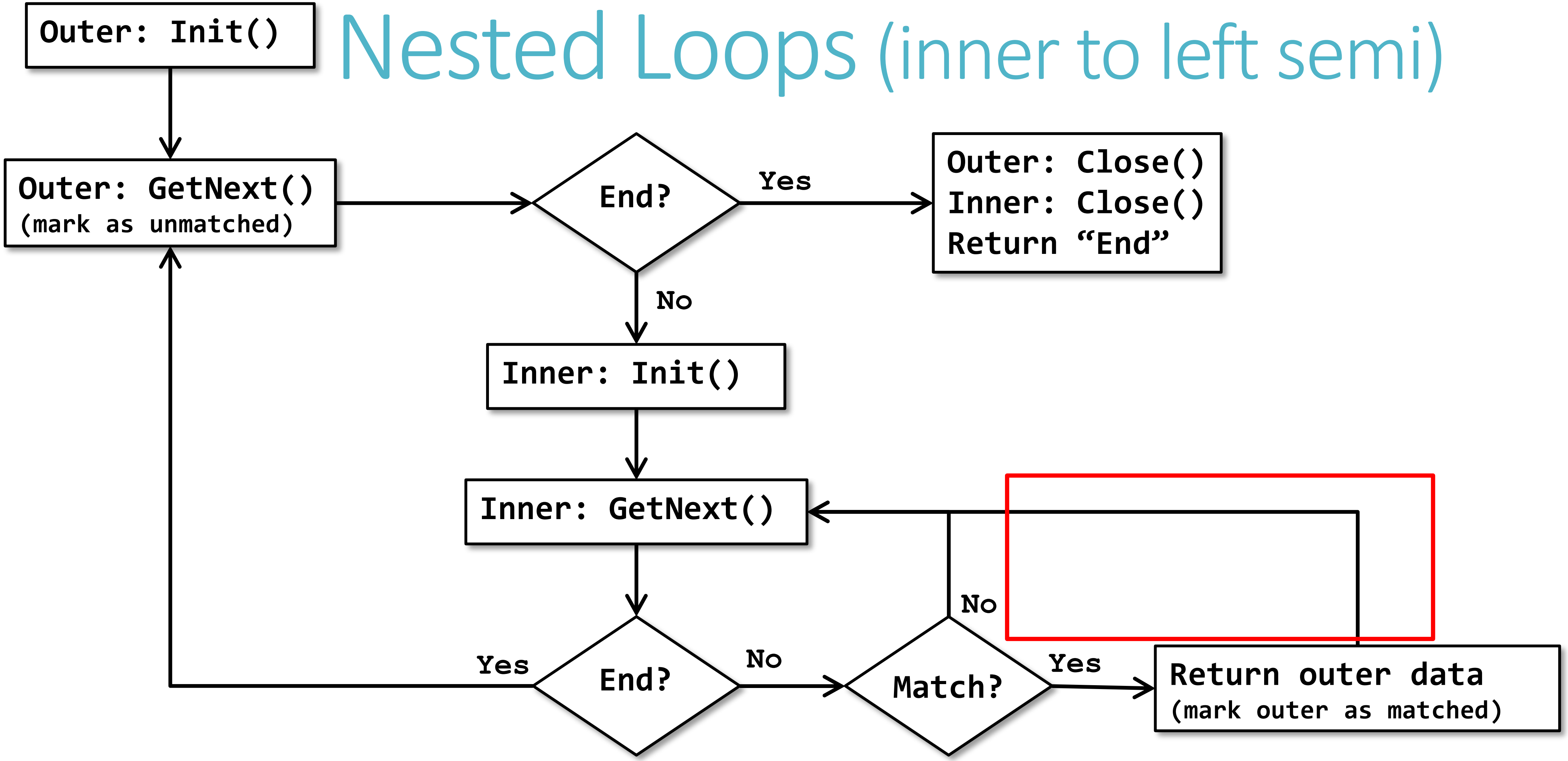


Nested Loops (inner to left semi)



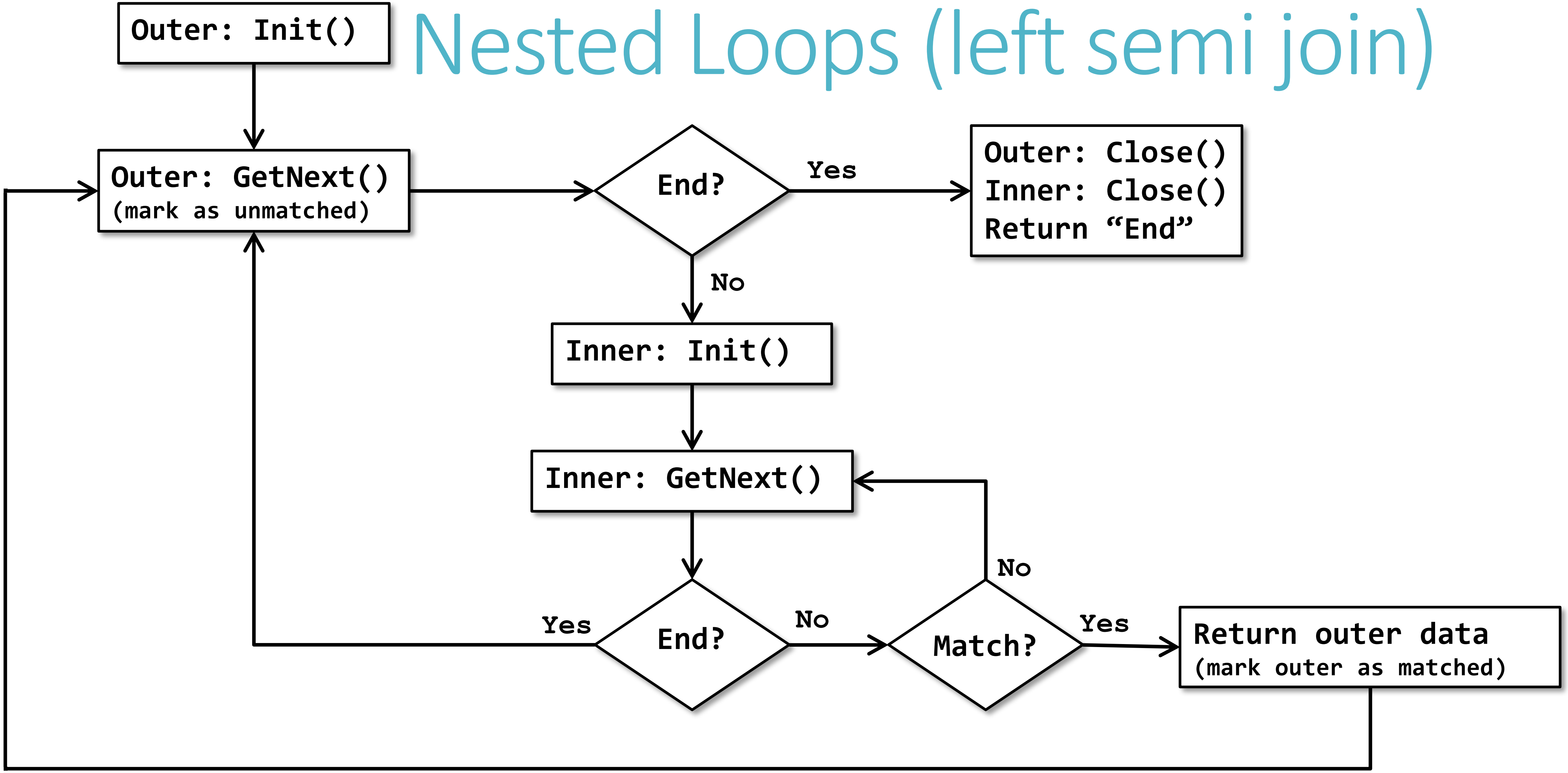
SQLServerFast.com execution plan training, block 3, basic level, chapter 2: Nested Loops - (c) Hugo Kornelis

Nested Loops (inner to left semi)



SQLServerFast.com execution plan training, block 3, basic level, chapter 2: Nested Loops - (c) Hugo Kornelis

Nested Loops (left semi join)



Nested Loops (advanced)

Effect of a join type change

FruitNum	FruitName
1	Apple
3	Cherry
2	Lime
3	Melon
2	Orange

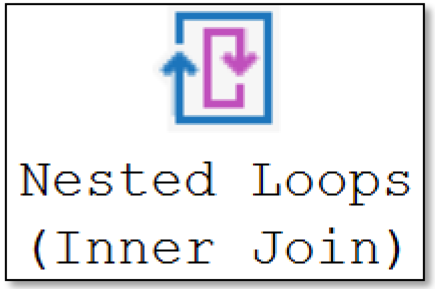
ColorNum	ColorName	ColorCode
4	Blue	#0000FF
2	Orange	#FFA500
1	Red	#FF0000
2	Yellow	#FFFF00

Left Semi Join results

FruitNum	FruitName
1	Apple
2	Lime
2	Orange

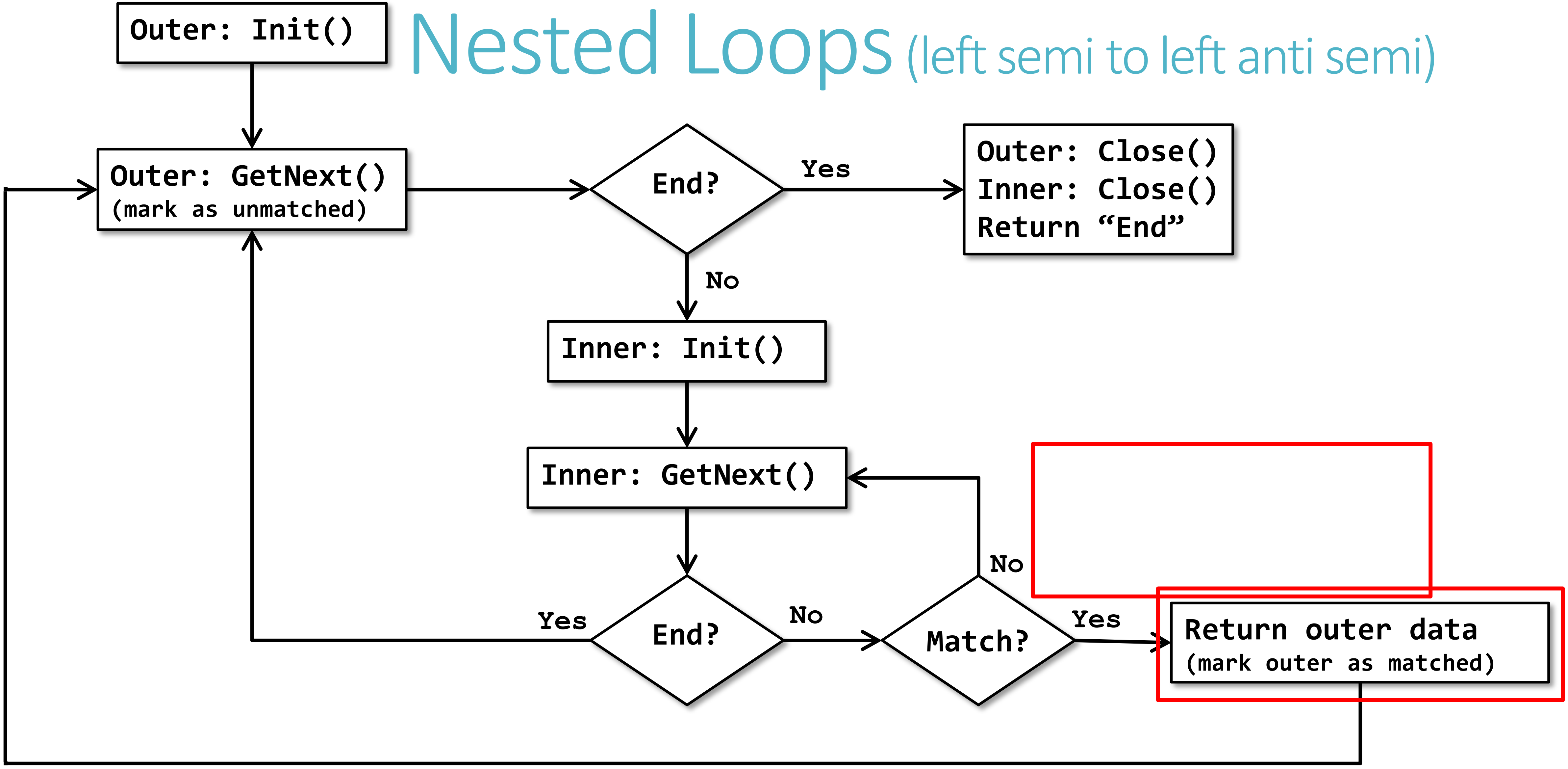
Left Anti Semi Join results

FruitNum	FruitName
3	Cherry
3	Melon



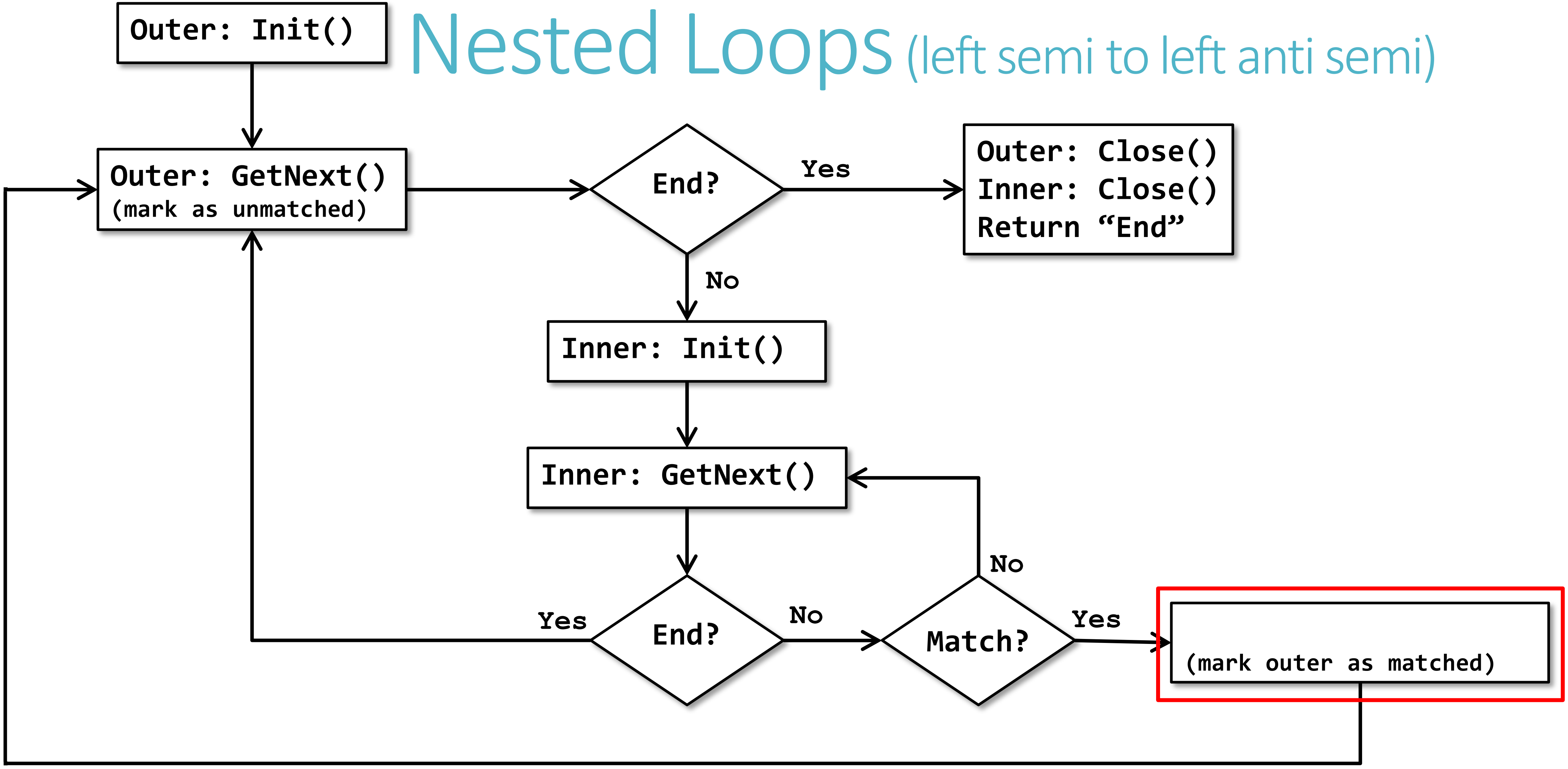
Nested Loops
(Inner Join)

Nested Loops (left semi to left anti semi)



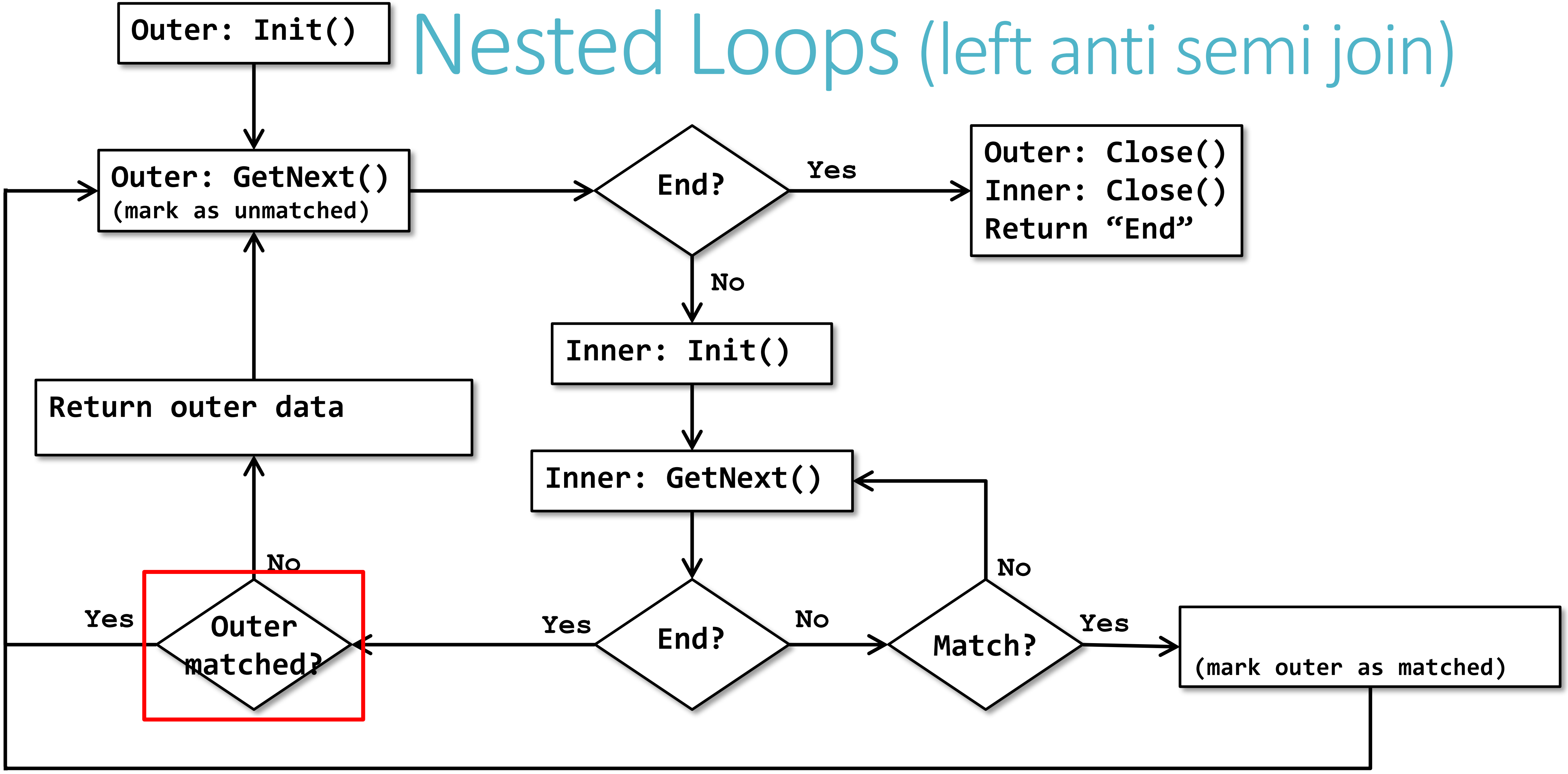
SQLServerFast.com execution plan training, block 3, basic level, chapter 2: Nested Loops - (c) Hugo Kornelis

Nested Loops (left semi to left anti semi)



SQLServerFast.com execution plan training, block 3, basic level, chapter 2: Nested Loops - (c) Hugo Kornelis

Nested Loops (left anti semi join)



SQLServerFast.com execution plan training, block 3, basic level, chapter 2: Nested Loops - (c) Hugo Kornelis

Nested Loops (advanced)

Effect of a join type change

FruitNum	FruitName
1	Apple
3	Cherry
2	Lime
3	Melon
2	Orange

ColorNum	ColorName	ColorCode
4	Blue	#0000FF
2	Orange	#FFA500
1	Red	#FF0000
2	Yellow	#FFFF00

Left Semi Join results

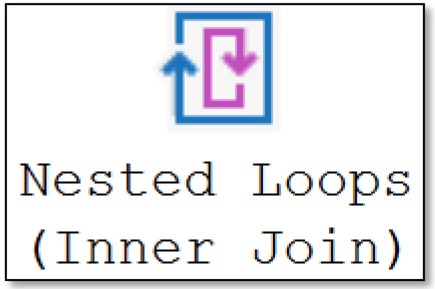
FruitNum	FruitName
1	Apple
2	Lime
2	Orange

Left Anti Semi Join results

FruitNum	FruitName
3	Cherry
3	Melon

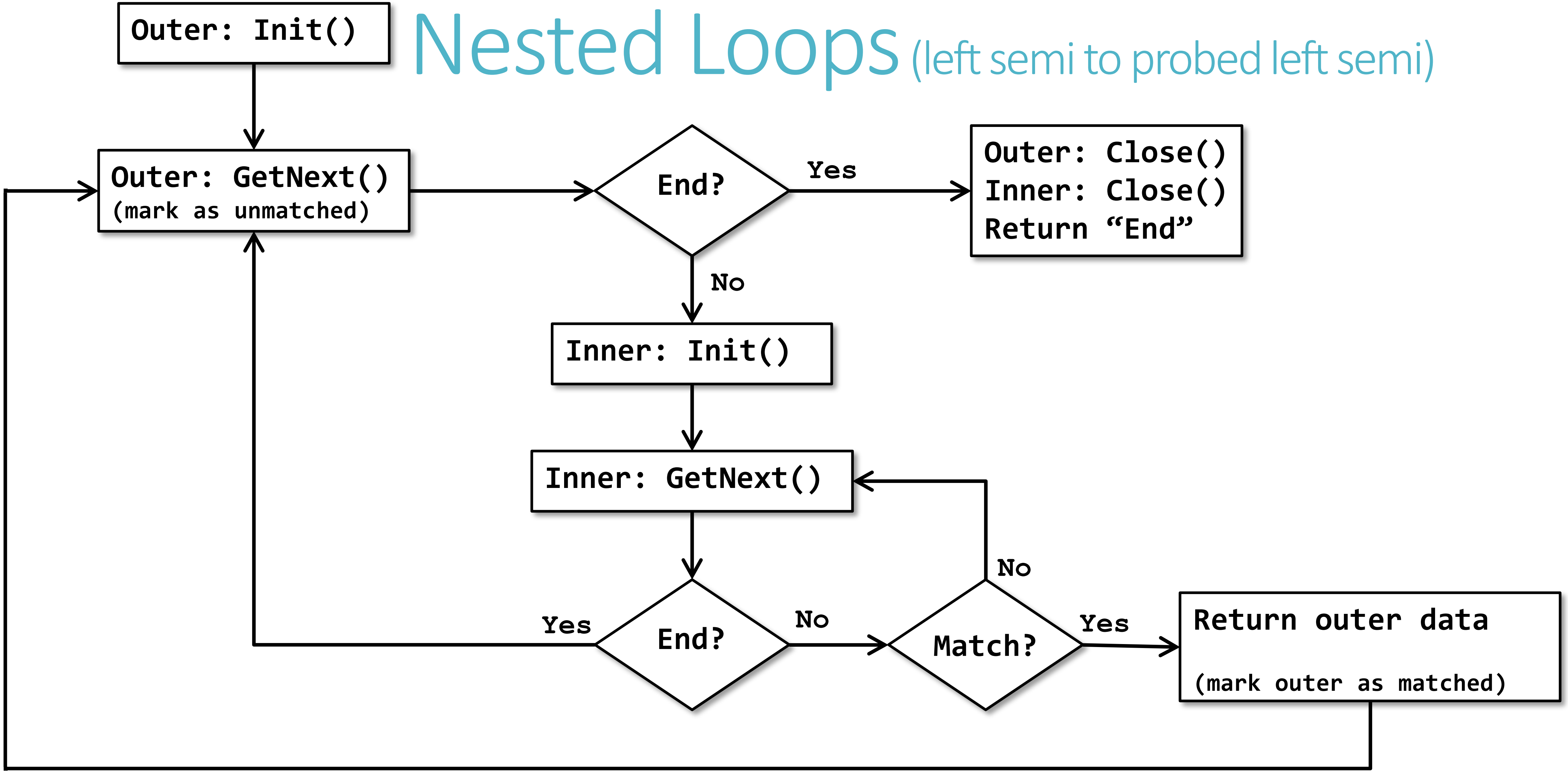
Probed Left Semi Join results

FruitNum	FruitName	Probe
1	Apple	True
2	Lime	True
2	Orange	True
3	Cherry	False
3	Melon	False



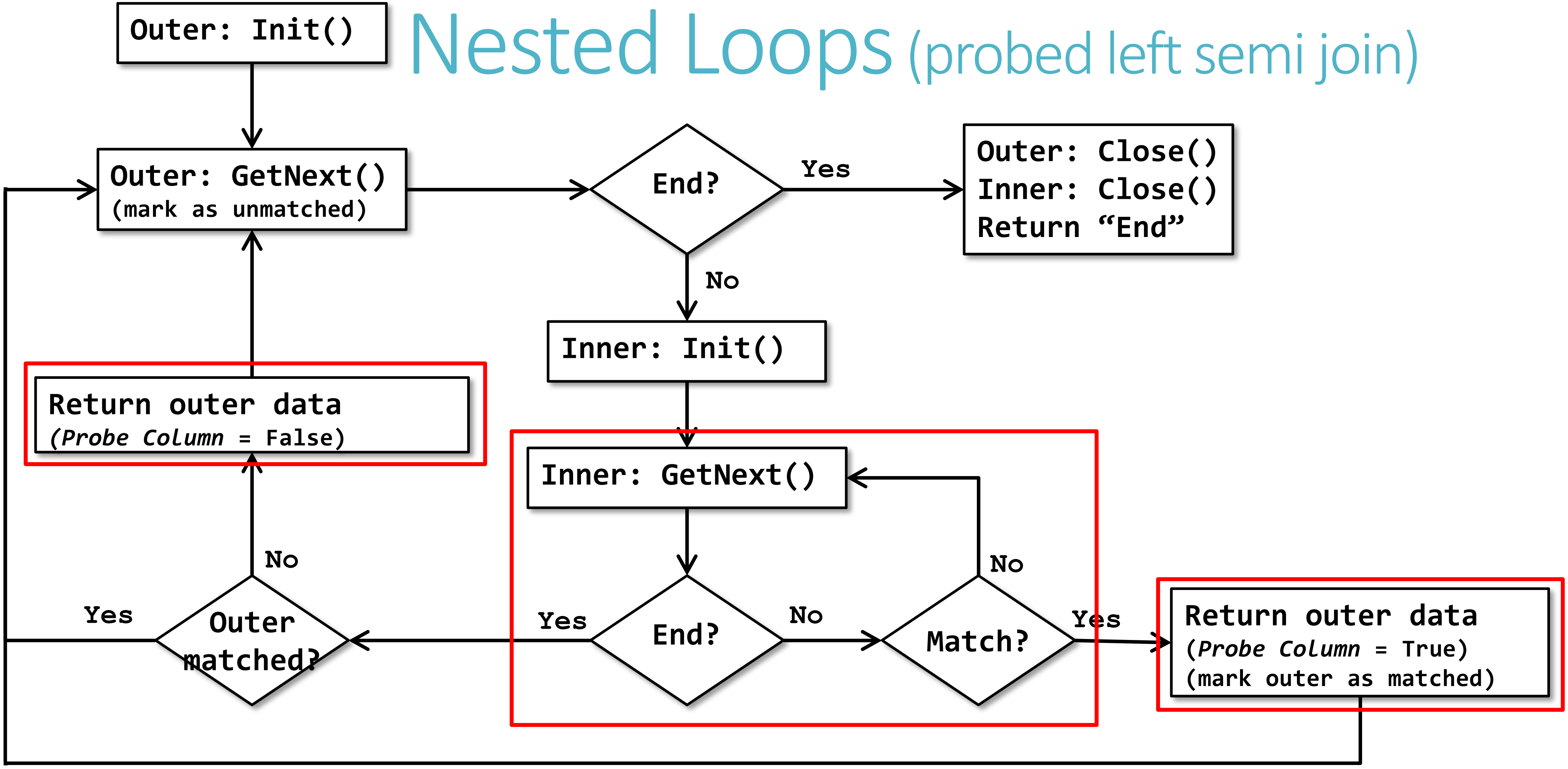
Nested Loops
(Inner Join)

Nested Loops (left semi to probed left semi)



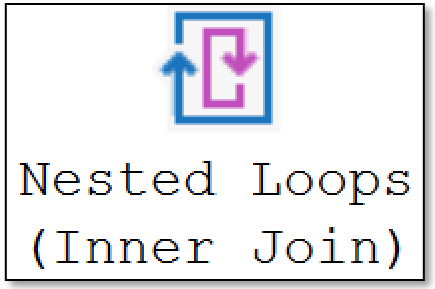
SQLServerFast.com execution plan training, block 3, basic level, chapter 2: Nested Loops - (c) Hugo Kornelis

Nested Loops (probed left semi join)

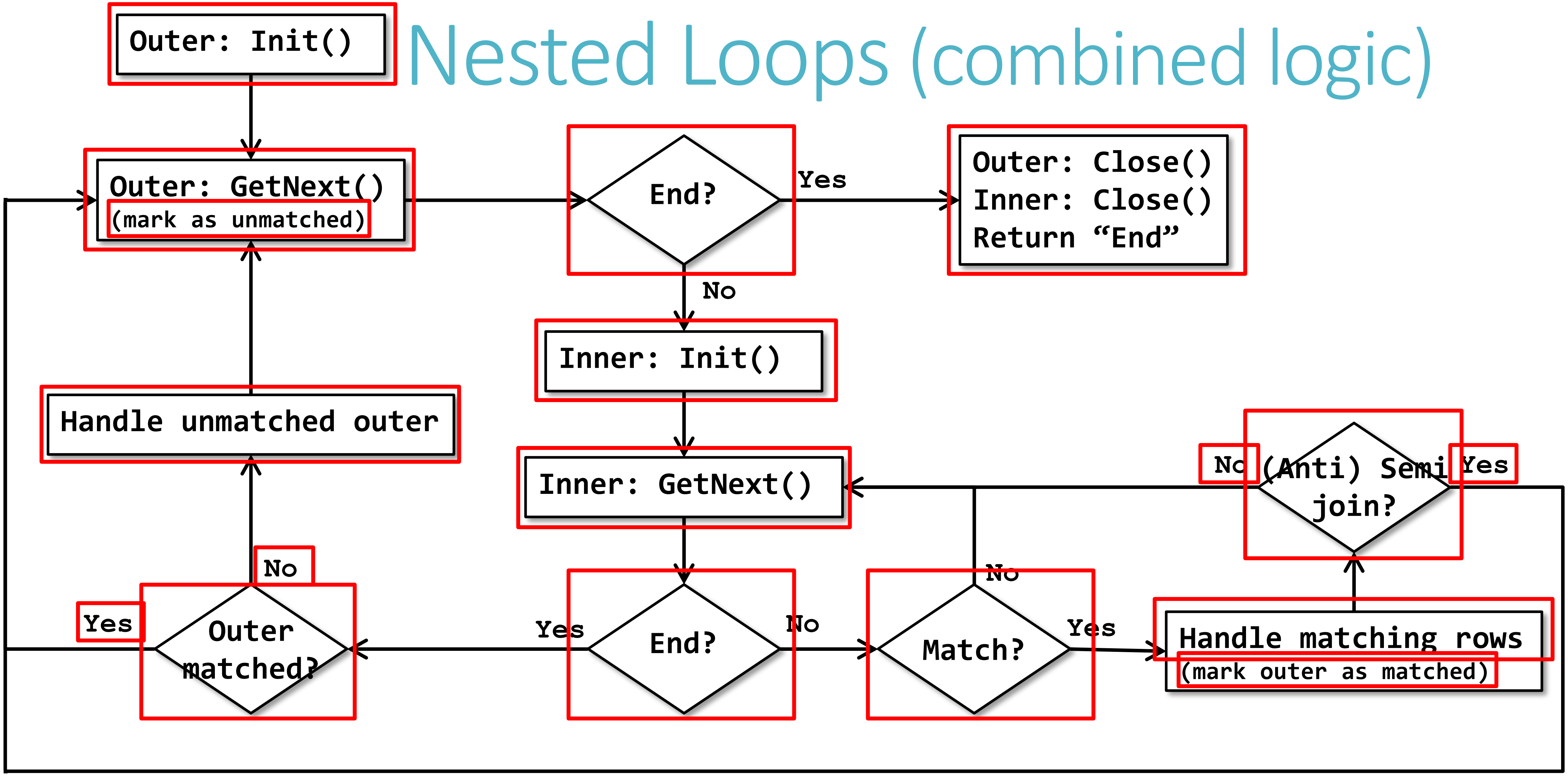


Nested Loops (advanced)

- Supported logical operations
  - Inner Join
  - Left Outer Join
  - Left Semi Join
  - Left Semi Join (probed)
  - Left Anti Semi Join
- Single code unit that combines the five flowcharts



Nested Loops (combined logic)



Outer: Init()

Outer: GetNext()
(mark as unmatched)

End?

Yes

Outer: Close()
Inner: Close()
Return "End"

No

Inner: Init()

Inner: GetNext()

Handle unmatched outer

No

Yes

Outer matched?

Yes

End?

No

Match?

No

Yes

Handle matching rows
(mark outer as matched)

(Anti) Semi join?

No

Yes

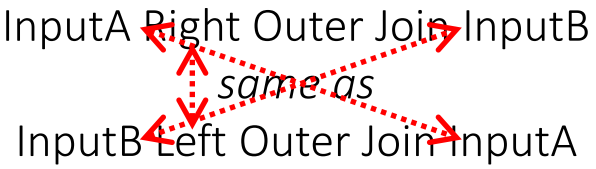
Nested Loops (advanced)

Supported logical operations

- Inner Join
- Left Outer Join
- Left Semi Join
- Left Semi Join (probed)
- Left Anti Semi Join

Unsupported join types

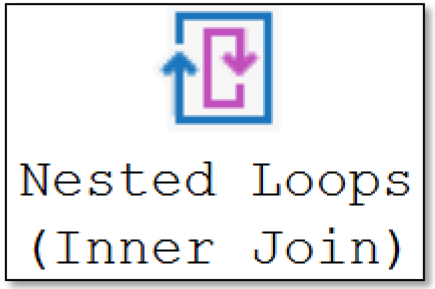
- Full Outer Join
- Right Outer Join
- Right Semi Join
- Right Semi Join (probed)
- Right Anti Semi Join



InputA Right Outer Join InputB

same as

InputB Left Outer Join InputA



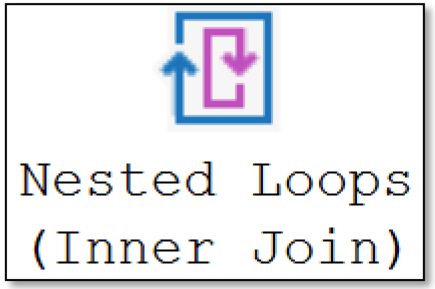
Nested Loops
(Inner Join)

Nested Loops (advanced)

Supported logical operations	Unsupported join types
Inner Join	Full Outer Join
Left Outer Join ↔	Right Outer Join
Left Semi Join ↔	Right Semi Join
Left Semi Join (probed) ↔	Right Semi Join (probed)
Left Anti Semi Join ↔	Right Anti Semi Join

Performance effect

- Small left input, cheap right input
- Optimizer can reverse inputs for Inner Join


Nested Loops
(Inner Join)

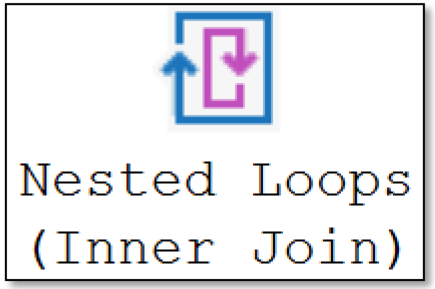
Nested Loops (advanced)

Supported logical operations

- Inner Join → preserves order
- Left Outer Join → preserves order
- Left Semi Join → preserves order
- Left Semi Join (probed) → preserves order
- Left Anti Semi Join → preserves order

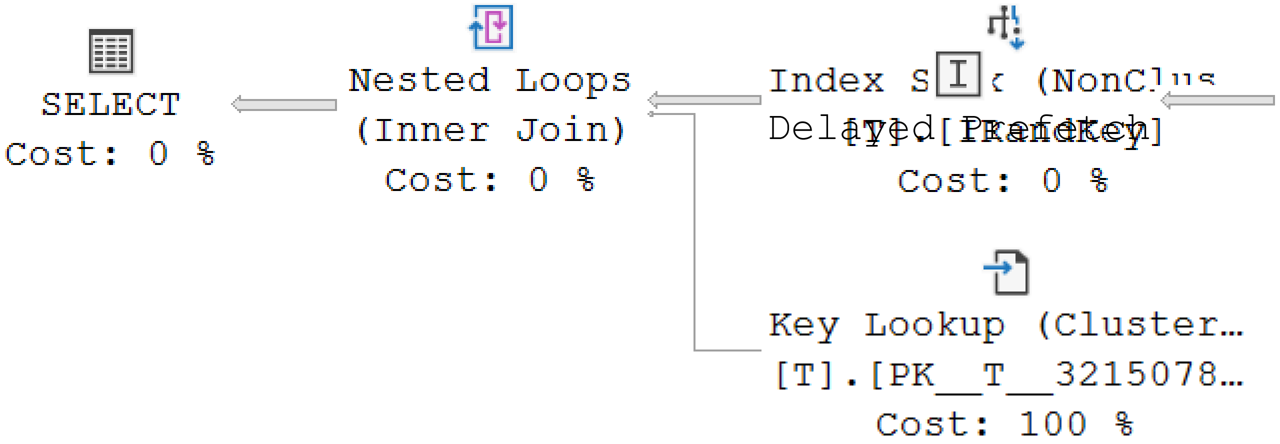
But not always!!!!

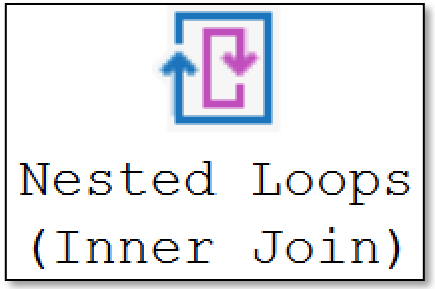
- Order may be affected by advanced optimizations



Nested Loops (advanced)

- Advanced performance optimizations
  - Ordered prefetching
    - Optimizes *physical* reads on bottom input
    - Property: *WithOrderedPrefetch*
    - Invisible node in execution plan

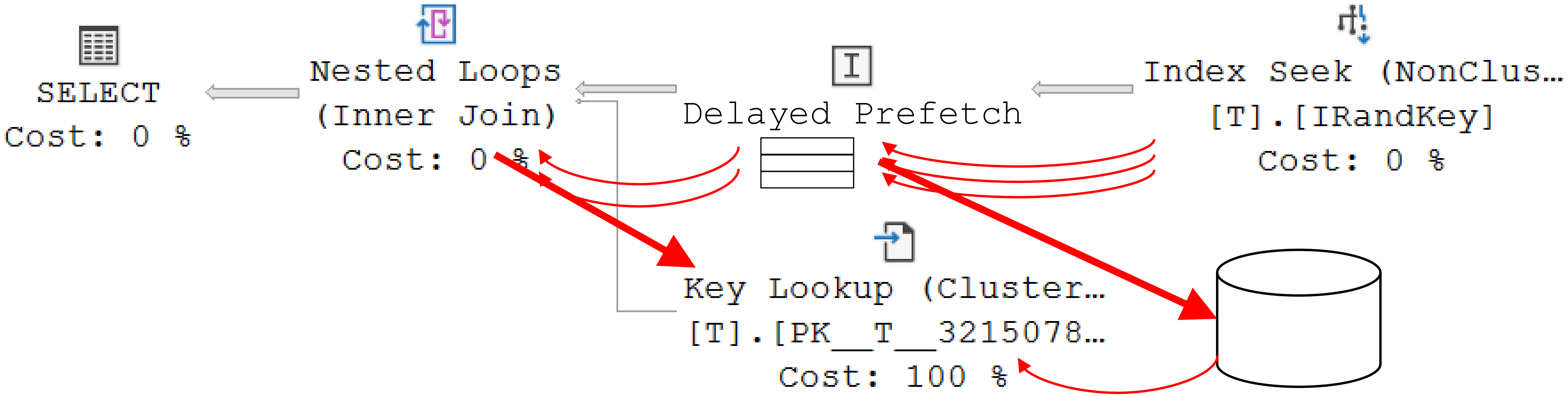


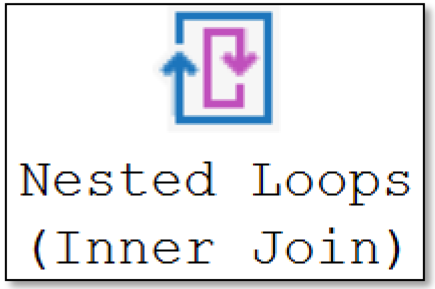


Nested Loops (advanced)

Delayed Prefetch

- Reads a few rows ahead
- Checks whether required “lower input” pages are in buffer pool
  - If not: Issues “scatter read” request to I/O subsystem
  - Most rows will be in buffer pool when Key Lookup needs them

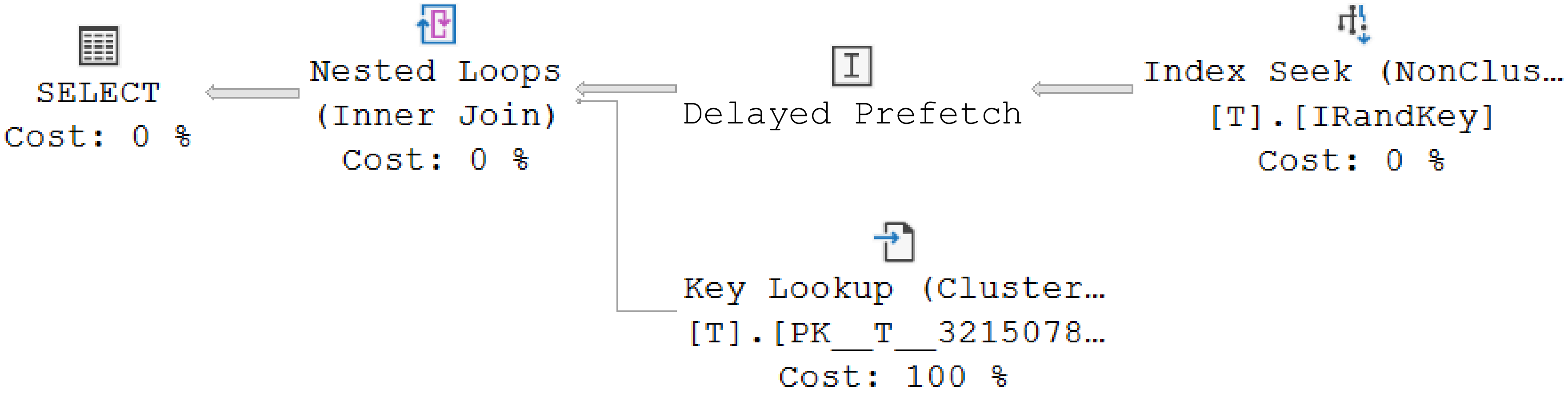


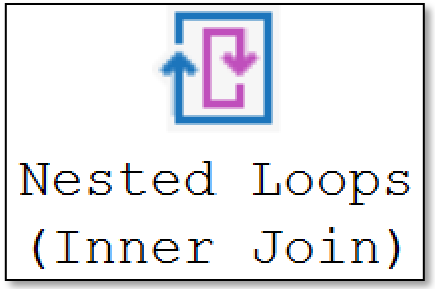


Nested Loops (advanced)

Delayed Prefetch

- Reads a few rows ahead
- Checks whether required “lower input” pages are in buffer pool
- Reads for this check are not reported in the execution plan!
- Does not check all rows once it finds most/all are already in buffer pool

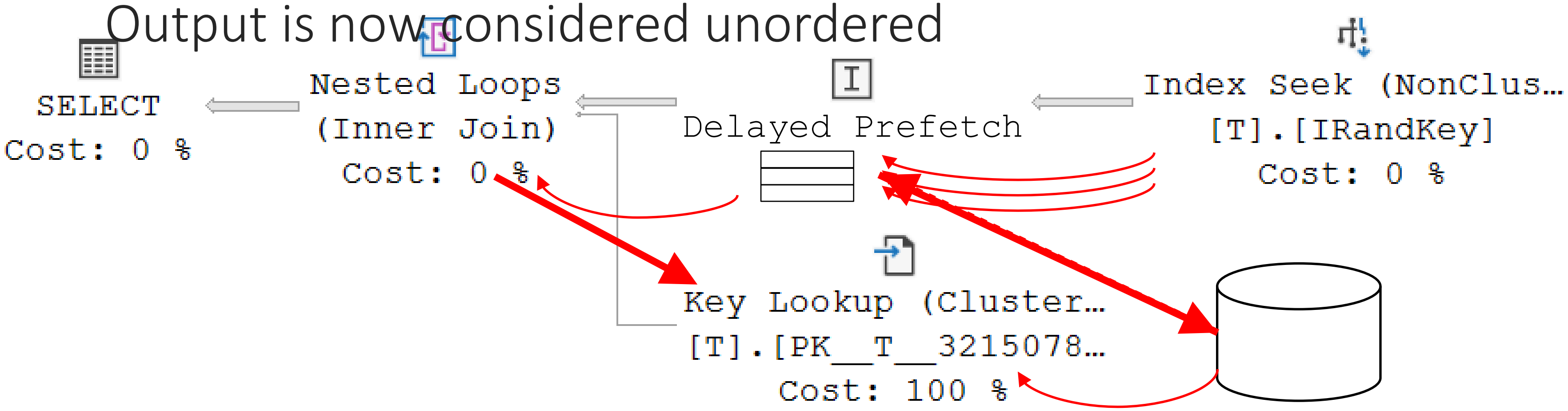


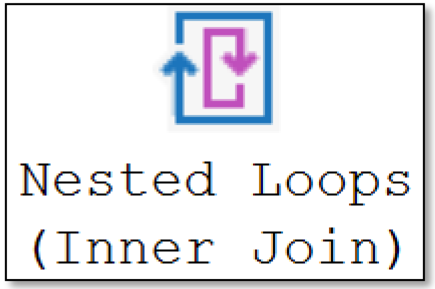


Nested Loops (advanced)

Advanced performance optimizations

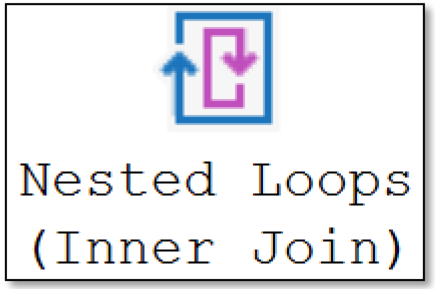
- Ordered prefetching
- Unordered prefetching
  - Property: *WithUnorderedPrefetch*
  - Delayed Prefetch returns first row for which read request completes
  - Output is now considered unordered





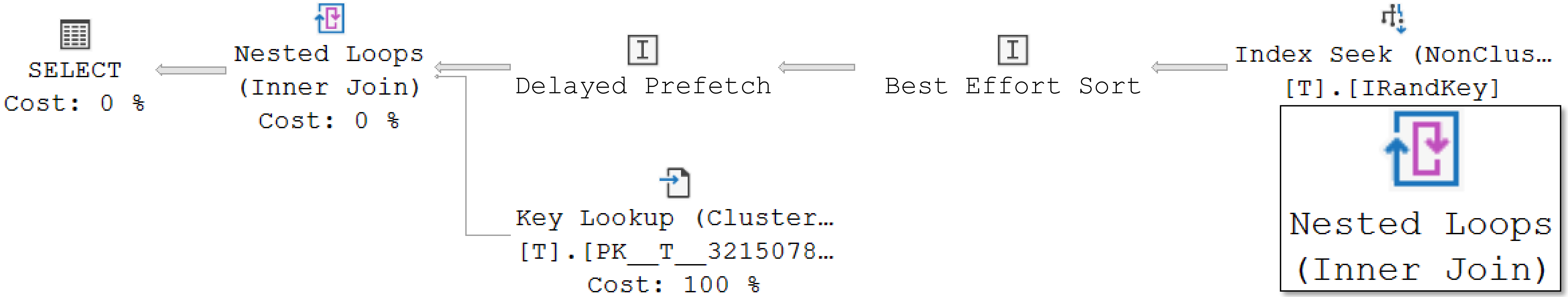
Nested Loops (advanced)

- Advanced performance optimizations
  - Ordered prefetching
  - Unordered prefetching
  - Optimized Nested Loops
    - Actively reorders data
    - Reordering based on index used in lower input
      - Increases chance that required page is still in buffer pool
      - Pattern of physical I/Os optimized for reading from spinning disk



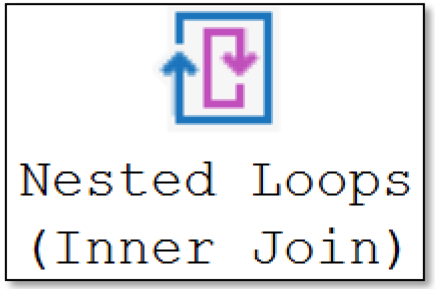
Nested Loops (advanced)

- Advanced performance optimizations
  - Ordered prefetching
  - Unordered prefetching
  - Optimized Nested Loops
    - When out of memory, no tempdb spill



Nested Loops (advanced)

- Advanced performance optimizations
  - Ordered prefetching
  - Unordered prefetching
  - Optimized Nested Loops
    - When out of memory, no tempdb spill, but sorts multiple batches
    - Output is considered unordered
    - Often combined with unordered prefetching, never with ordered prefetching
    - Operator is now blocking
      - But "semi blocking", not fully blocking



Nested Loops
(Inner Join)

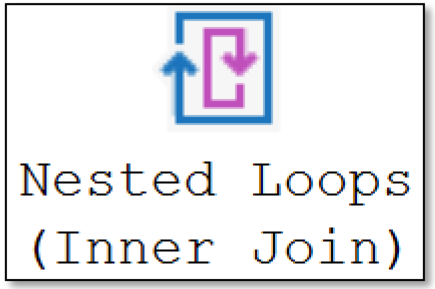
Nested Loops (advanced)

Advanced performance optimizations

- Ordered prefetching
- Unordered prefetching
- Optimized Nested Loops

All these optimizations require a lower side input with:

- Key Lookup, RID Lookup, Clustered Index Seek, or Index Seek
- Possibly some more complex cases as well



Summary

- Nested Loops (advanced)
  - Logic for all supported join types
  - Workarounds for unsupported join types
  - Optimizations:
    - Prefetching (ordered and unordered)
    - Optimized Nested Loops

Next chapters

- Chapter 2: Merge Join (advanced)
  - Logic for all supported join types
  - Many-to-many Merge Join
  - Some special cases
- Chapter 3: Hash Match (advanced)
- Chapter 4: Adaptive Join (advanced)