

**WHO TAKES OUT
YOUR TRASH?**

@sannekalkman







Magic is just (computer) science you don't understand yet







Allocate memory



Allocate memory



Find the garbage



Allocate memory



Find the garbage



Free up memory



Running out of memory

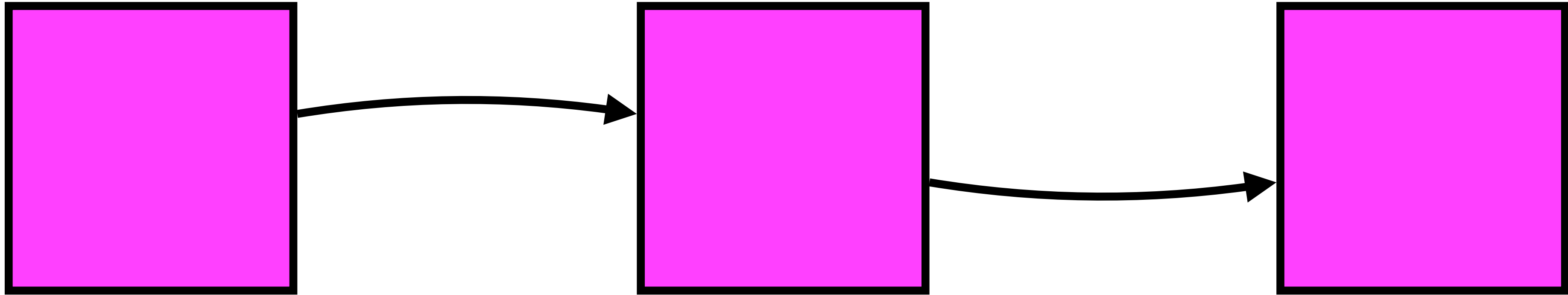


Running out of memory

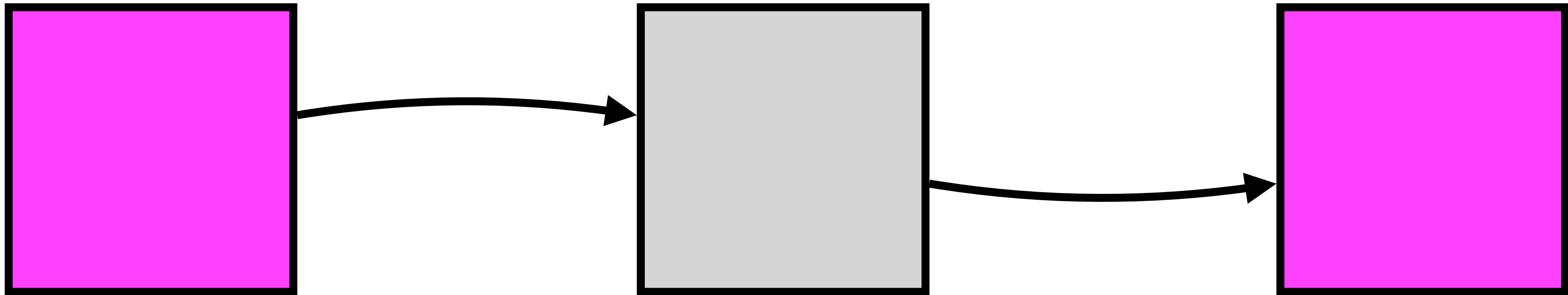


Breaking things

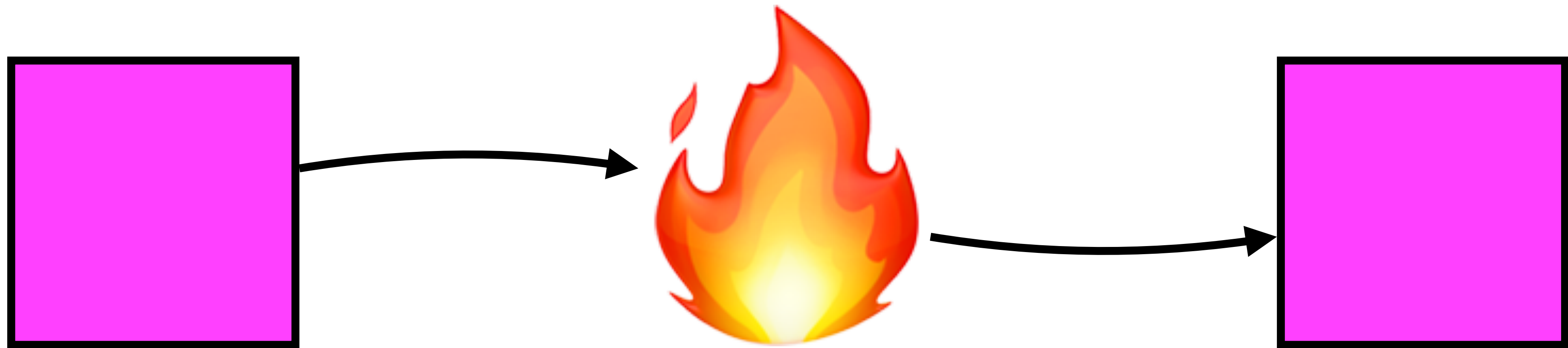
DANGLING POINTERS



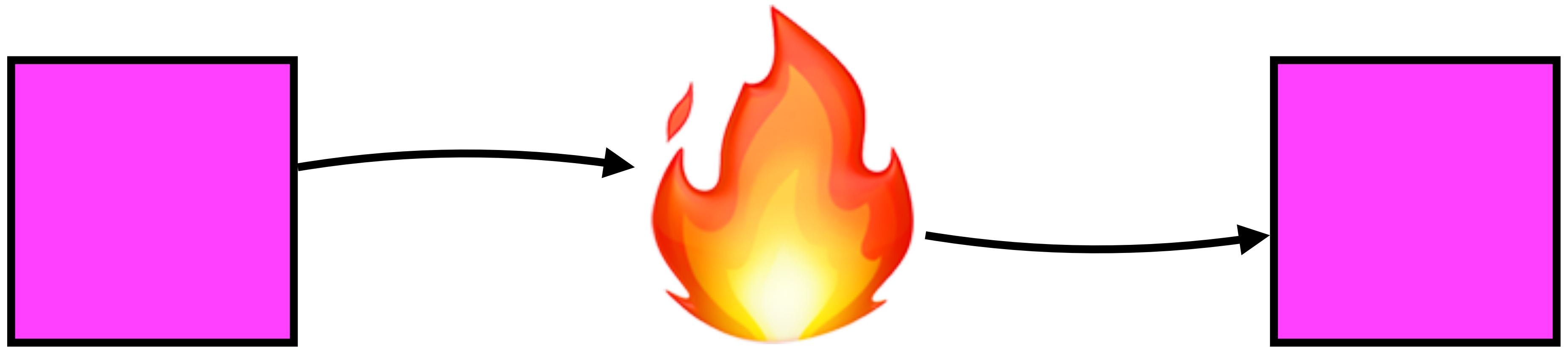
DANGLING POINTERS



DANGLING POINTERS



DANGLING POINTERS



Memory Leak

REFERENCE COUNTING

If nothing points to it, nothing's using it

REFERENCE COUNTING

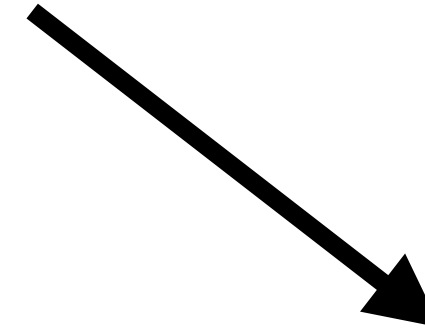
REFERENCE COUNTING

```
name = "codeBEAM"
```

REFERENCE COUNTING

name = "codeBEAM"

name

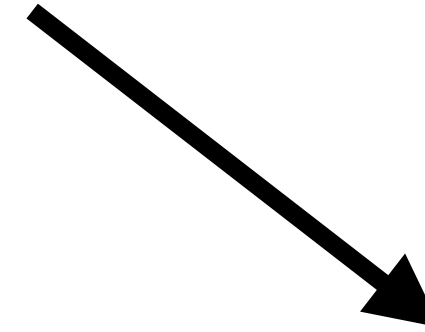


"codeBEAM"	1
------------	---

REFERENCE COUNTING

```
name = "codeBEAM"  
other_name = name
```

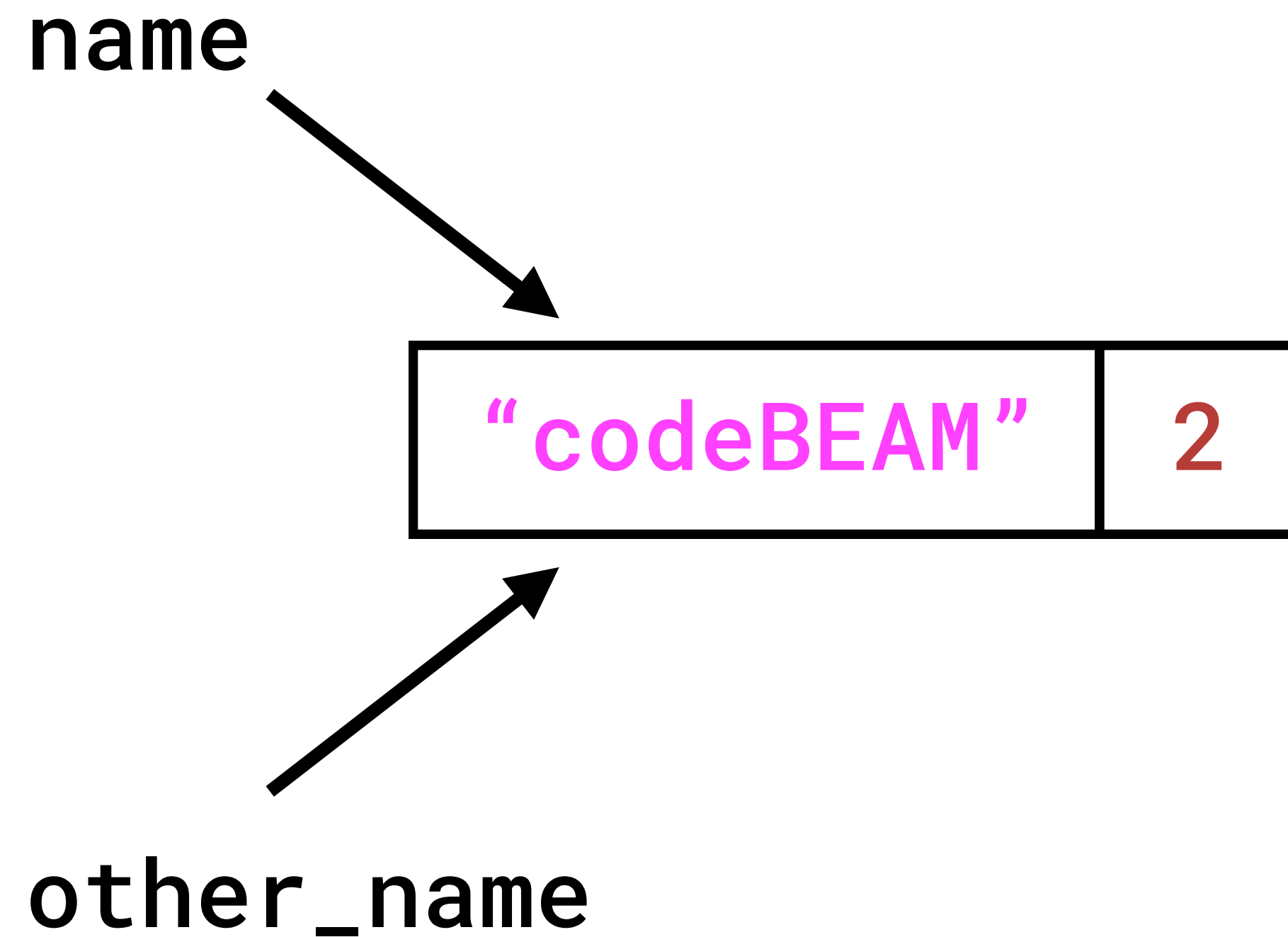
name



"codeBEAM"	1
------------	---

REFERENCE COUNTING

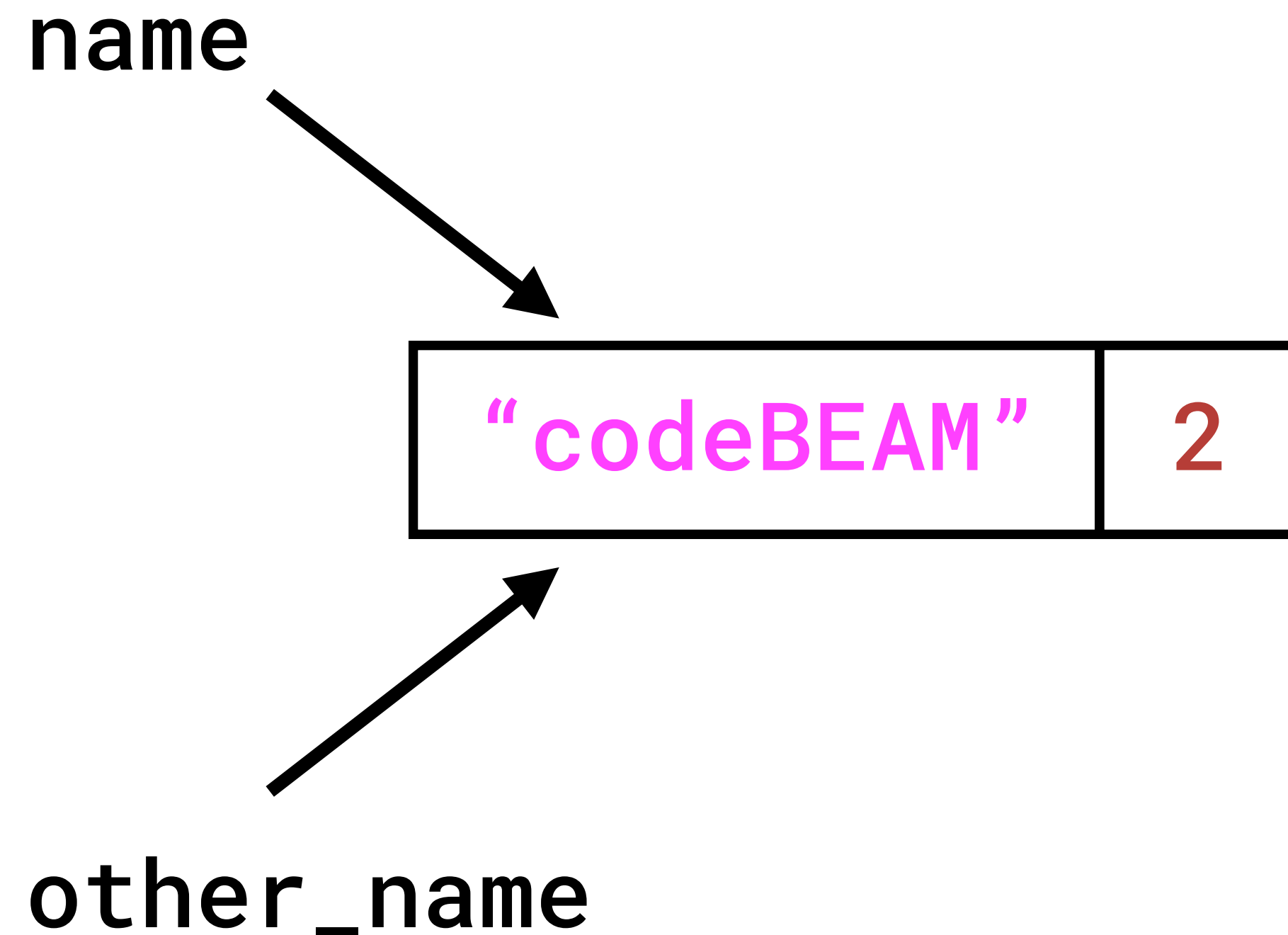
```
name = "codeBEAM"  
other_name = name
```



REFERENCE COUNTING

```
name = "codeBEAM"  
other_name = name
```

```
name = nil
```



REFERENCE COUNTING

```
name = "codeBEAM"  
other_name = name  
  
name = nil
```

name



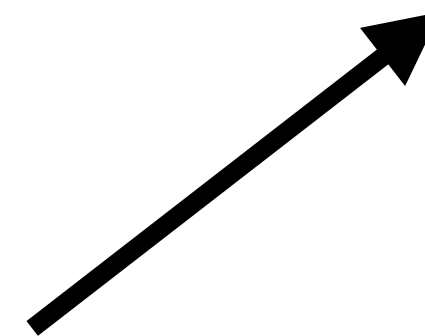
other_name

REFERENCE COUNTING

name

```
name = "codeBEAM"  
other_name = name
```

```
name = nil  
other_name = nil
```



other_name

REFERENCE COUNTING

name

```
name = "codeBEAM"  
other_name = name
```

```
name = nil  
other_name = nil
```

"codeBEAM"	0
------------	---

other_name

REFERENCE COUNTING

name

```
name = "codeBEAM"  
other_name = name
```

```
name = nil  
other_name = nil
```



other_name

CYCLES

CYCLES

```
a = {  
  other: nil  
}
```

CYCLES

```
a = {  
  other: nil  
}
```



CYCLES

```
a = {  
  other: nil  
}
```

```
b = {  
  other: a  
}
```



CYCLES

```
a = {  
  other: nil  
}
```



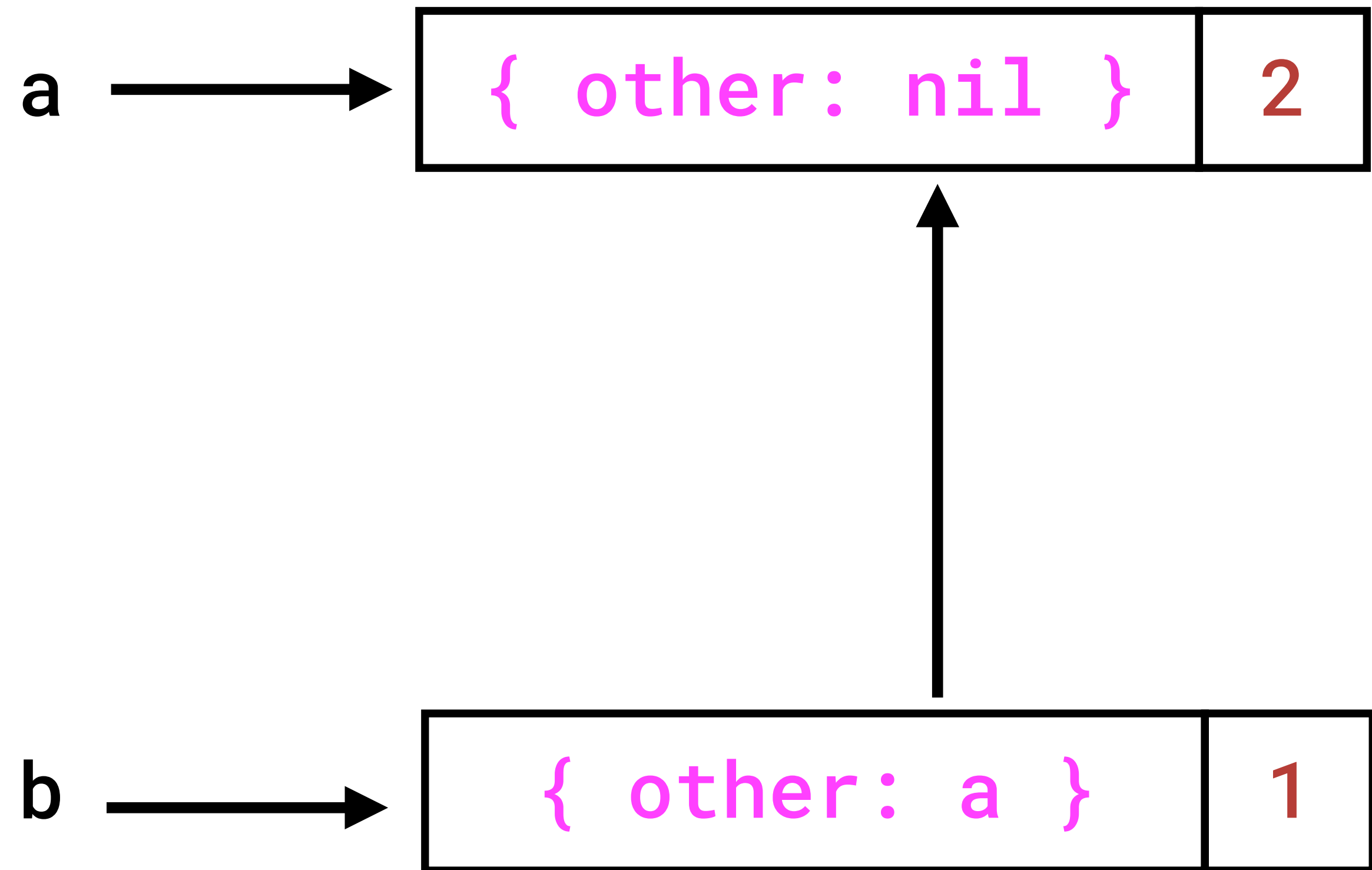
```
b = {  
  other: a  
}
```



CYCLES

```
a = {  
  other: nil  
}
```

```
b = {  
  other: a  
}
```

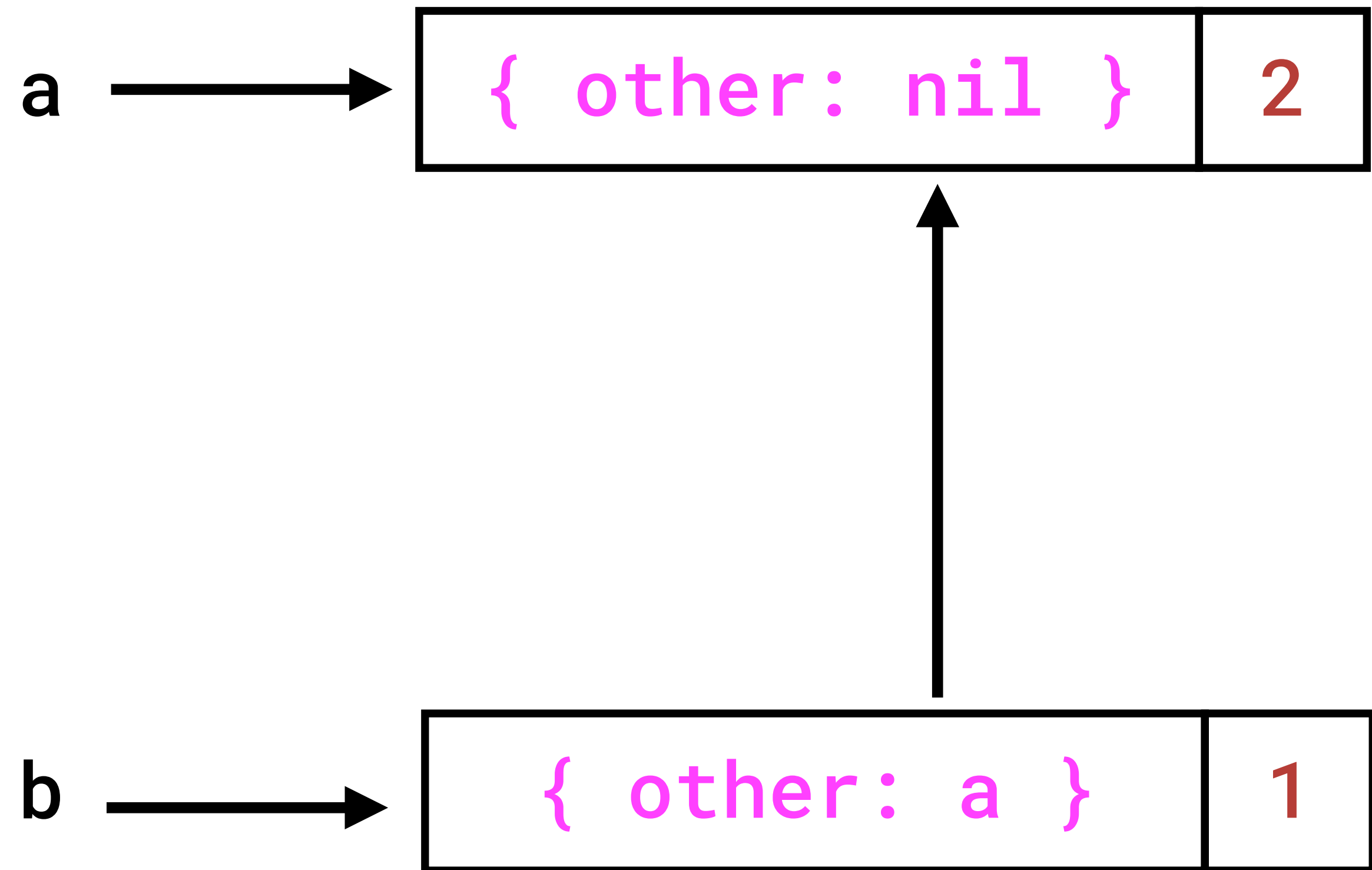


CYCLES

```
a = {  
  other: nil  
}
```

```
b = {  
  other: a  
}
```

```
a[:other] = b
```

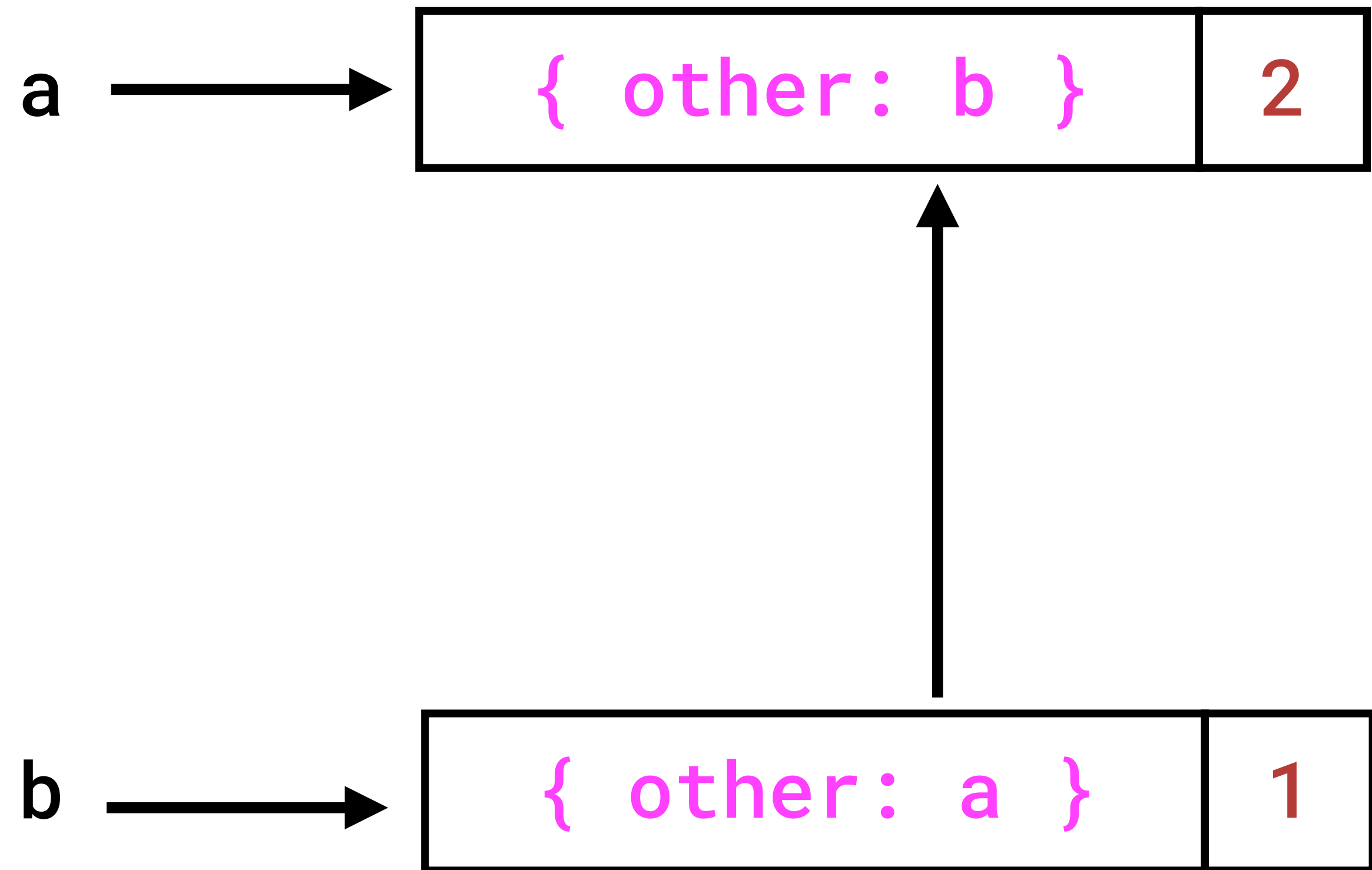


CYCLES

```
a = {  
  other: nil  
}
```

```
b = {  
  other: a  
}
```

```
a[:other] = b
```

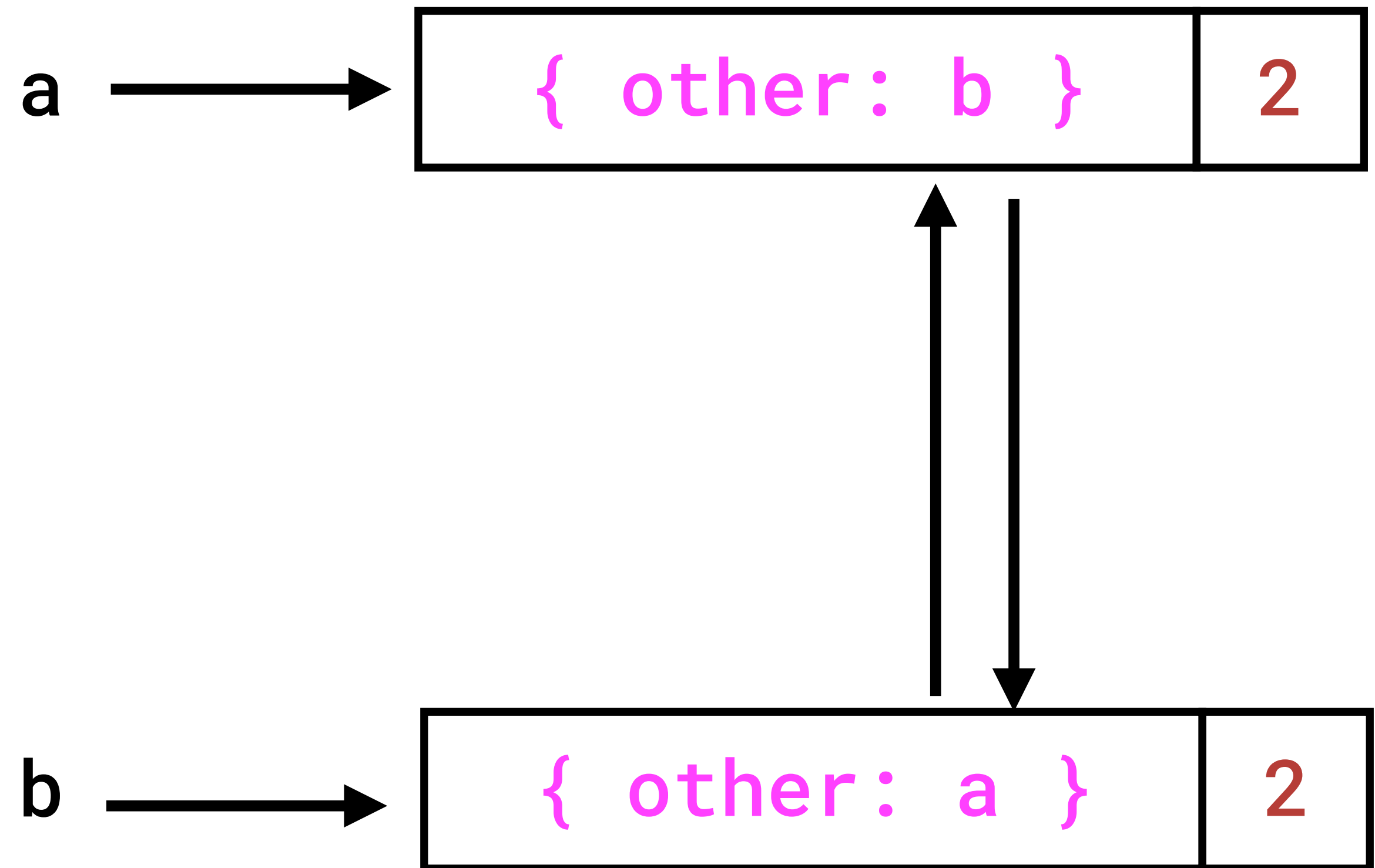


CYCLES

```
a = {  
  other: nil  
}
```

```
b = {  
  other: a  
}
```

```
a[:other] = b
```



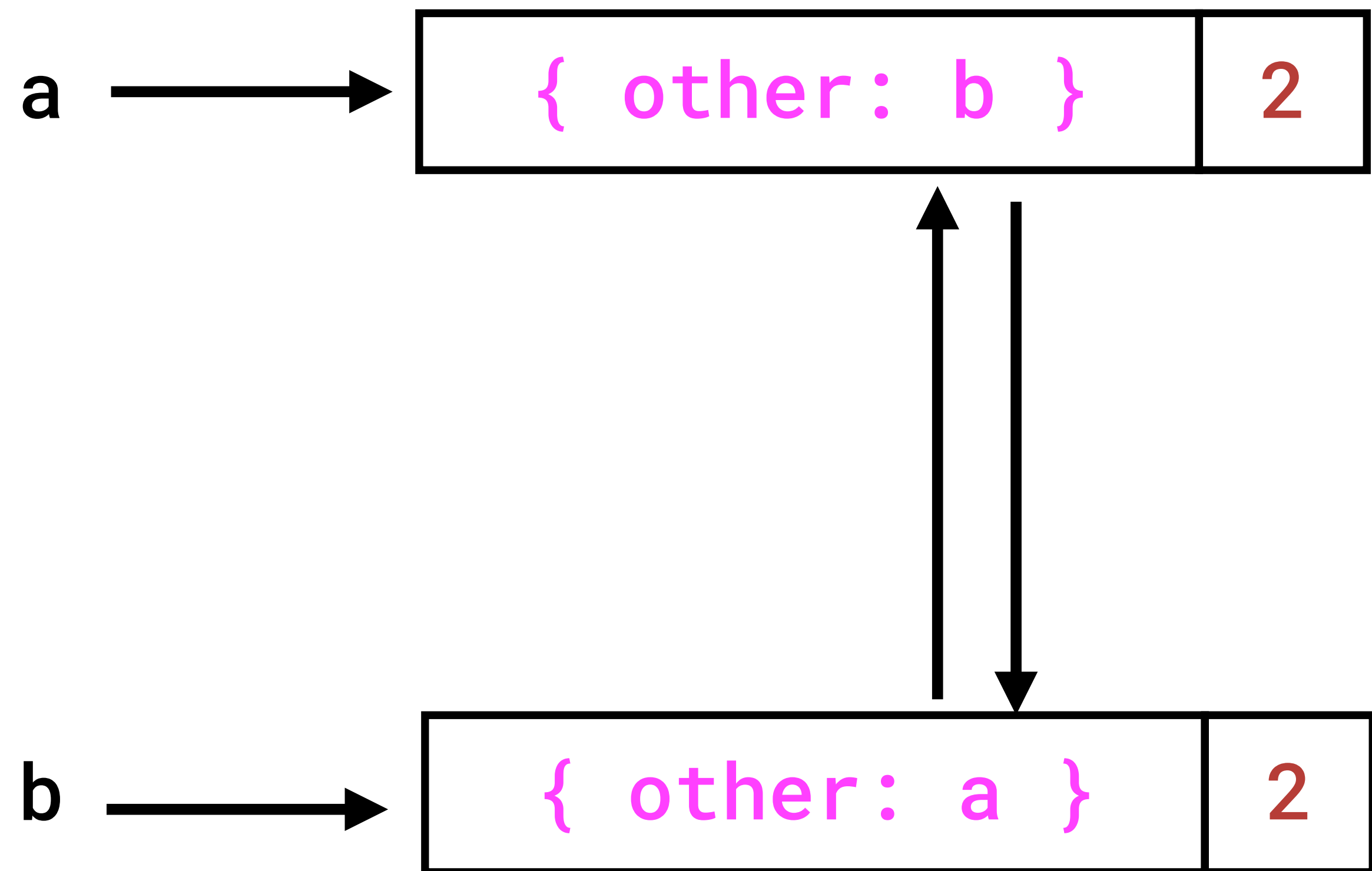
CYCLES

```
a = {  
  other: nil  
}
```

```
b = {  
  other: a  
}
```

```
a[:other] = b
```

```
a = nil  
b = nil
```



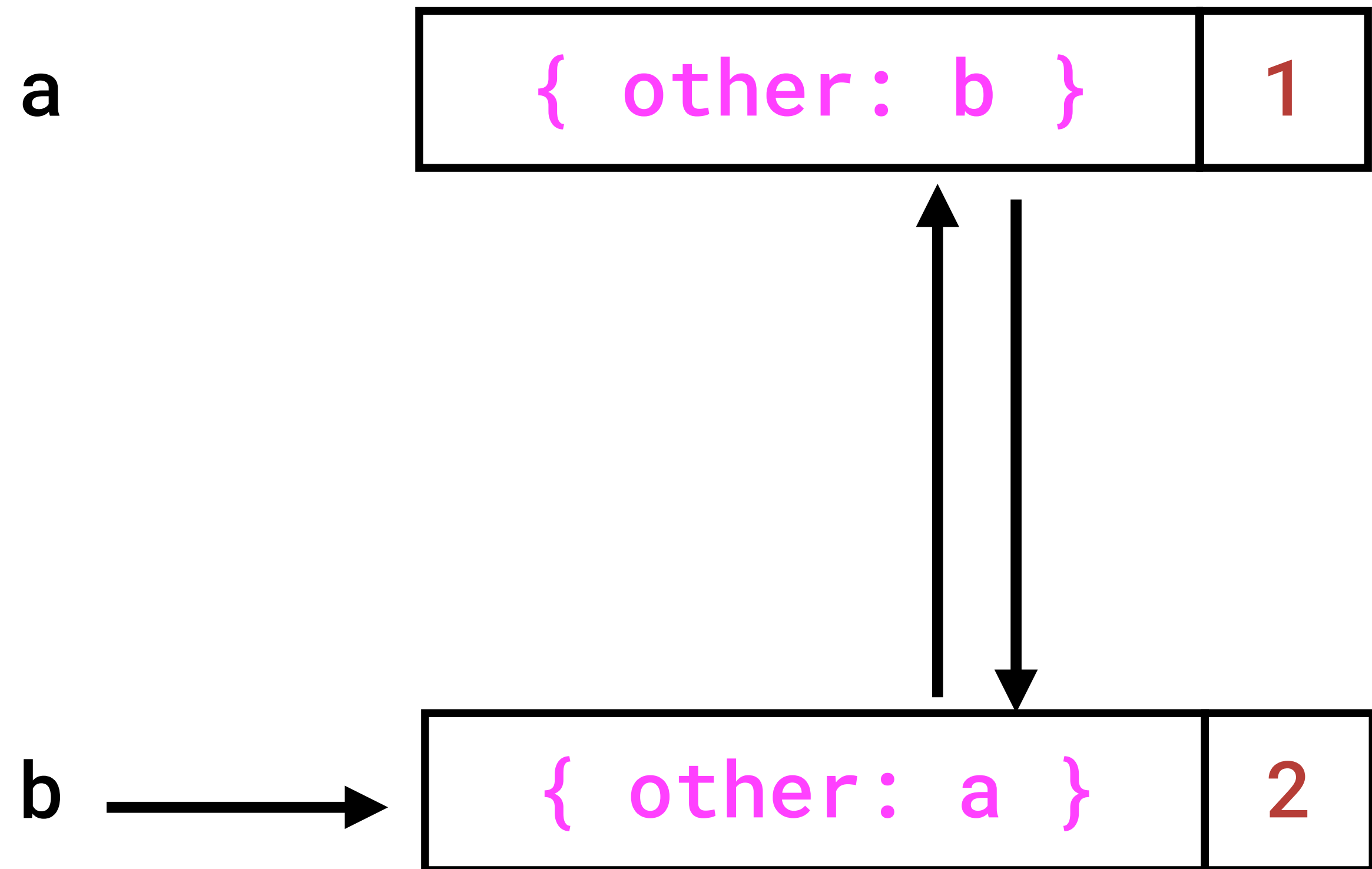
CYCLES

```
a = {  
  other: nil  
}
```

```
b = {  
  other: a  
}
```

```
a[:other] = b
```

```
a = nil  
b = nil
```



CYCLES

```
a = {  
  other: nil  
}
```

```
b = {  
  other: a  
}
```

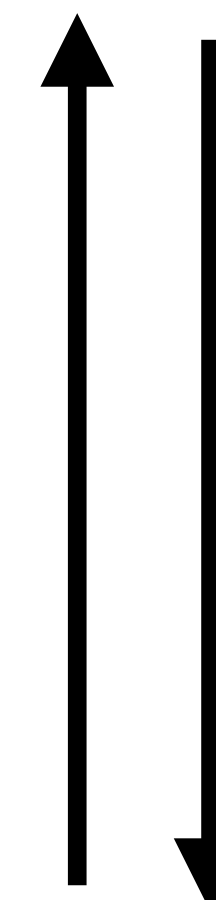
```
a[:other] = b
```

```
a = nil  
b = nil
```

a



b



MARK & SWEEP

If you can't get to it, you can't be using it



NY - NO

PO - PBA

ROA - ROM

TOR - UZ

YOD - Z

W - WSC

PRK - Q

SONN - STA

WIC - WJ

R - RJO

STE - SU

WAN - WED

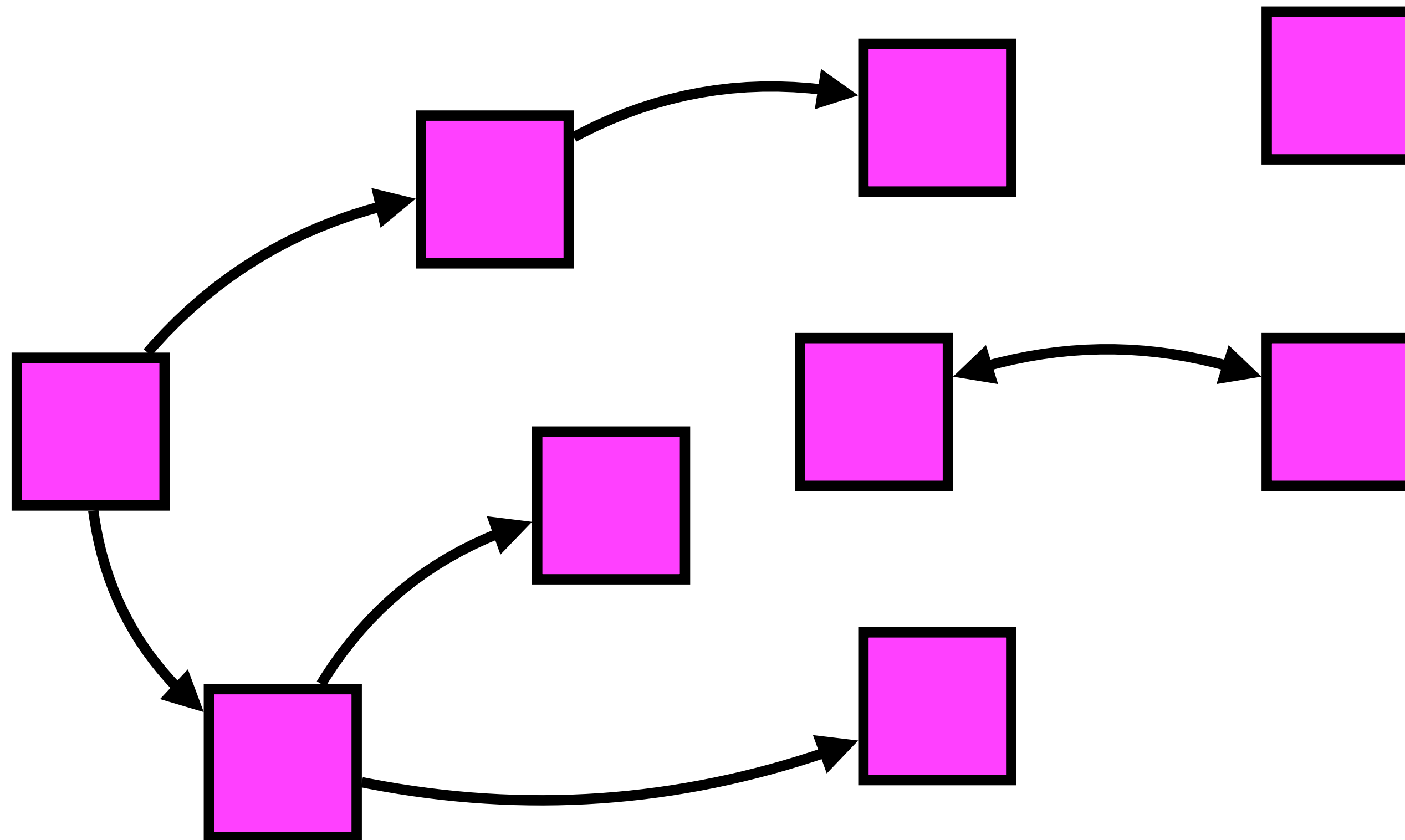
W - WJO

KST - ROR

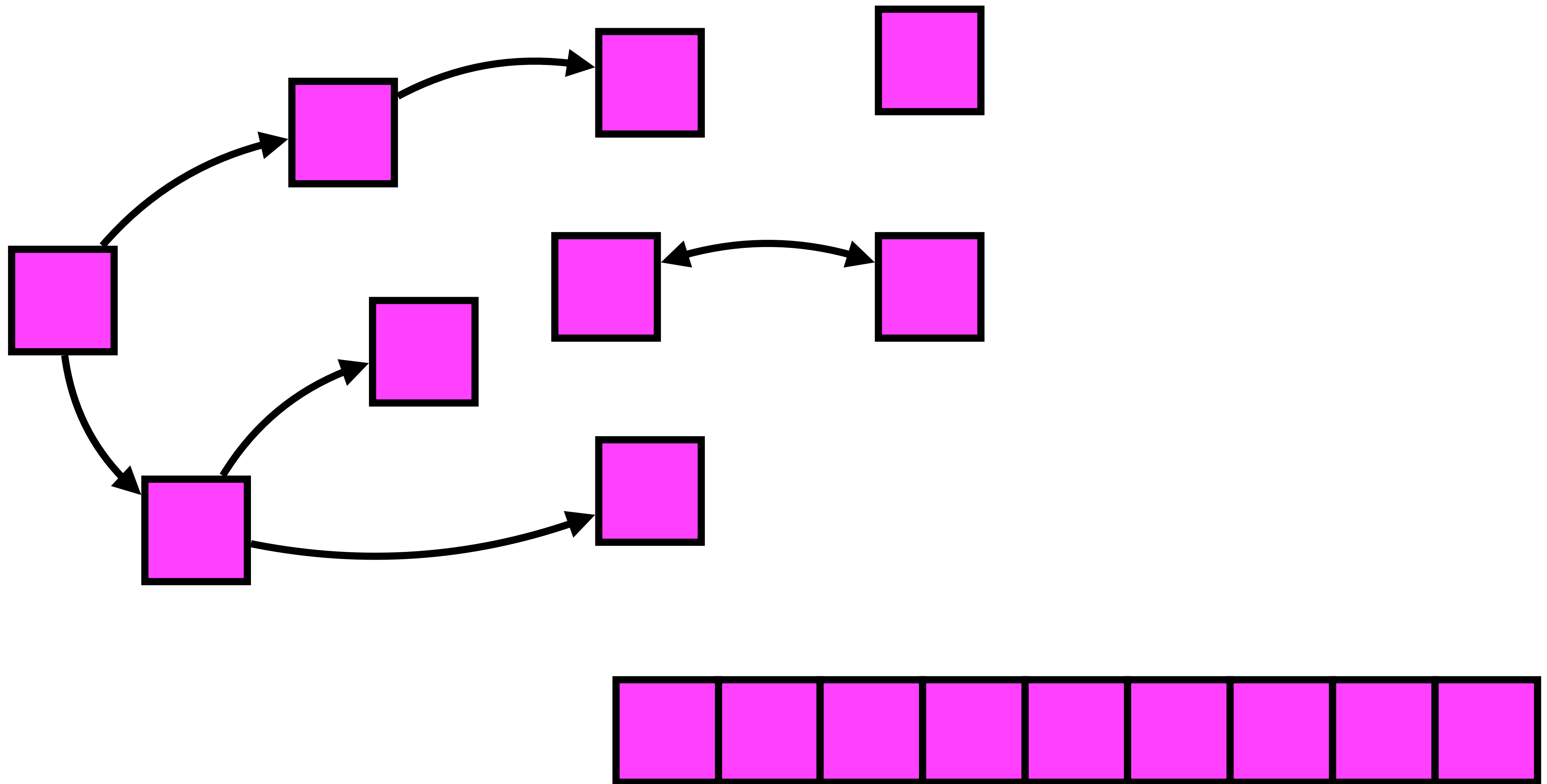
SW - SZ

WEE - WHA

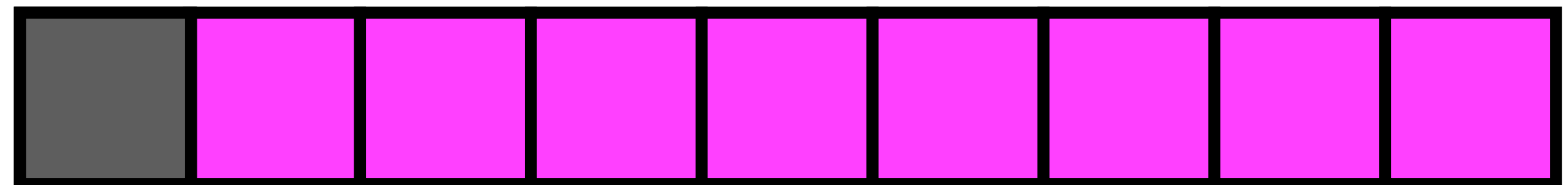
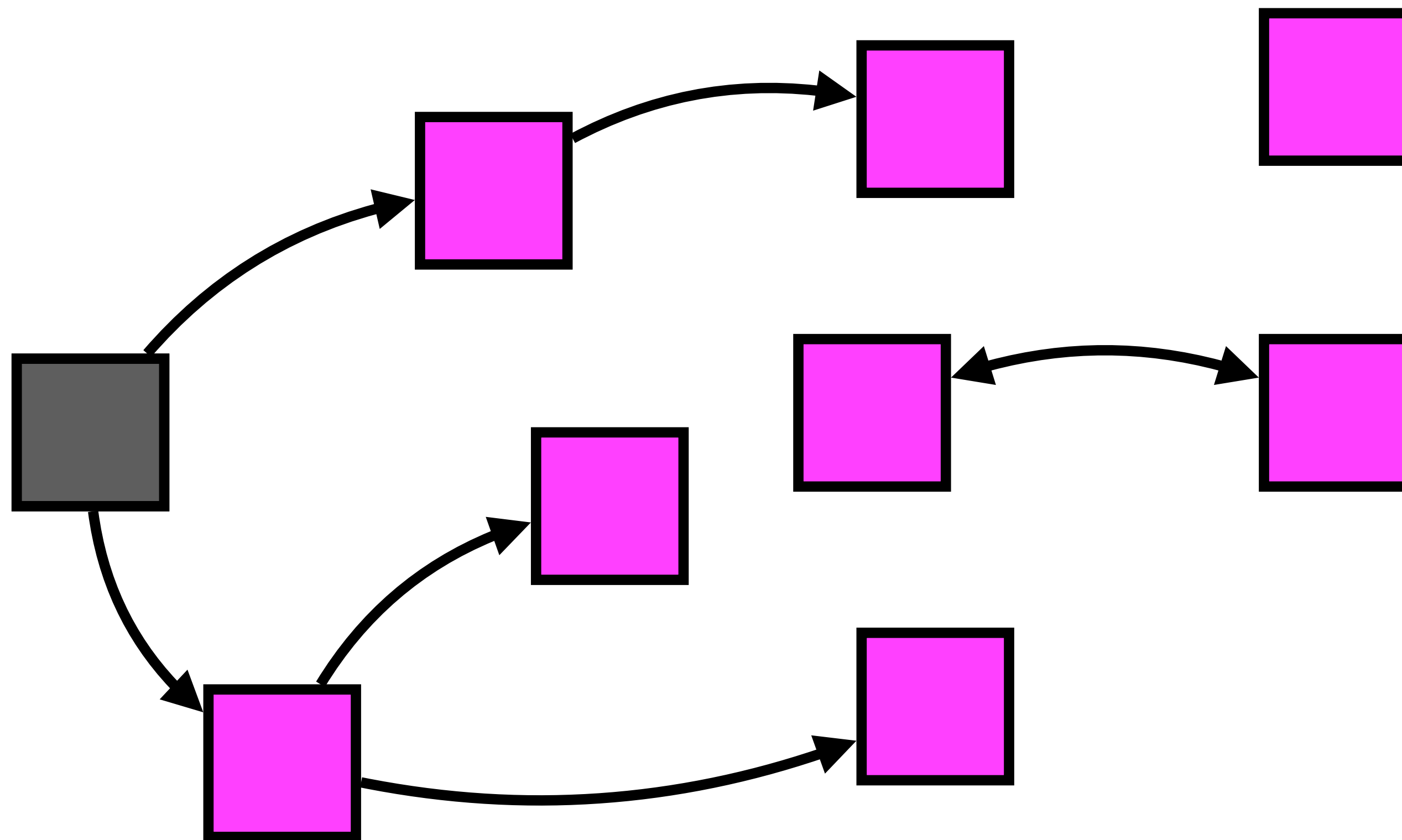
MARK & SWEEP



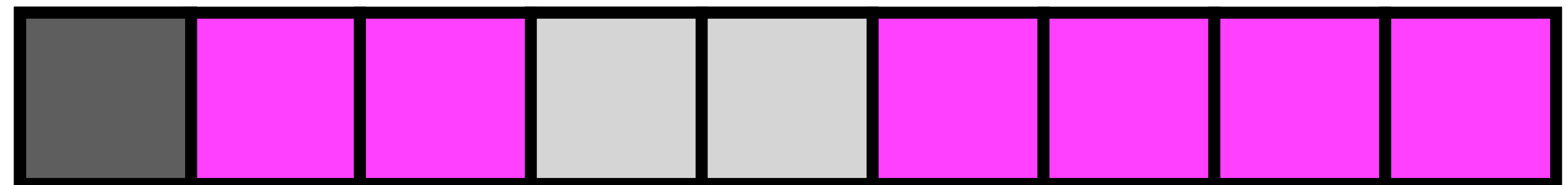
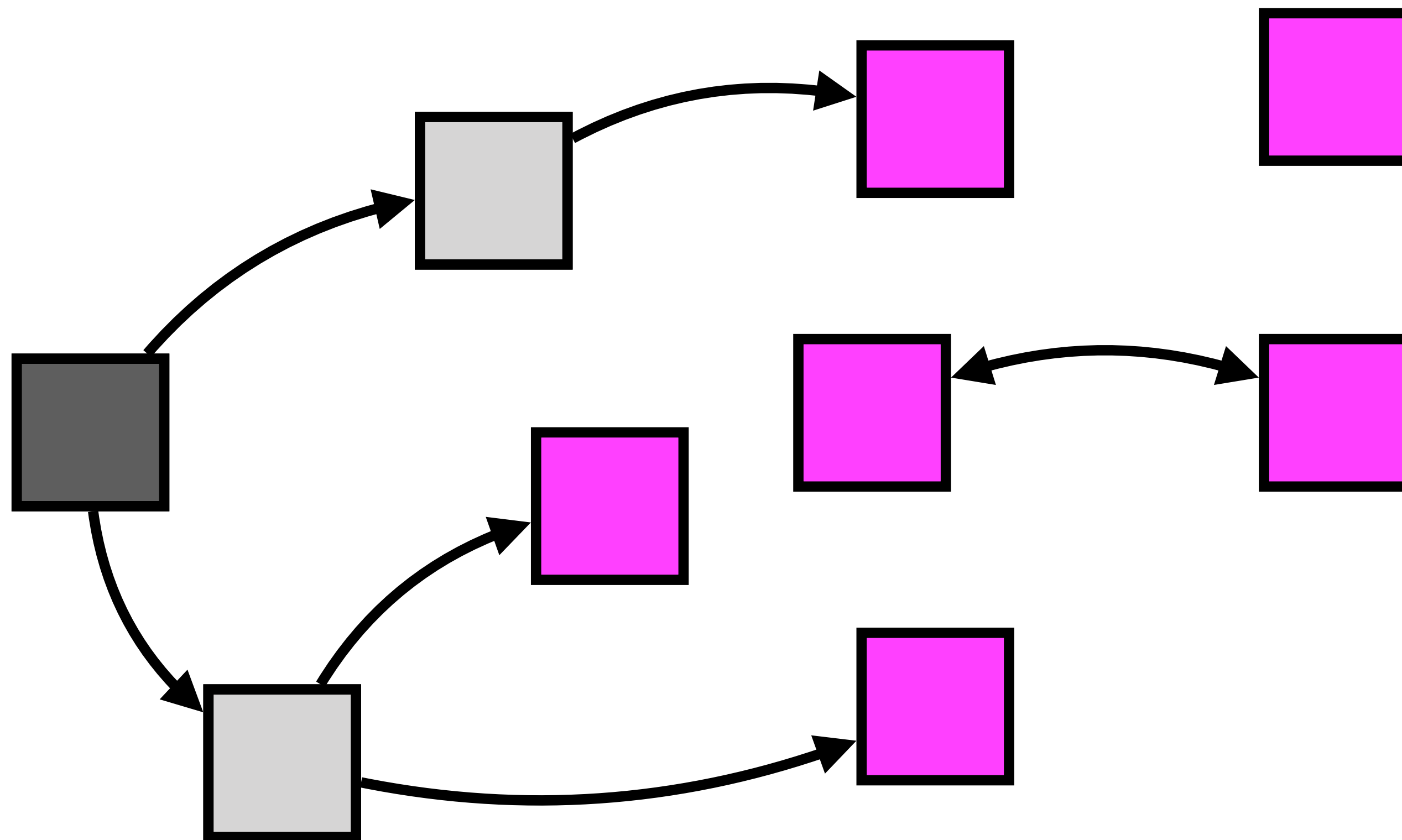
MARK & SWEEP



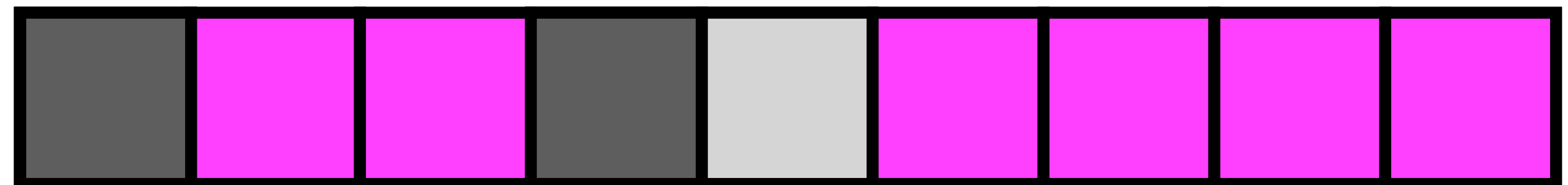
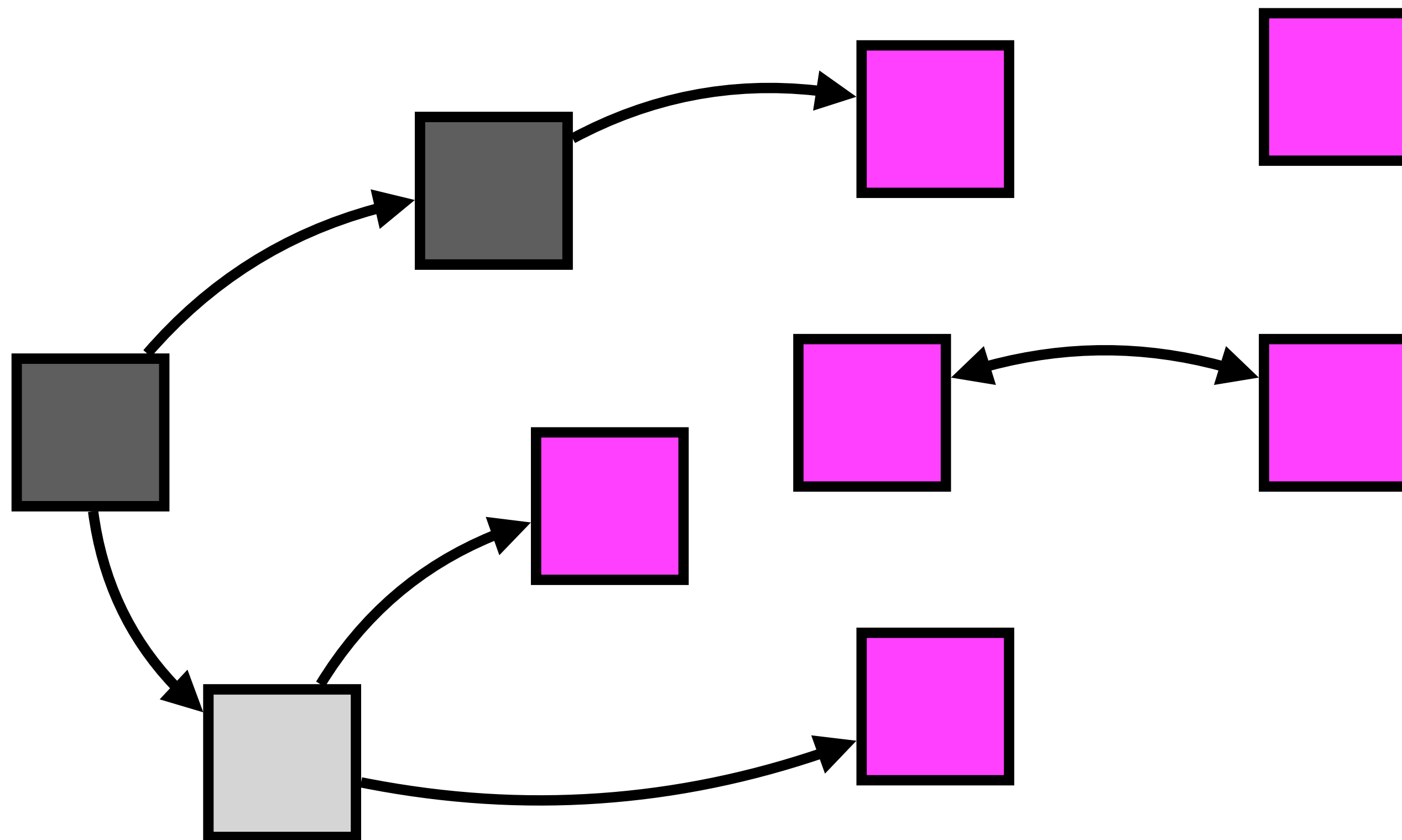
MARK & SWEEP



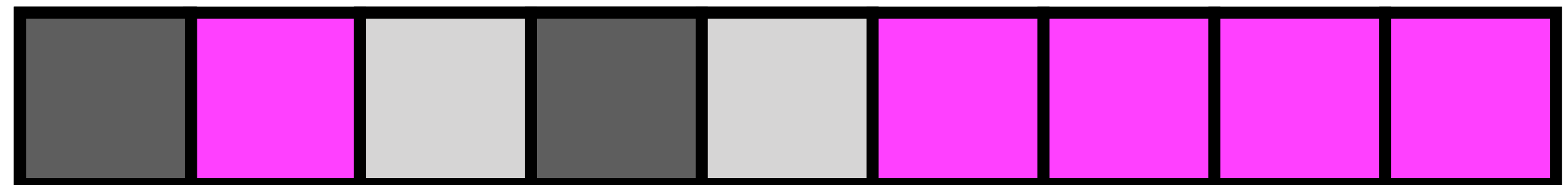
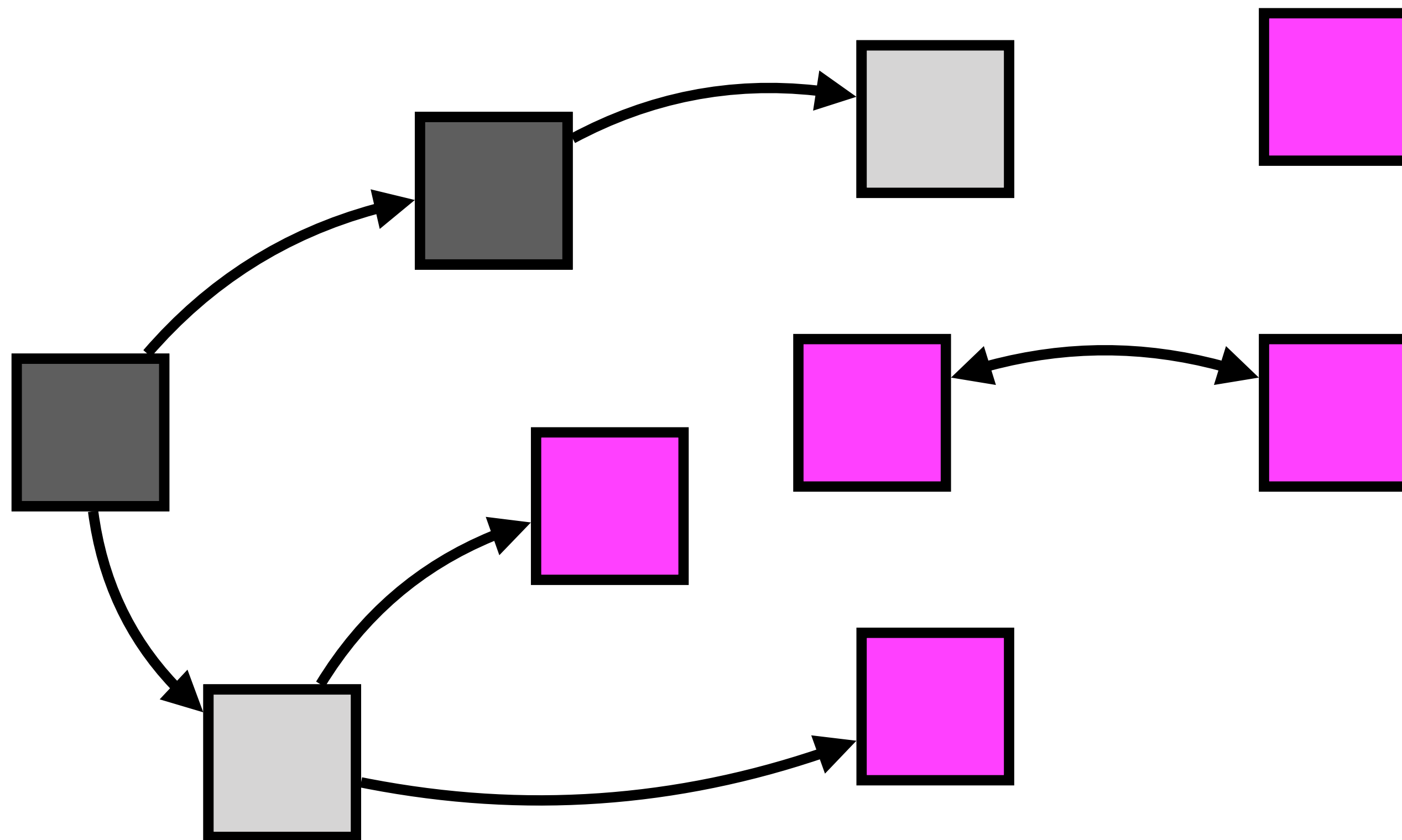
MARK & SWEEP



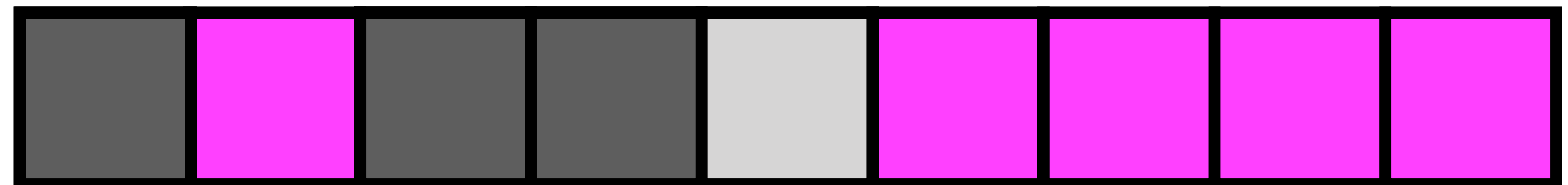
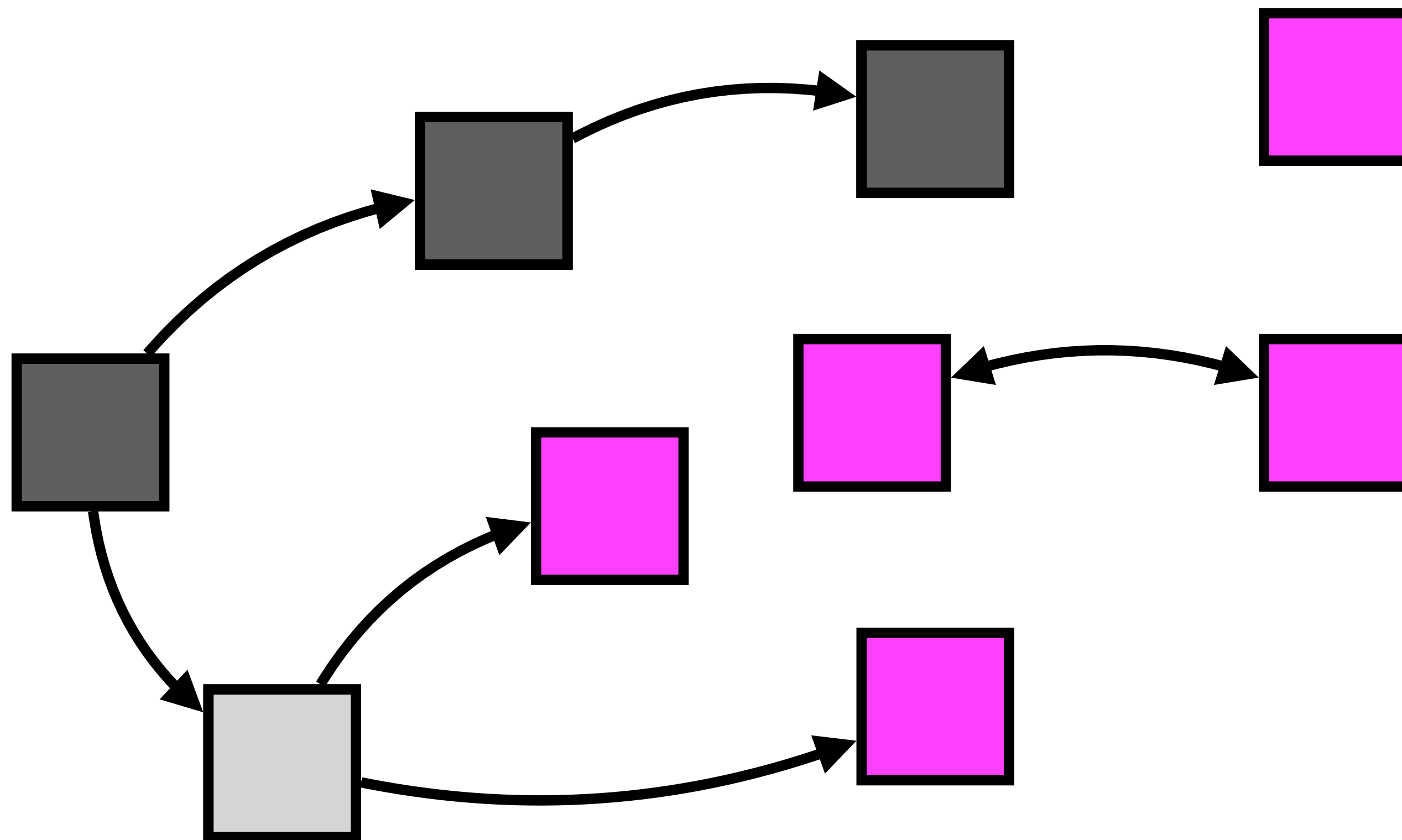
MARK & SWEEP



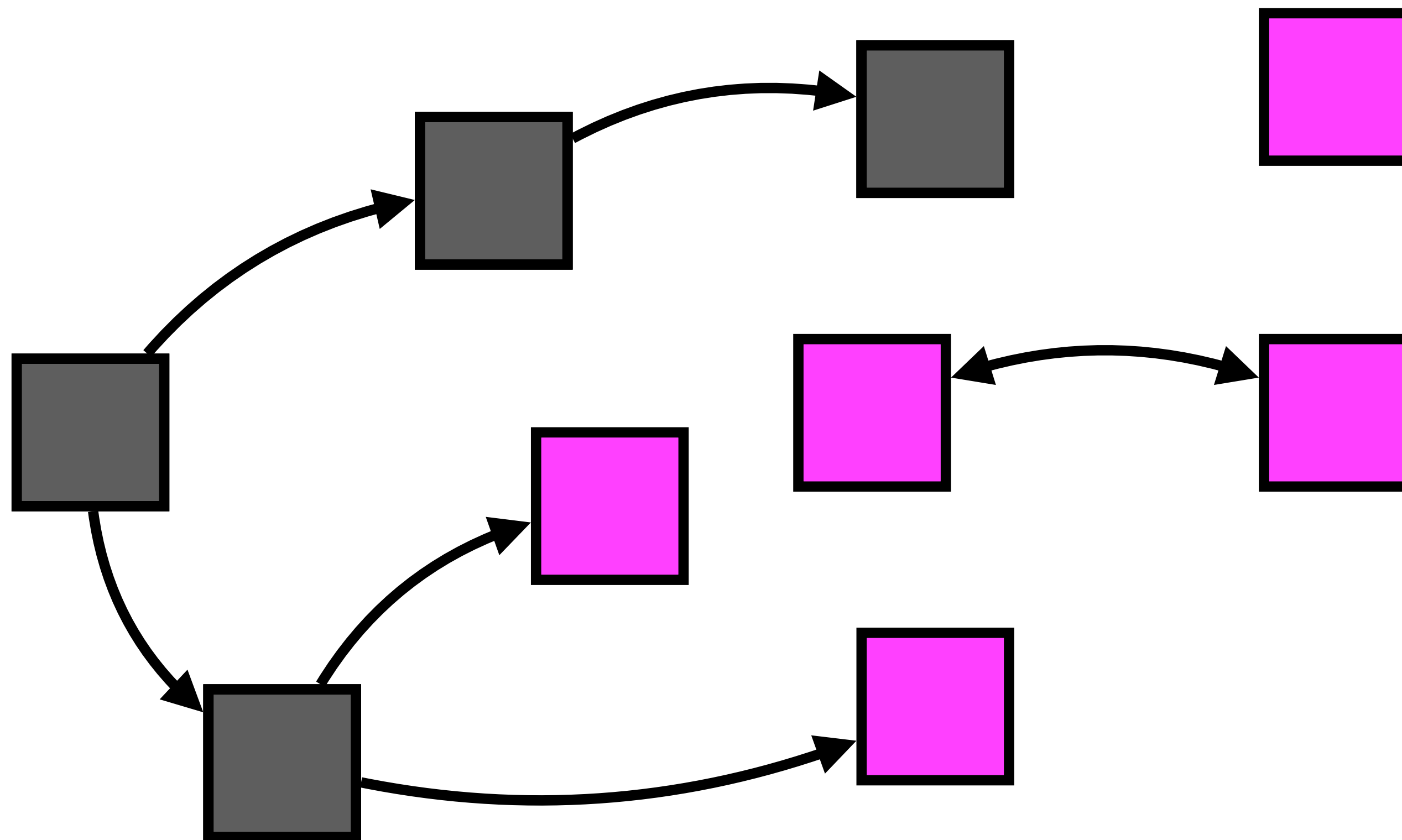
MARK & SWEEP



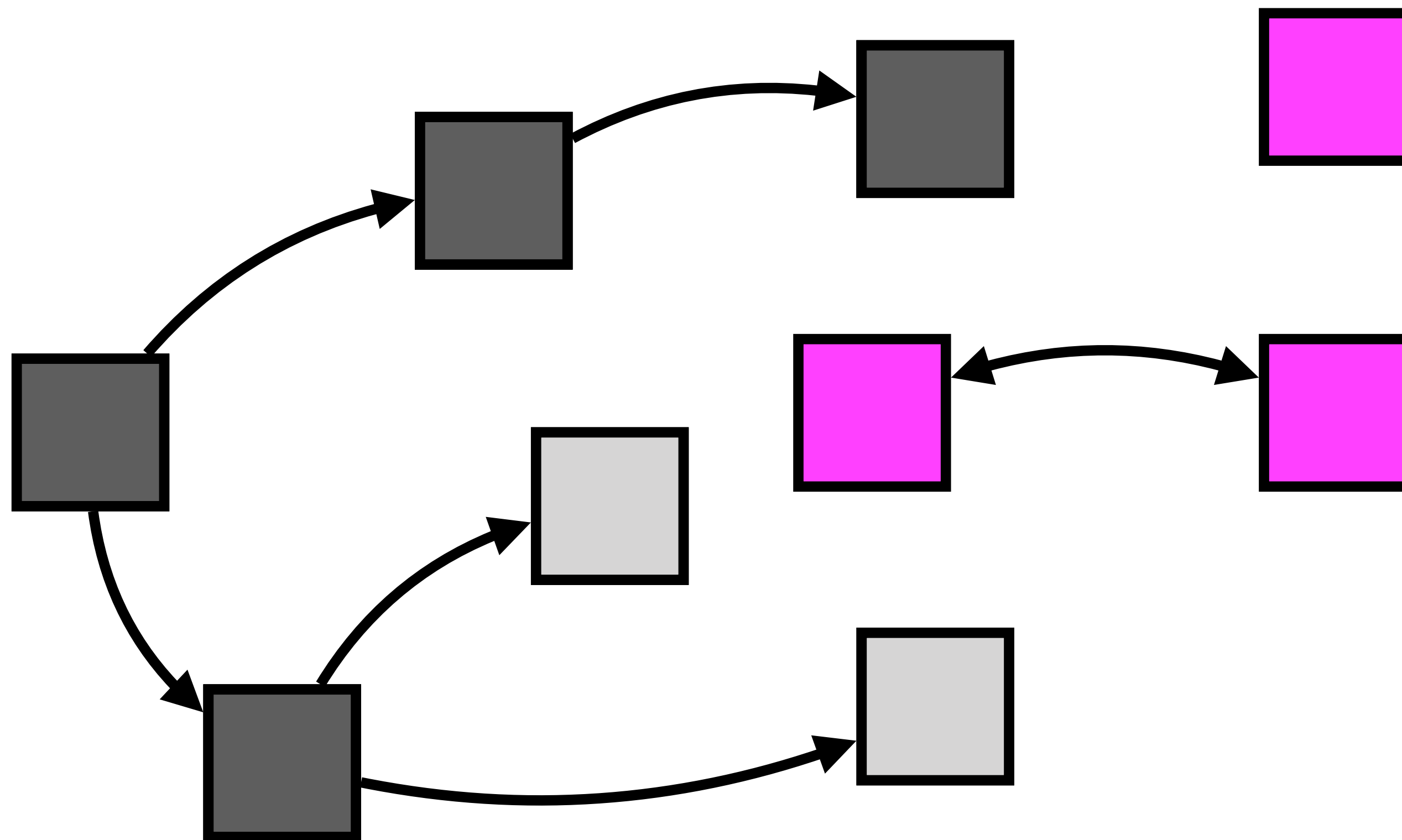
MARK & SWEEP



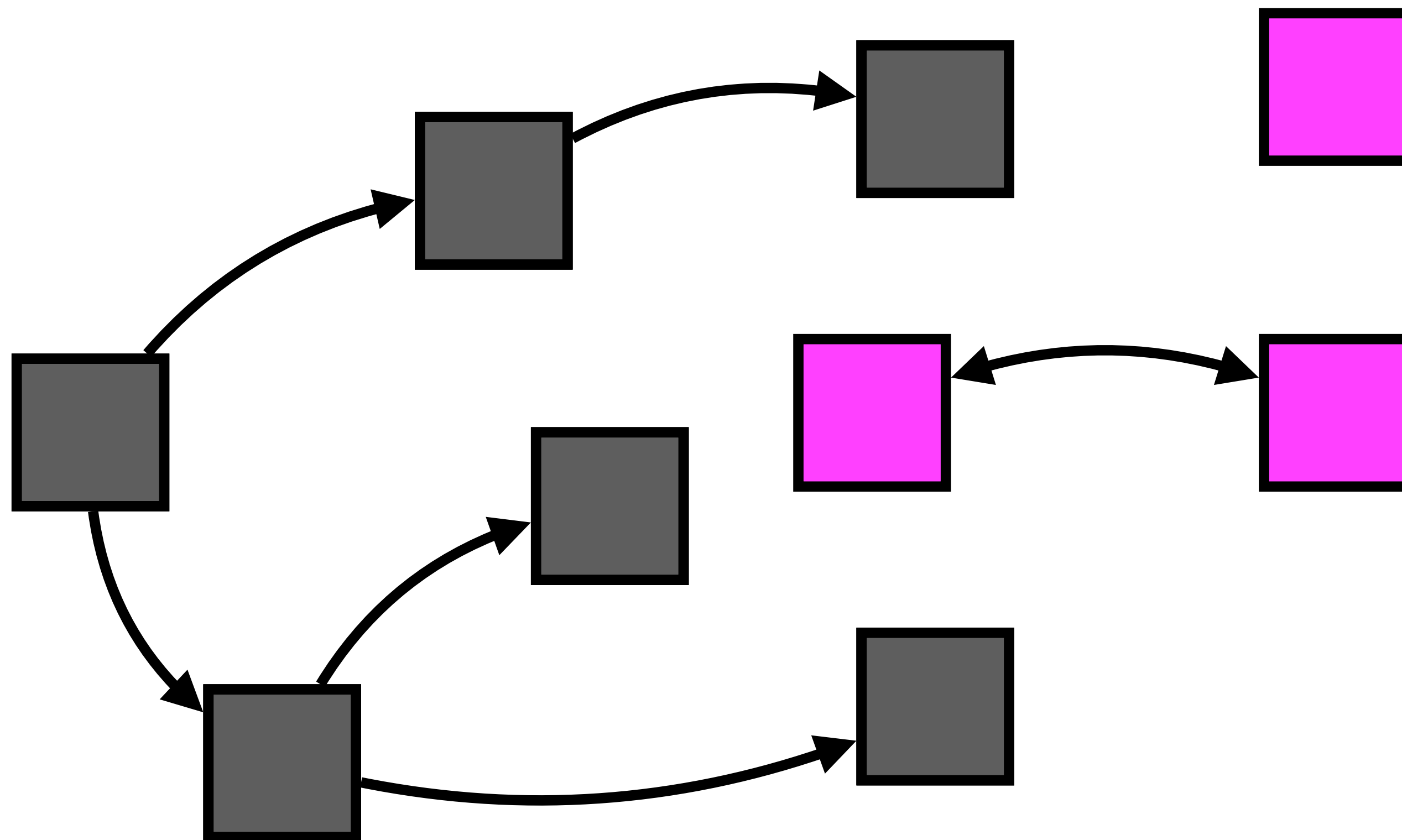
MARK & SWEEP



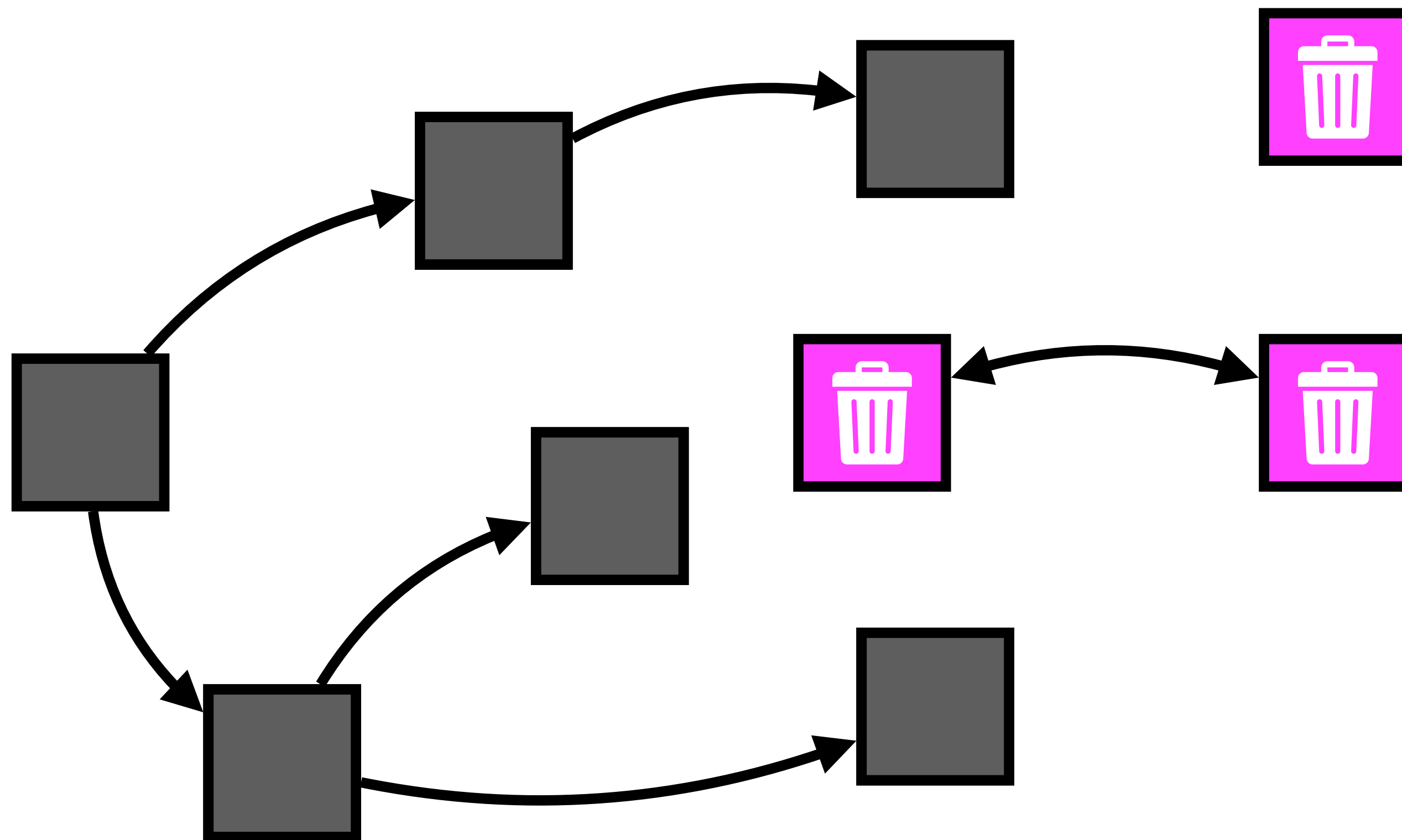
MARK & SWEEP



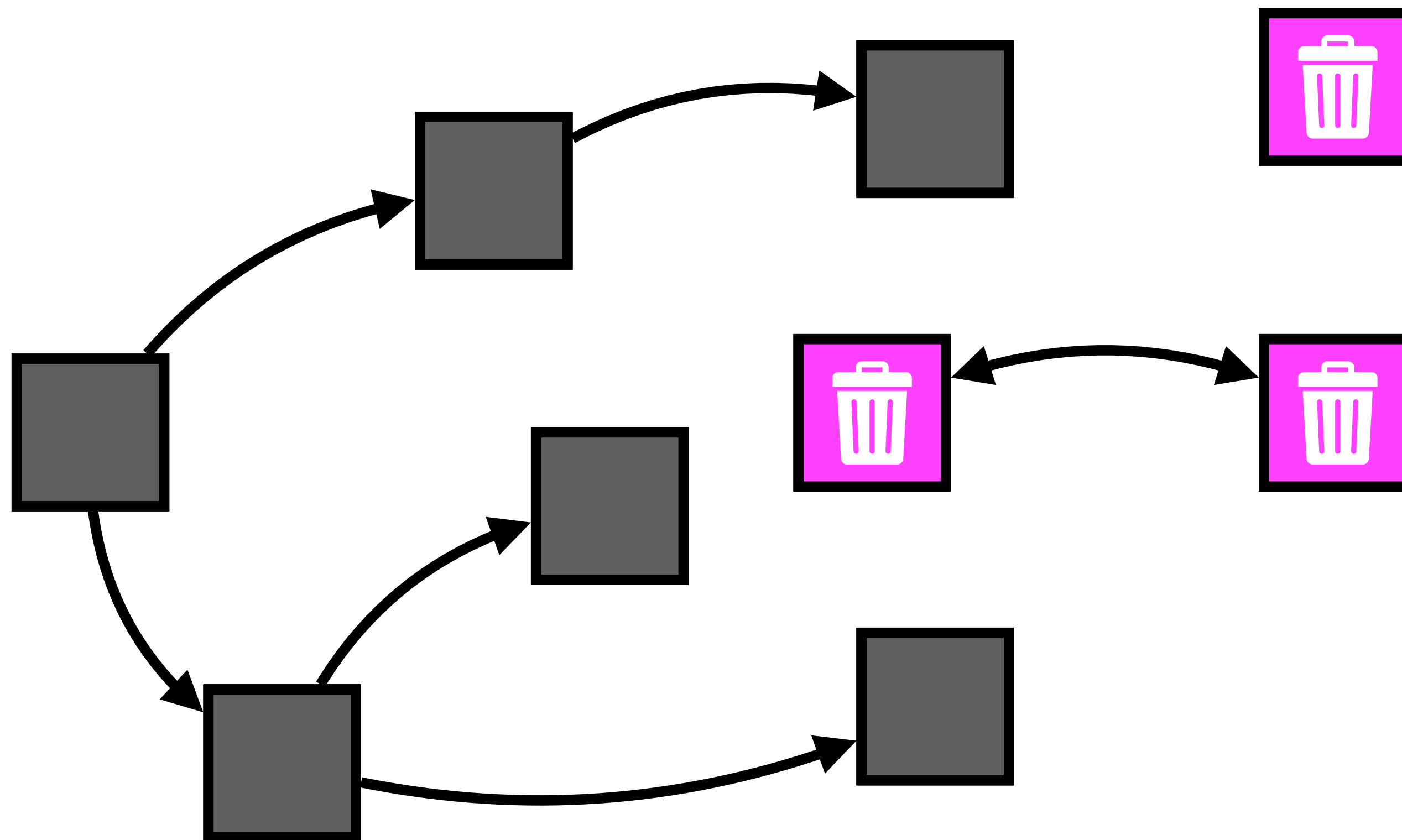
MARK & SWEEP



MARK & SWEEP



MARK & SWEEP



COPYING COLLECTION

Keep your stuff together

COPYING COLLECTION

FROM Space

0	1	2	3	4	5	6	7	8
---	---	---	---	---	---	---	---	---

TO Space

--	--	--	--	--	--	--	--	--

COPYING COLLECTION

FROM Space

0	1	2	3	4	5	6	7	8
---	---	---	---	---	---	---	---	---

TO Space

0								
---	--	--	--	--	--	--	--	--

COPYING COLLECTION

FROM Space

0	1	2	3	4	5	6	7	8
---	---	---	---	---	---	---	---	---

TO Space

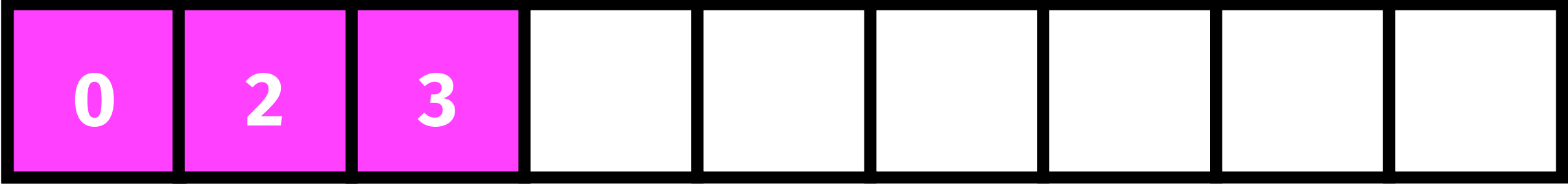
0	2							
---	---	--	--	--	--	--	--	--

COPYING COLLECTION

FROM Space



TO Space

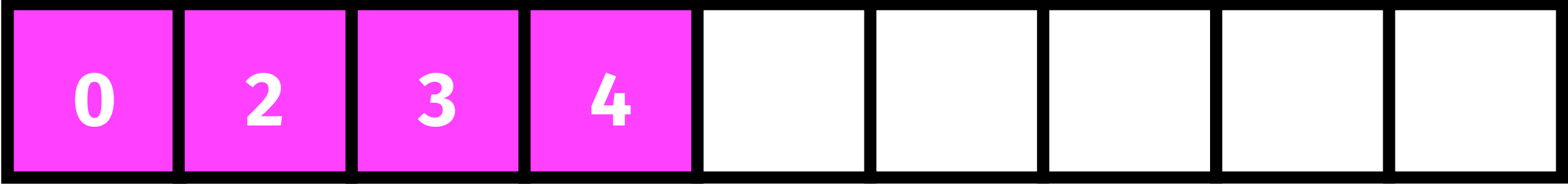


COPYING COLLECTION

FROM Space



TO Space



COPYING COLLECTION

FROM Space

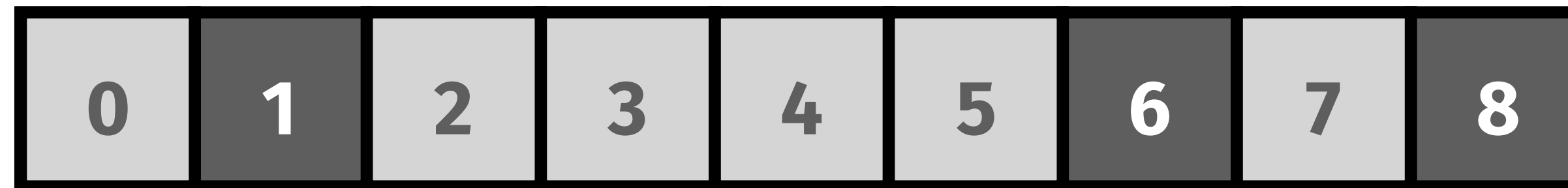


TO Space

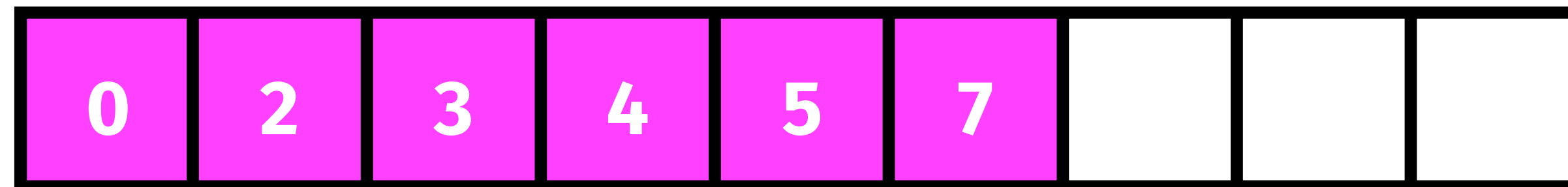


COPYING COLLECTION

FROM Space



TO Space



COPYING COLLECTION

FROM Space

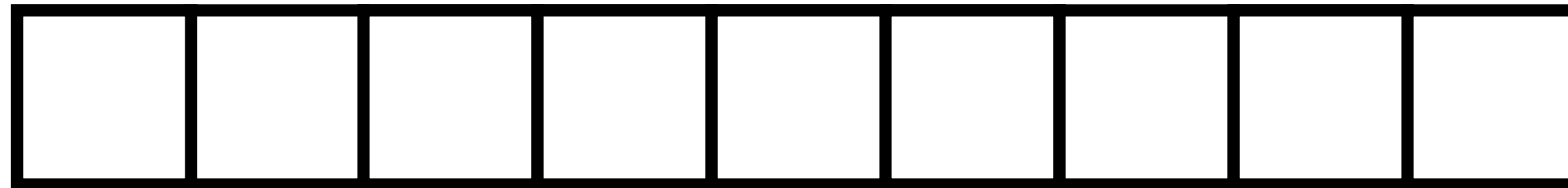


TO Space

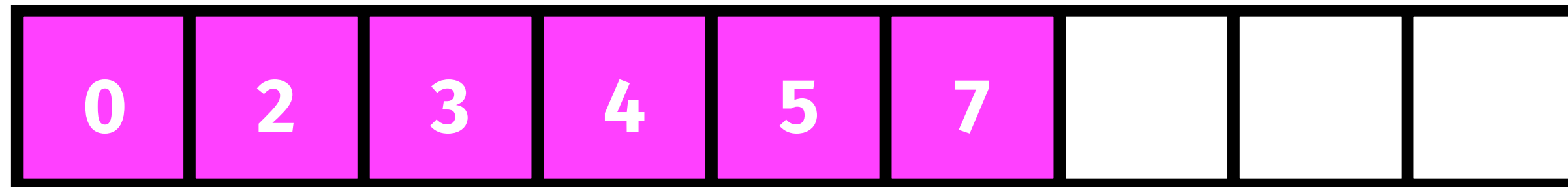


COPYING COLLECTION

FROM Space



TO Space

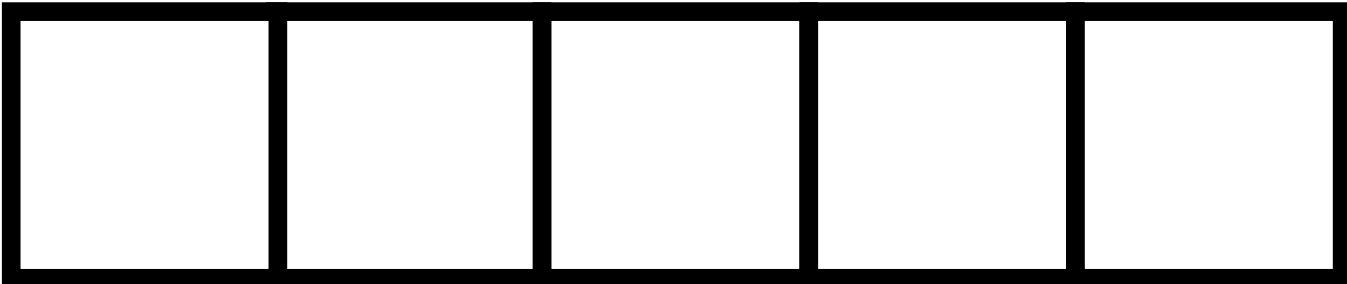


GENERATIONAL COLLECTION

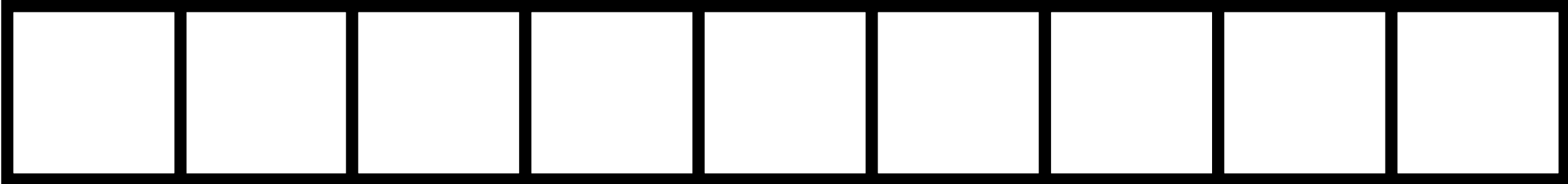
If you've been using it for a while, it's probably important

GENERATIONAL COLLECTION

New objects

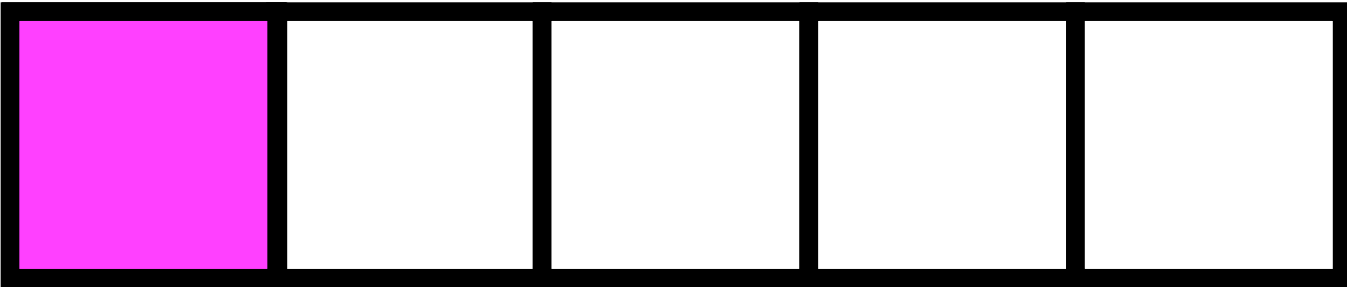


Old objects

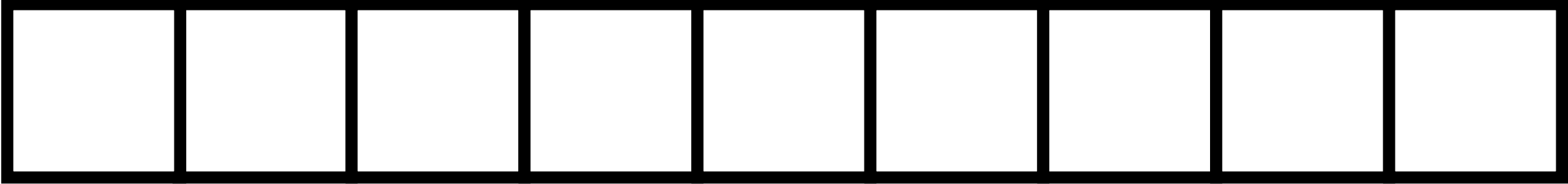


GENERATIONAL COLLECTION

New objects

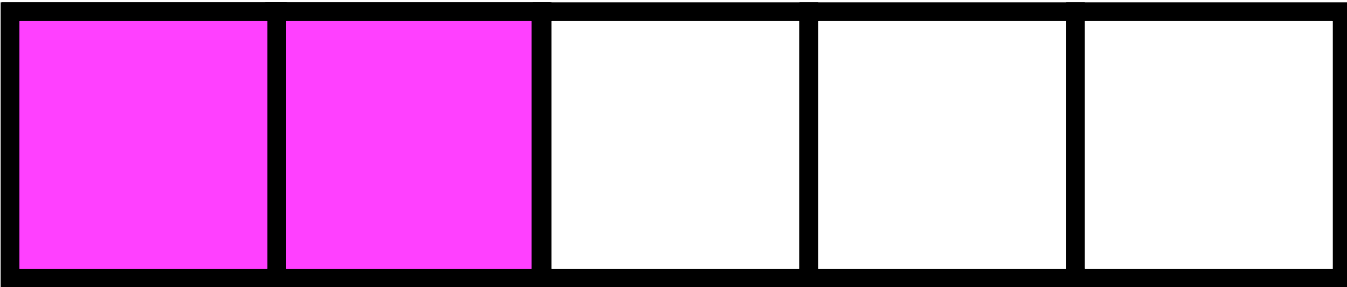


Old objects

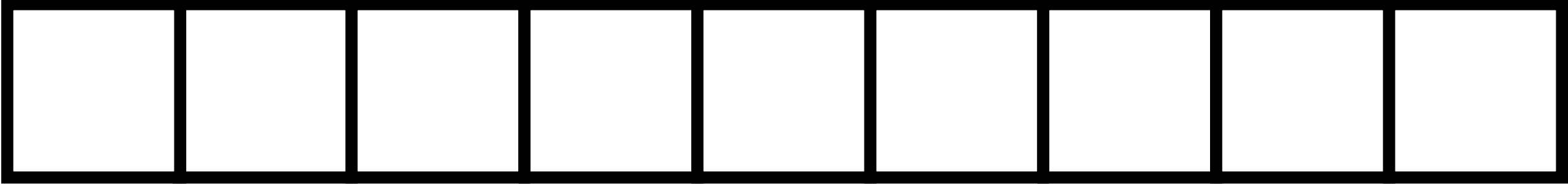


GENERATIONAL COLLECTION

New objects

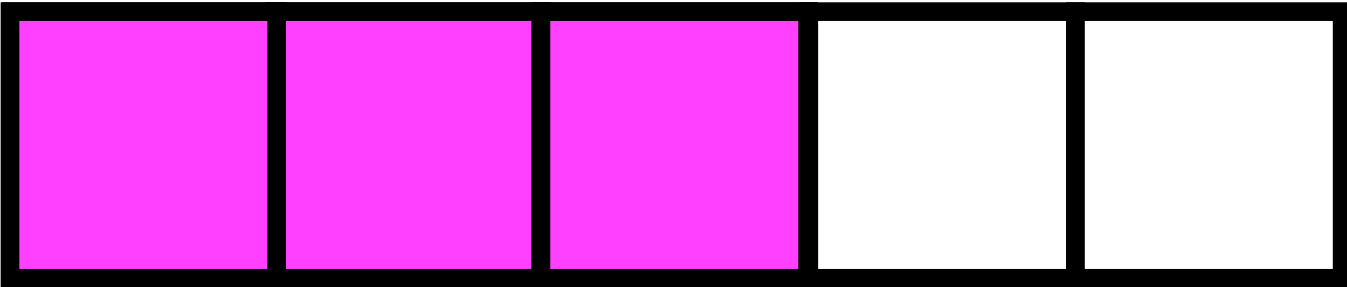


Old objects

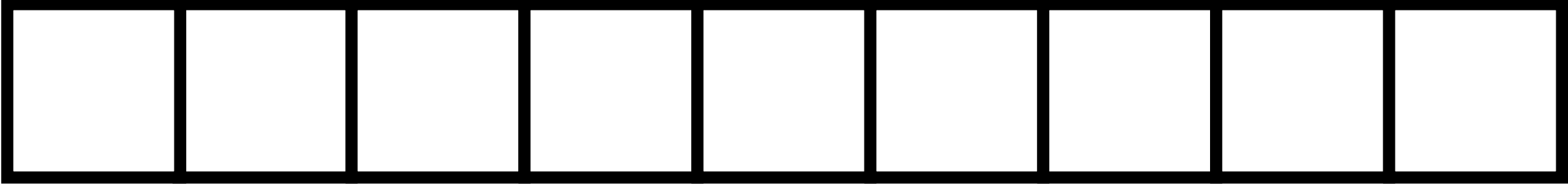


GENERATIONAL COLLECTION

New objects

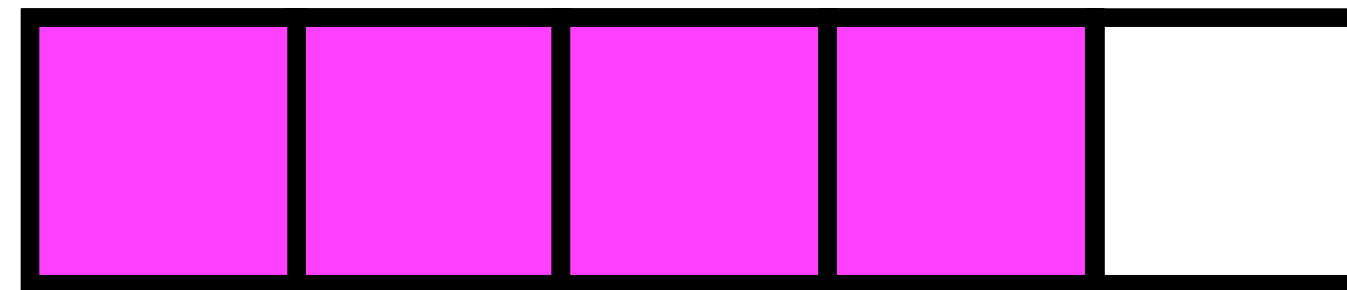


Old objects

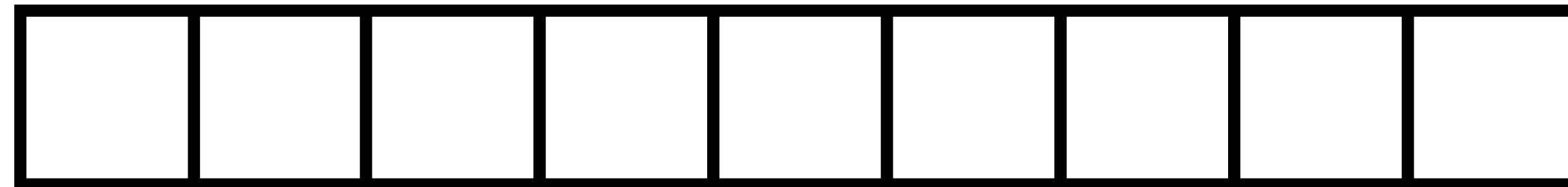


GENERATIONAL COLLECTION

New objects

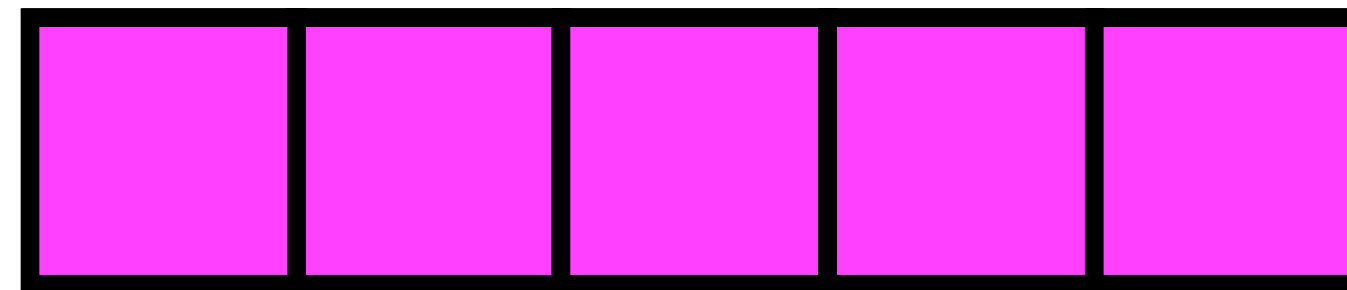


Old objects

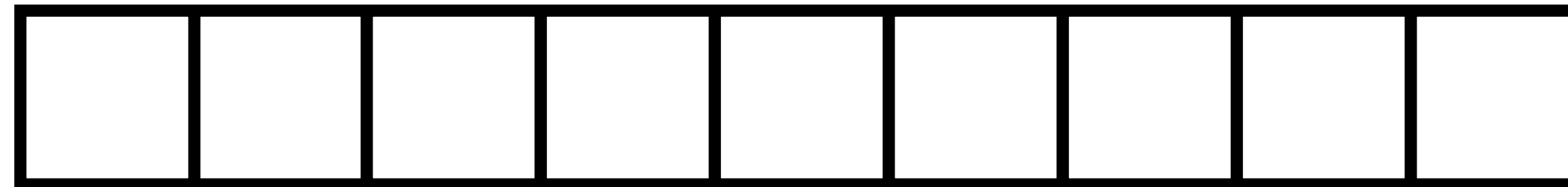


GENERATIONAL COLLECTION

New objects

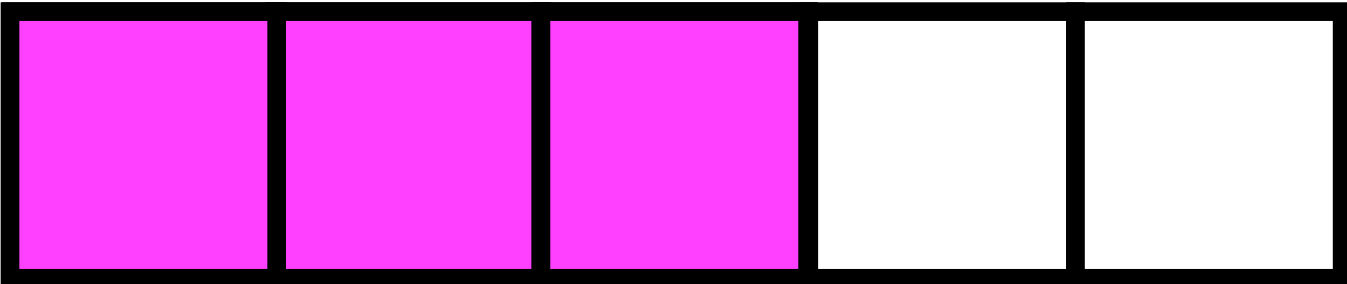


Old objects

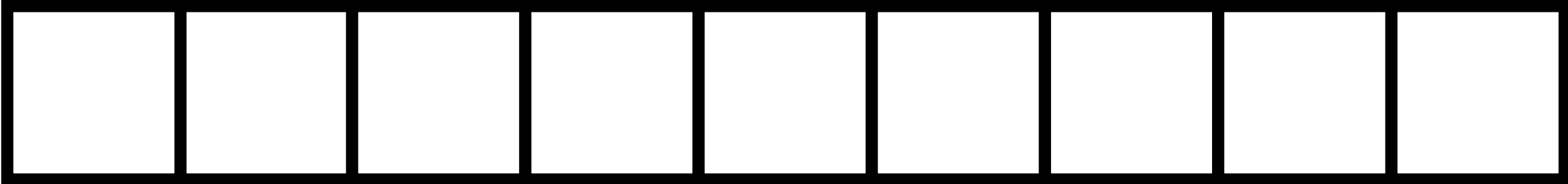


GENERATIONAL COLLECTION

New objects

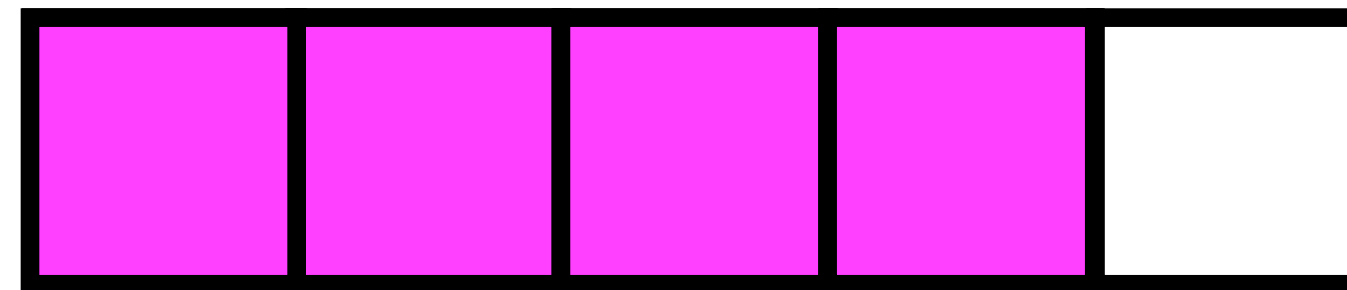


Old objects

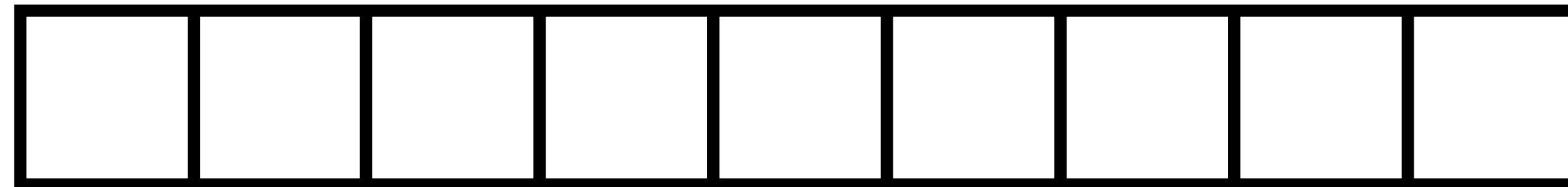


GENERATIONAL COLLECTION

New objects

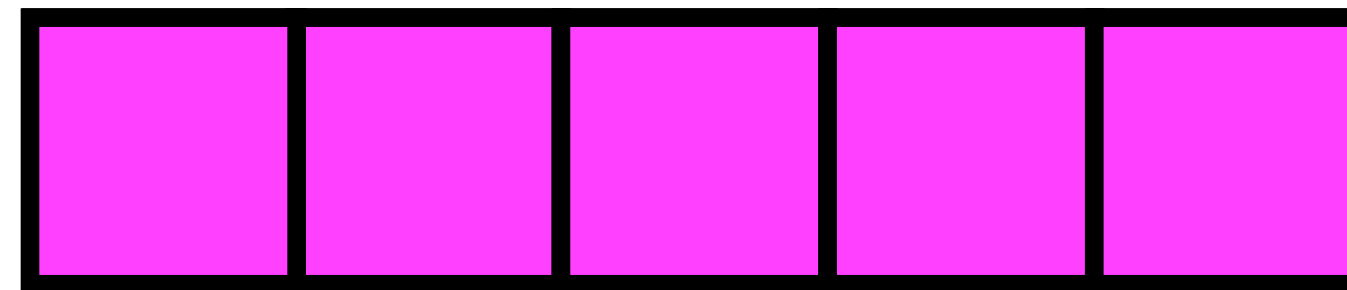


Old objects

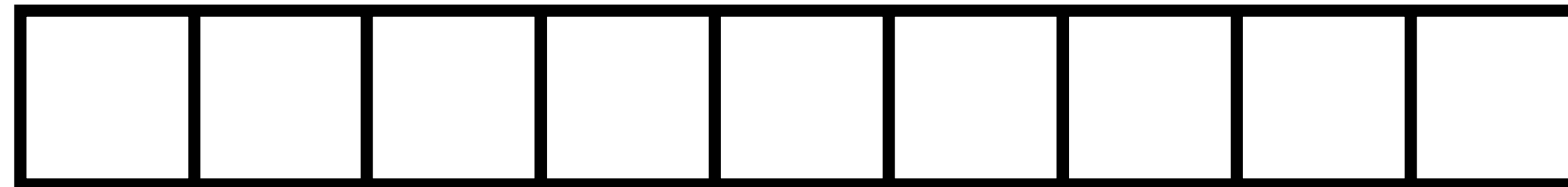


GENERATIONAL COLLECTION

New objects

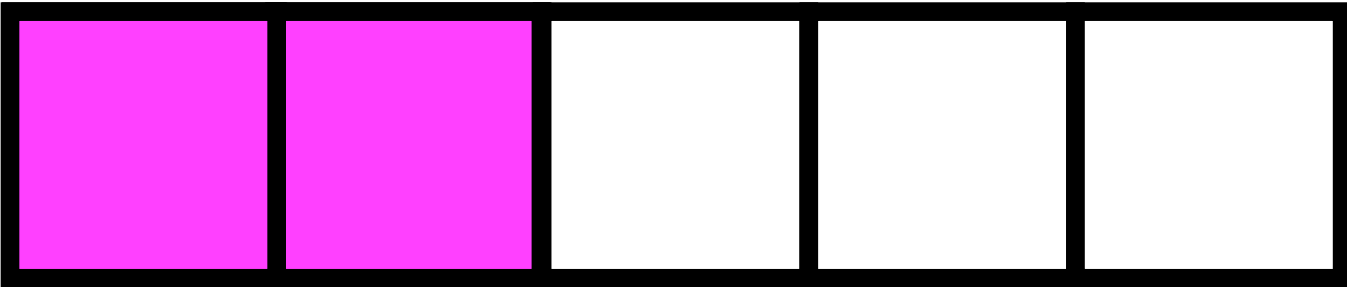


Old objects

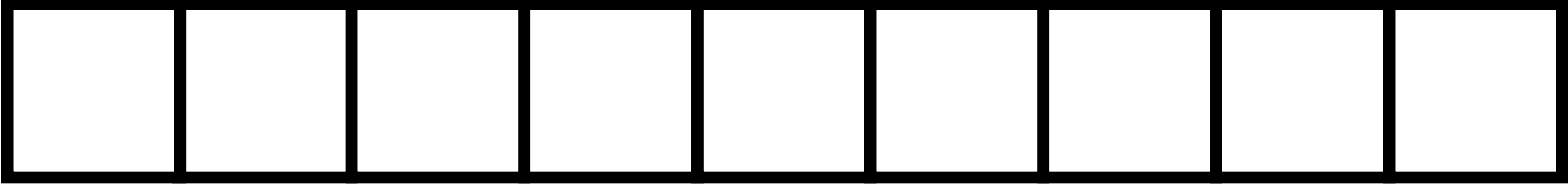


GENERATIONAL COLLECTION

New objects

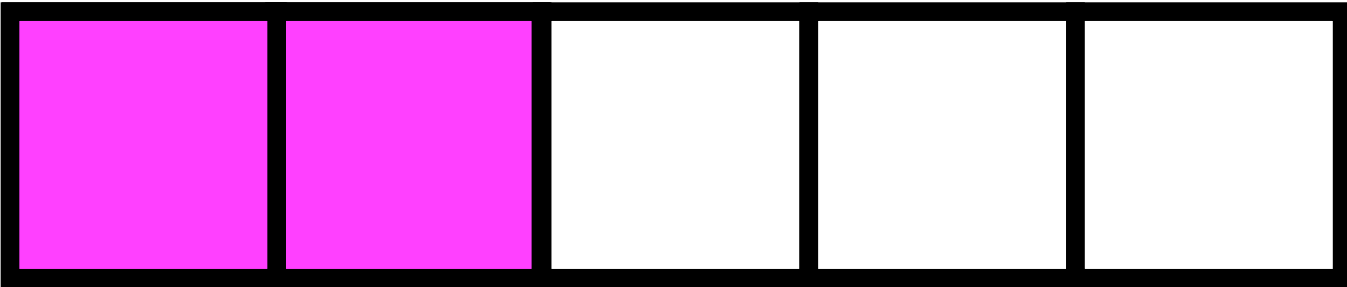


Old objects

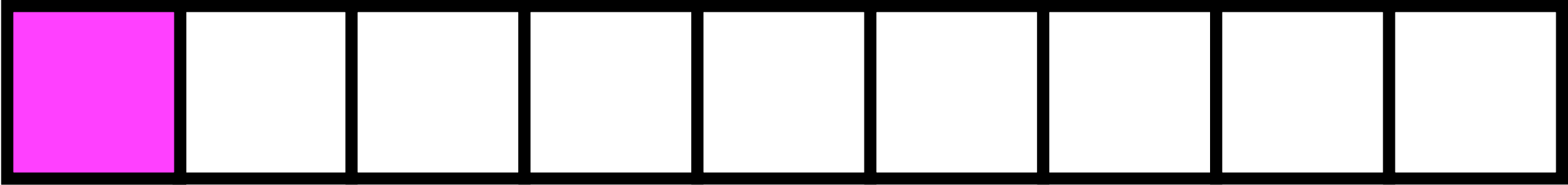


GENERATIONAL COLLECTION

New objects

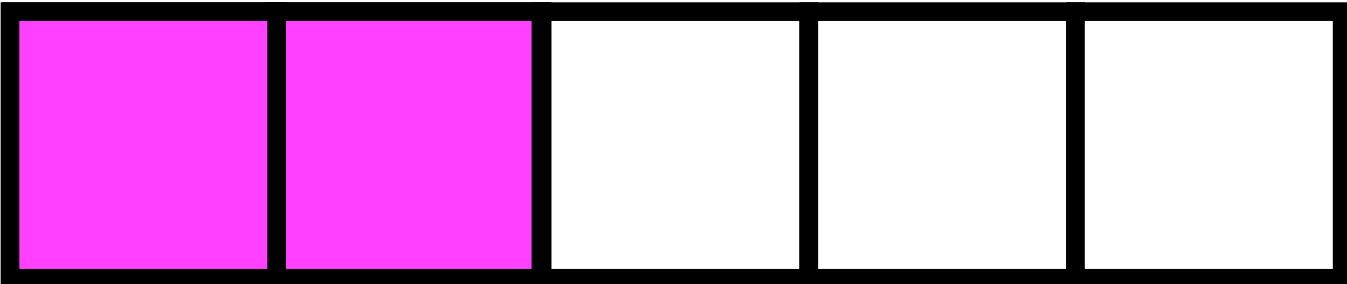


Old objects

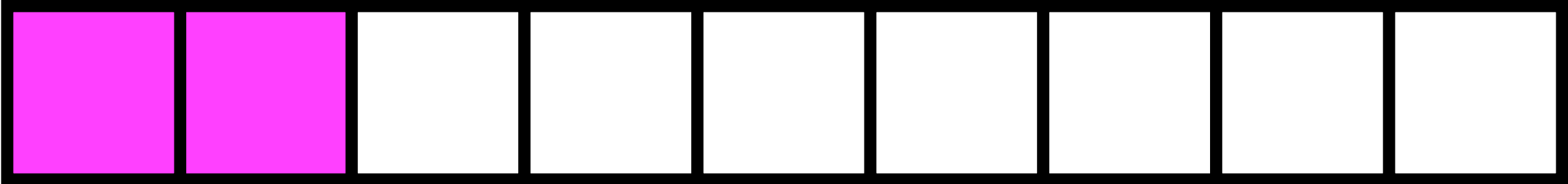


GENERATIONAL COLLECTION

New objects

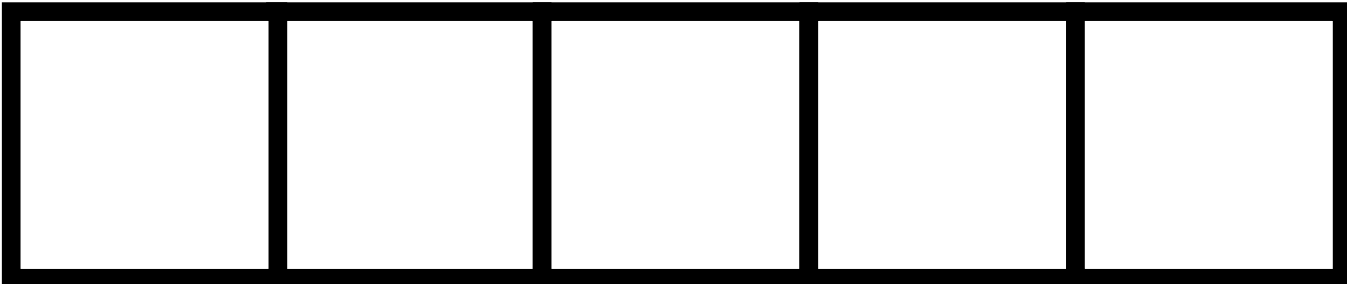


Old objects

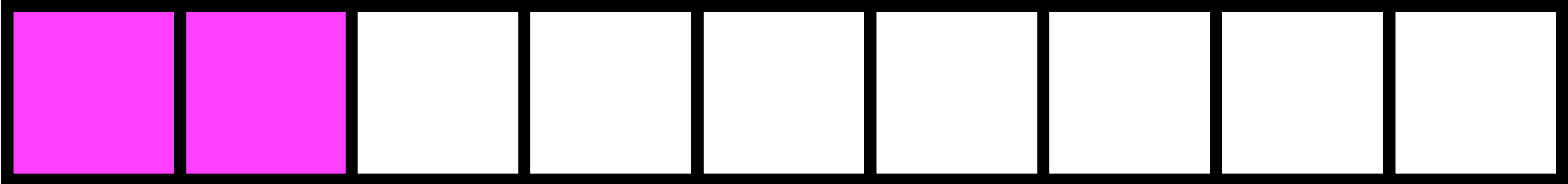


GENERATIONAL COLLECTION

New objects



Old objects



WHAT ABOUT THE BEAM?

Erlang & Elixir garbage collection



Processes

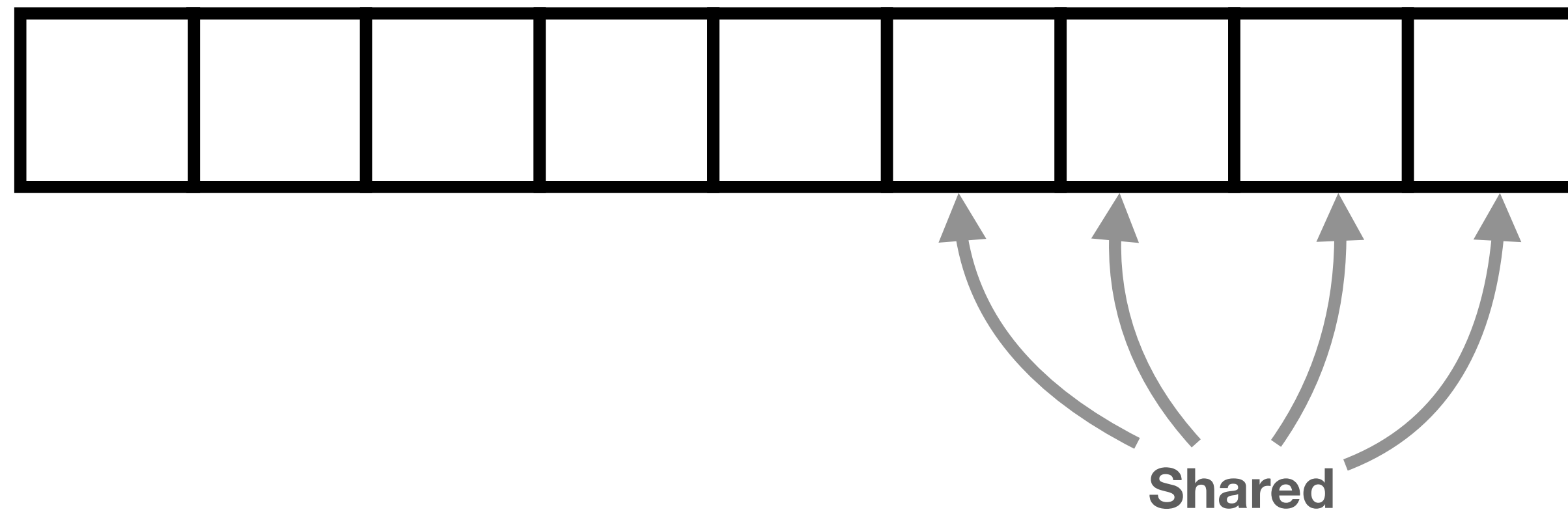


Processes

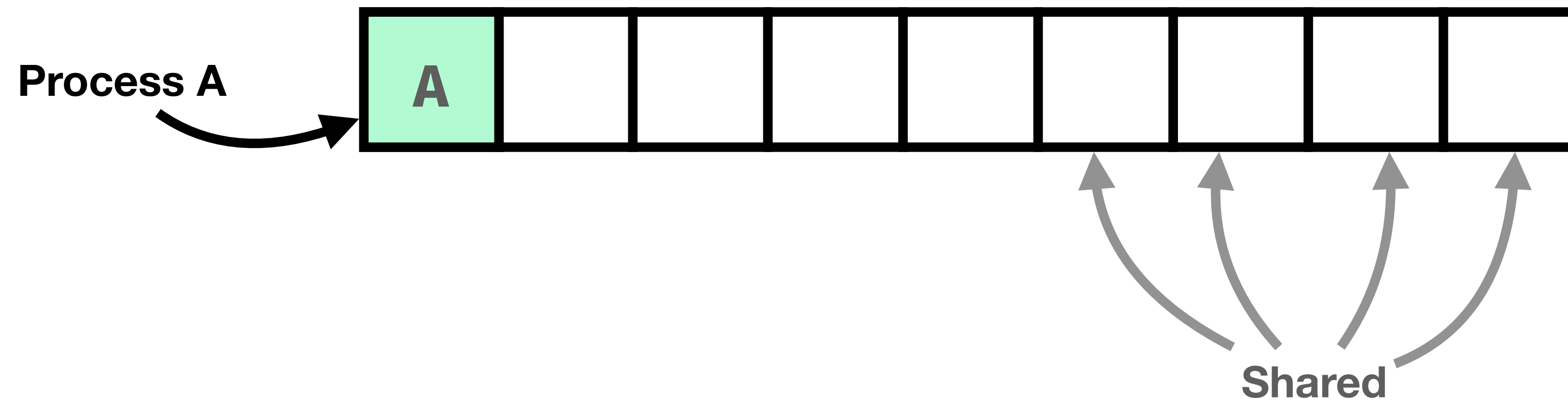


Immutability

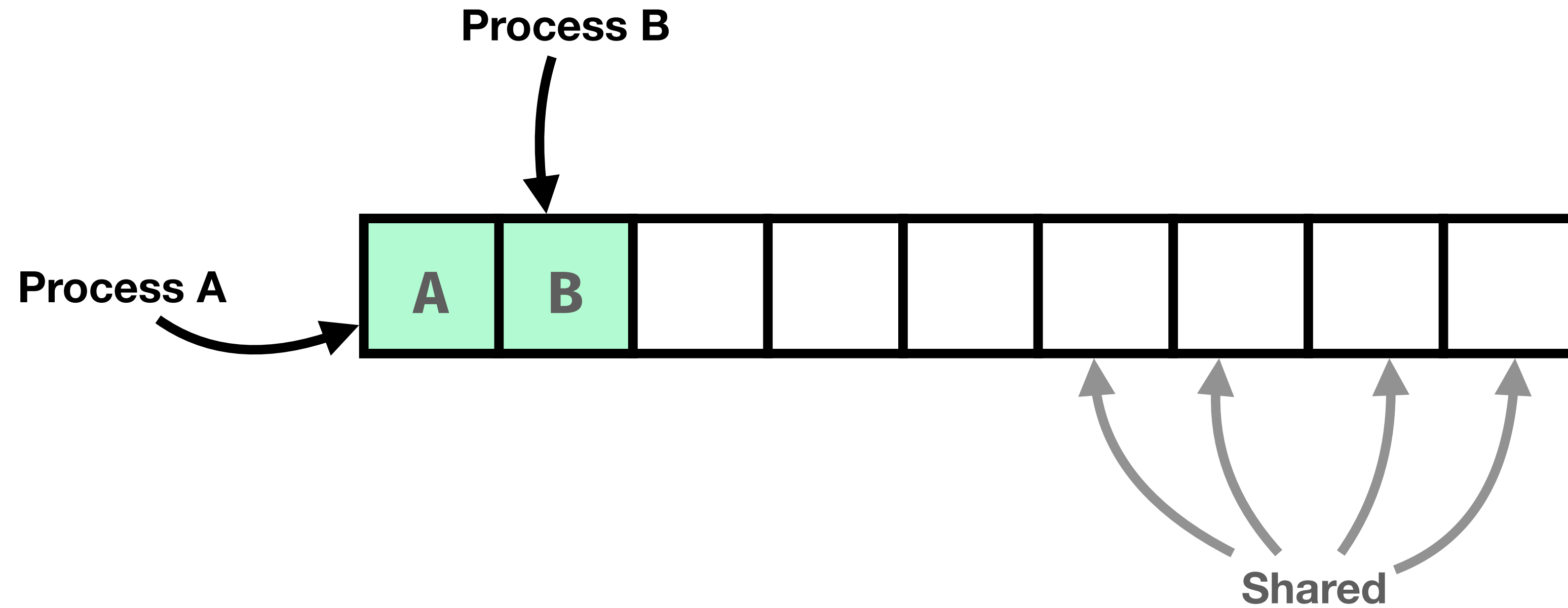
MEMORY LAYOUT



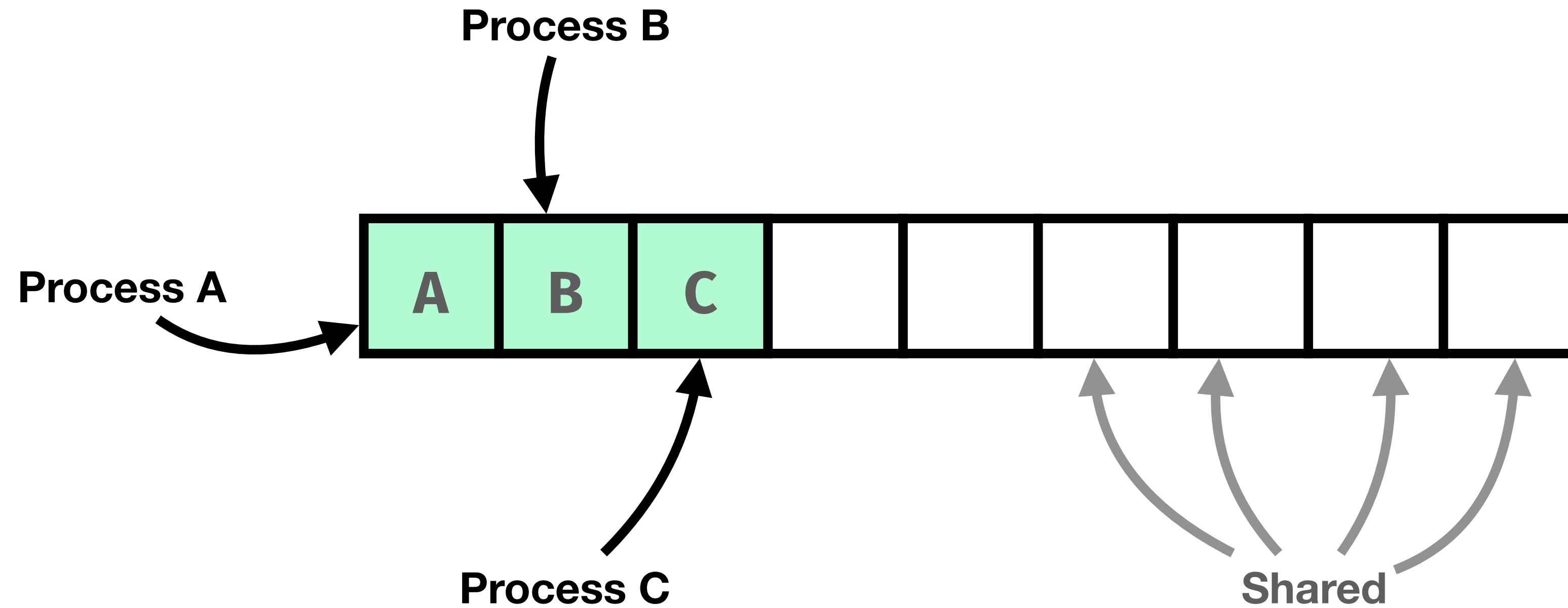
MEMORY LAYOUT



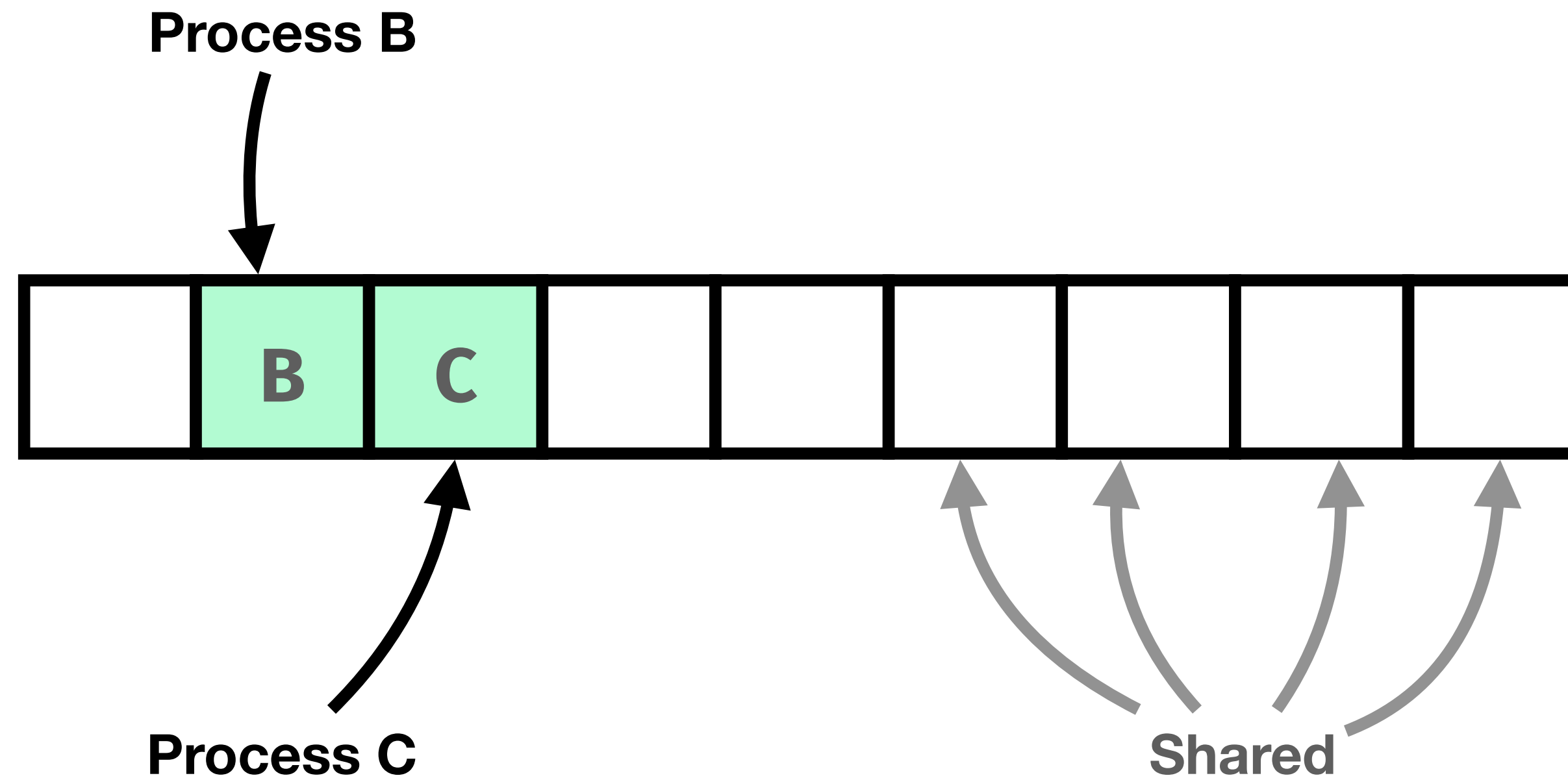
MEMORY LAYOUT



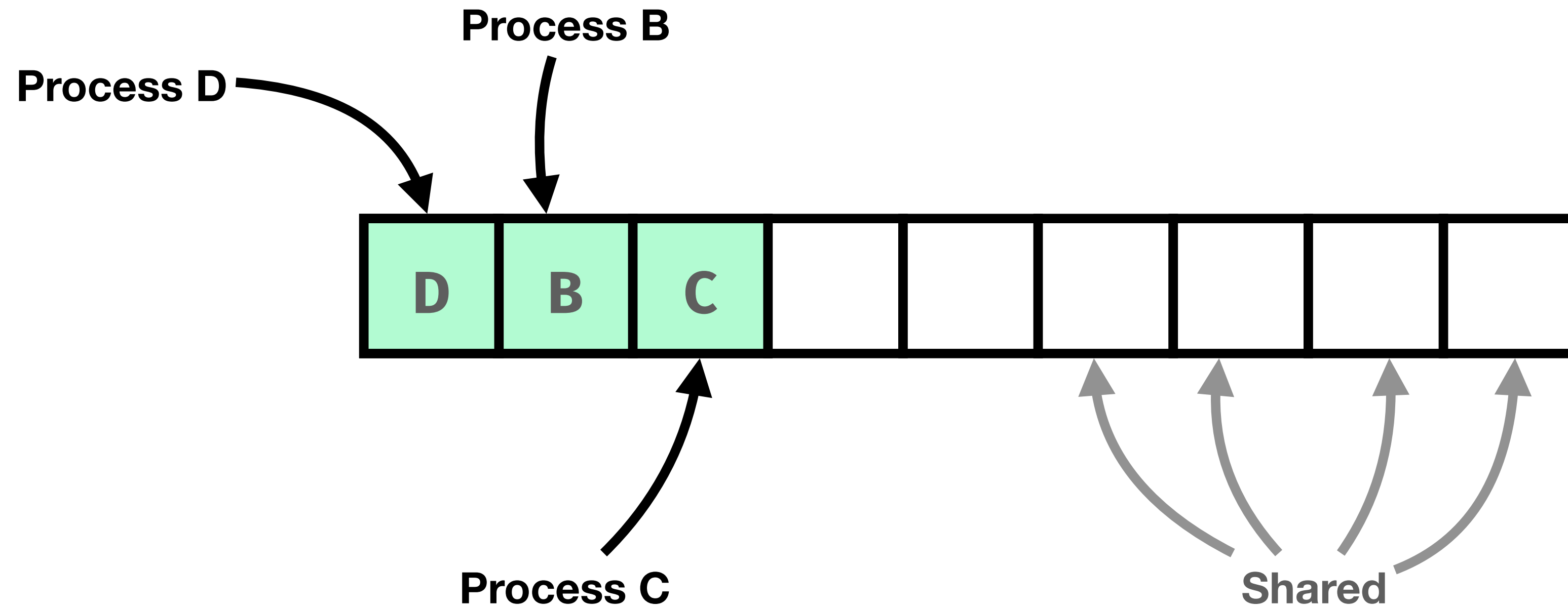
MEMORY LAYOUT



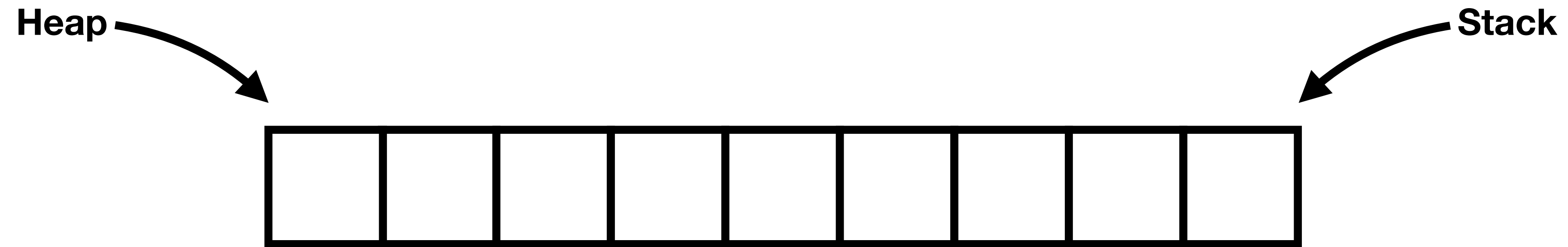
MEMORY LAYOUT



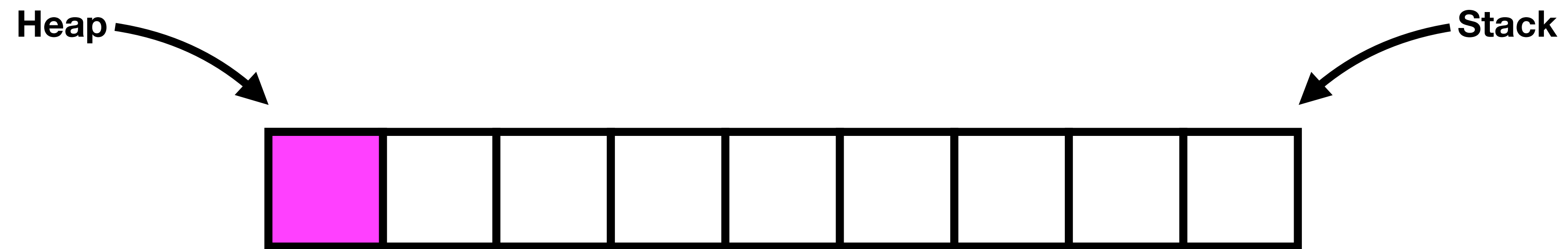
MEMORY LAYOUT



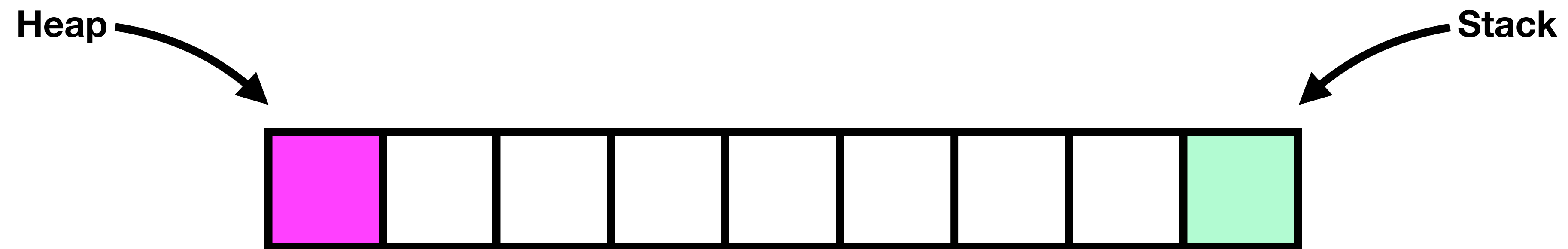
PROCESS MEMORY



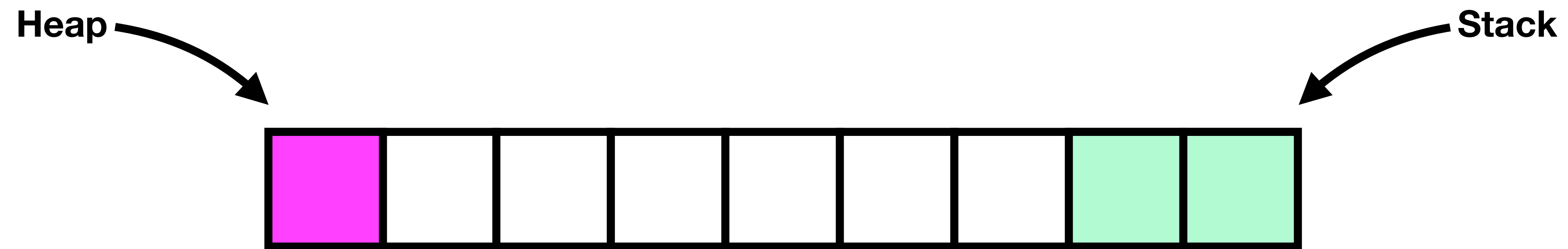
PROCESS MEMORY



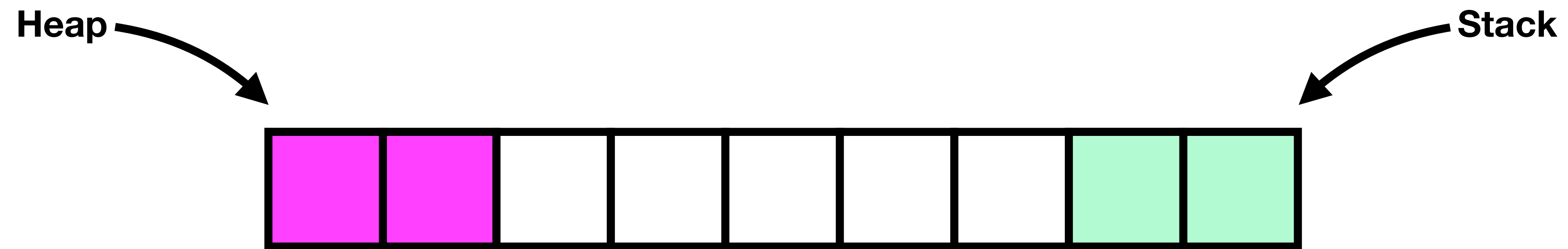
PROCESS MEMORY



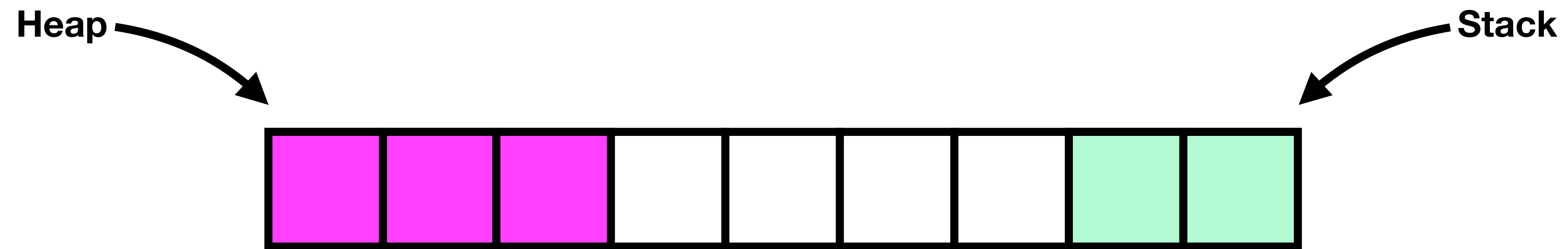
PROCESS MEMORY



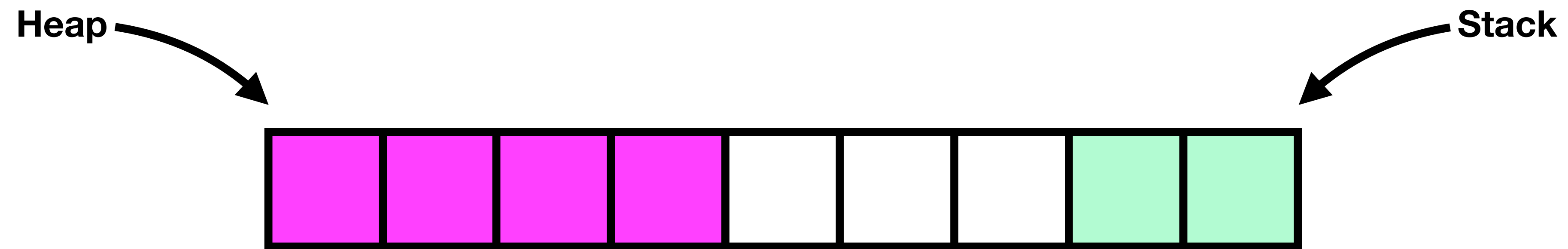
PROCESS MEMORY



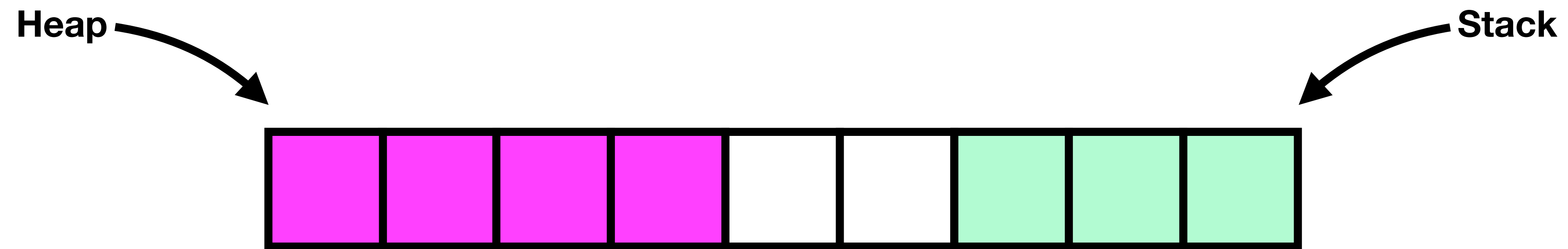
PROCESS MEMORY



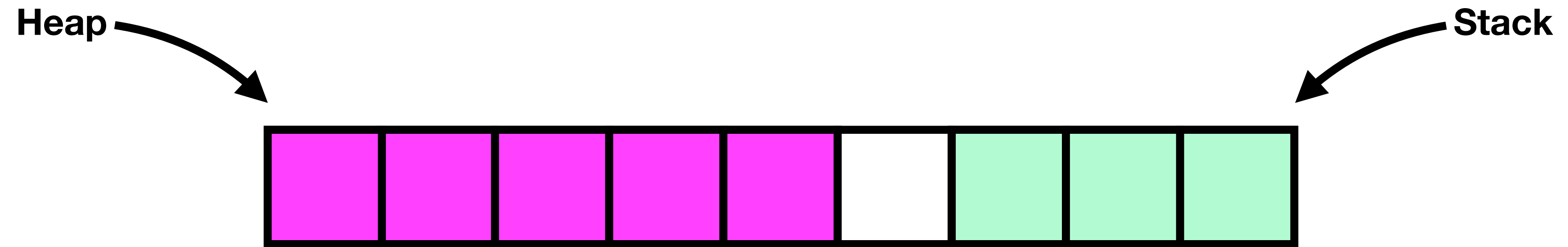
PROCESS MEMORY



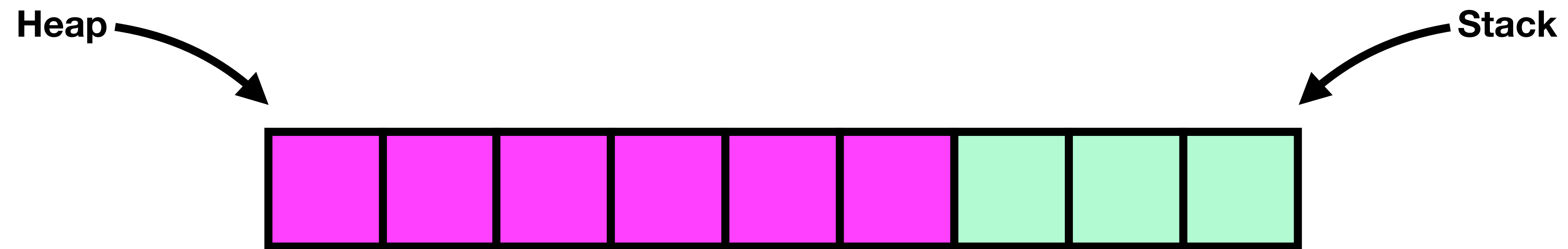
PROCESS MEMORY



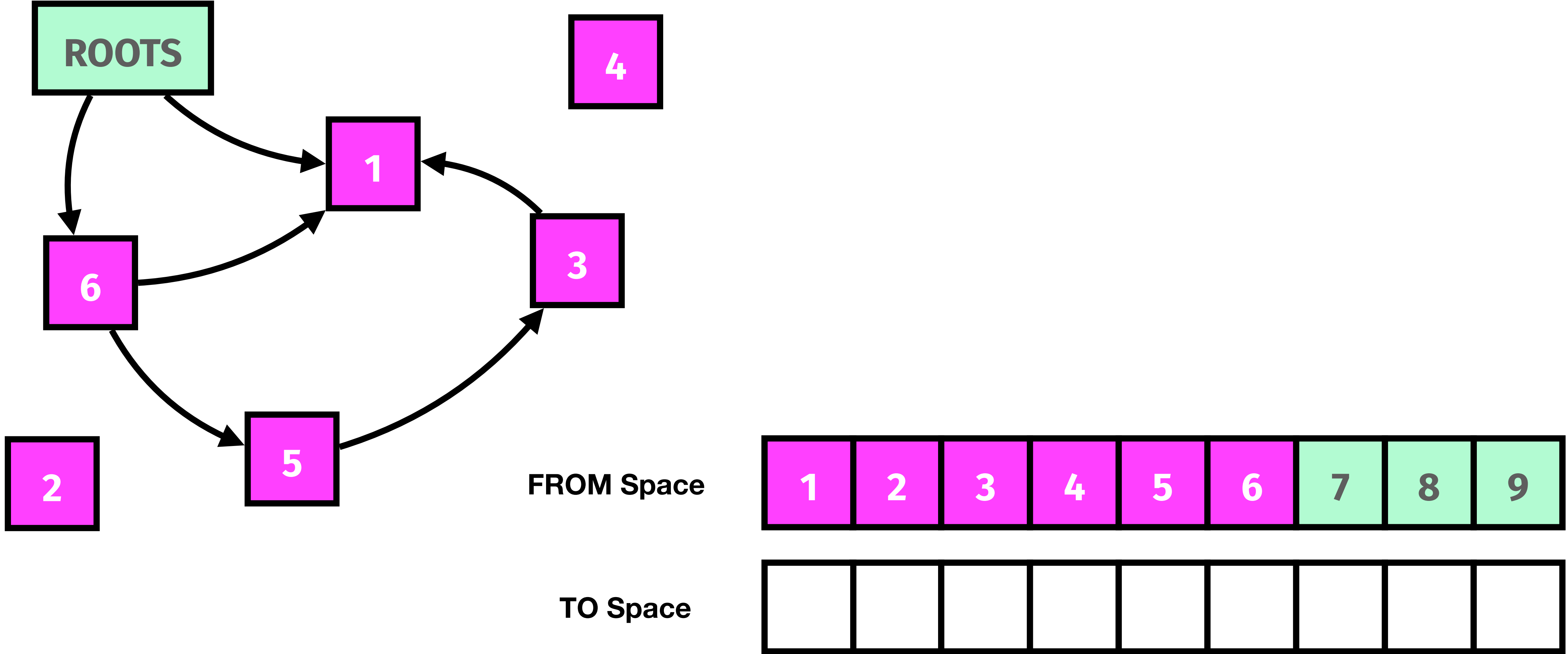
PROCESS MEMORY



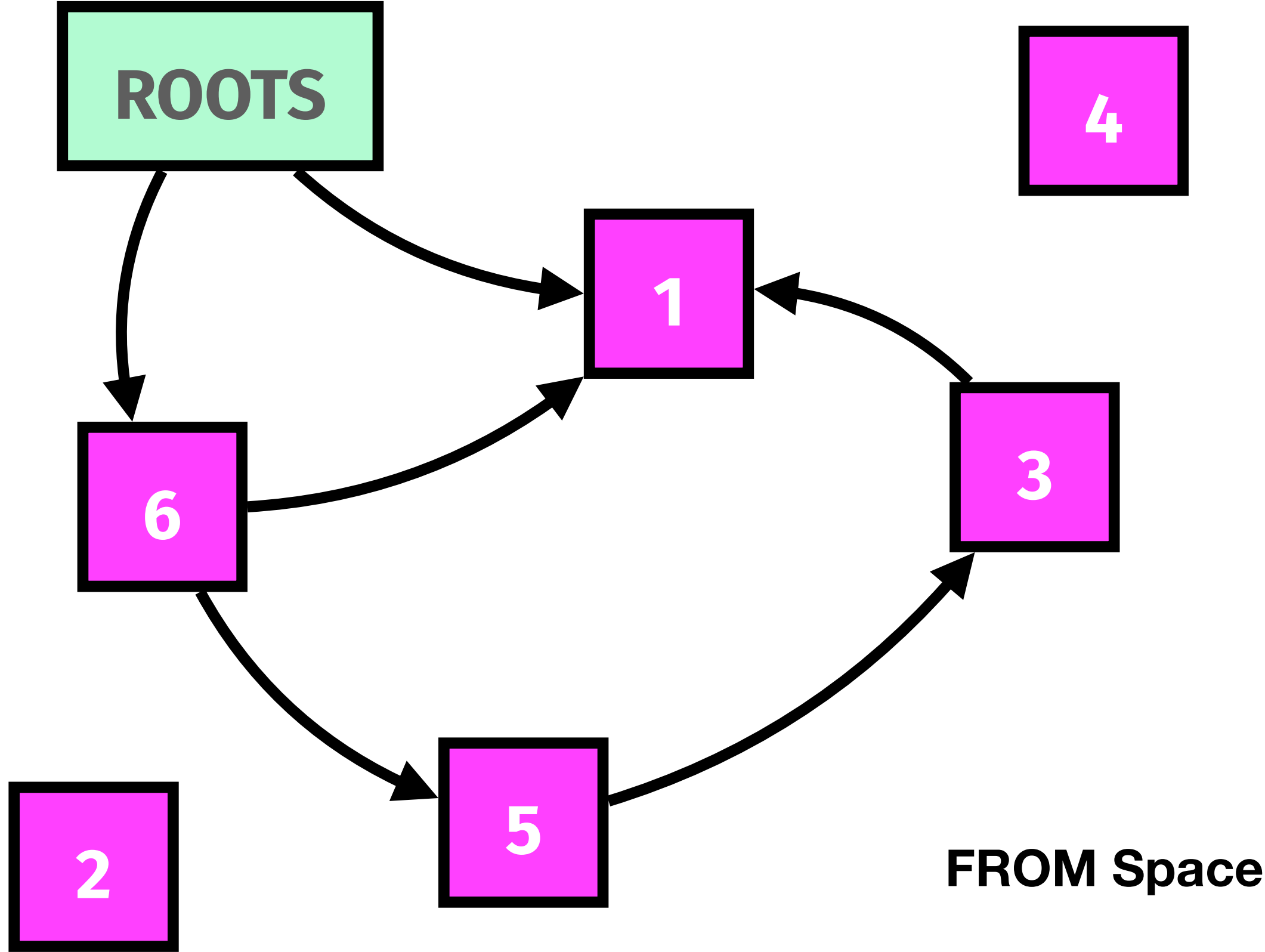
PROCESS MEMORY



COPYING COLLECTION



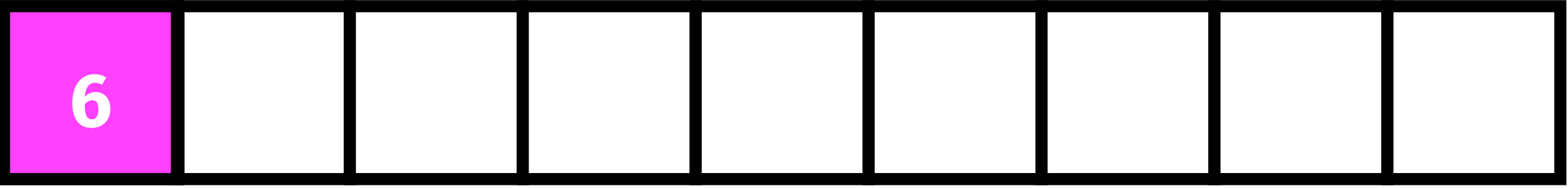
COPYING COLLECTION



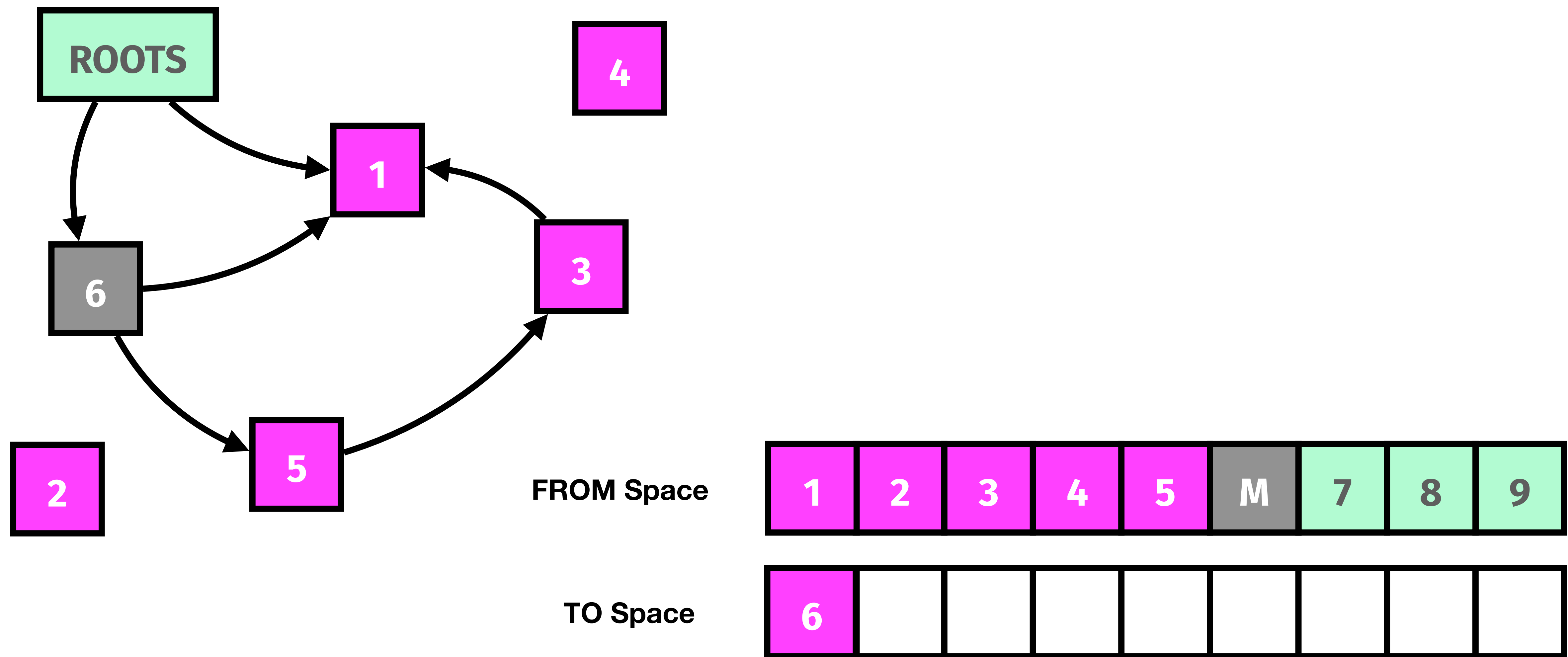
FROM Space



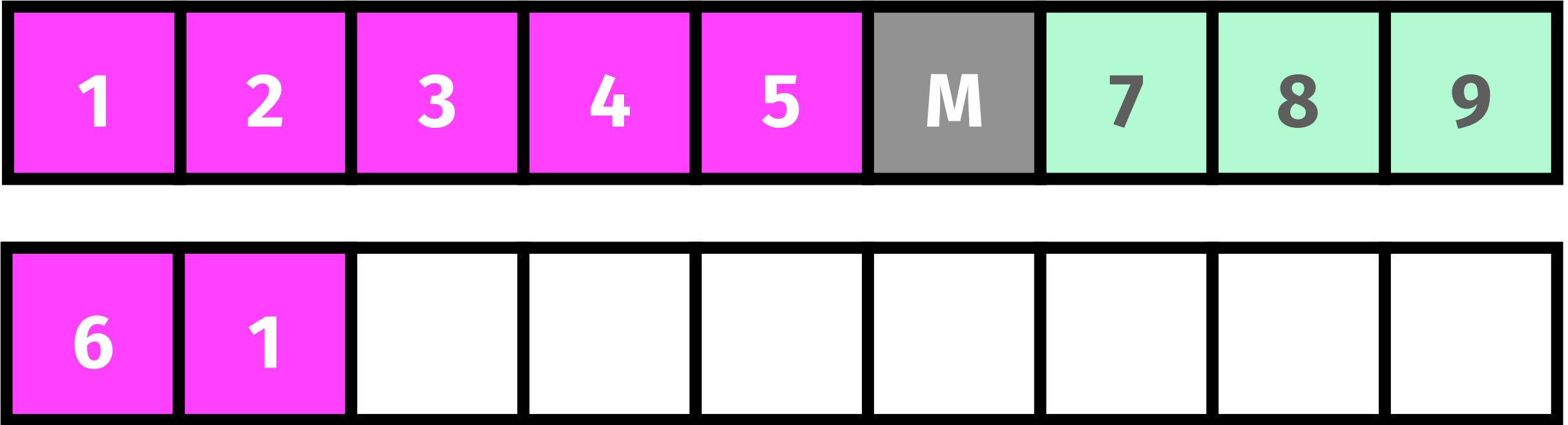
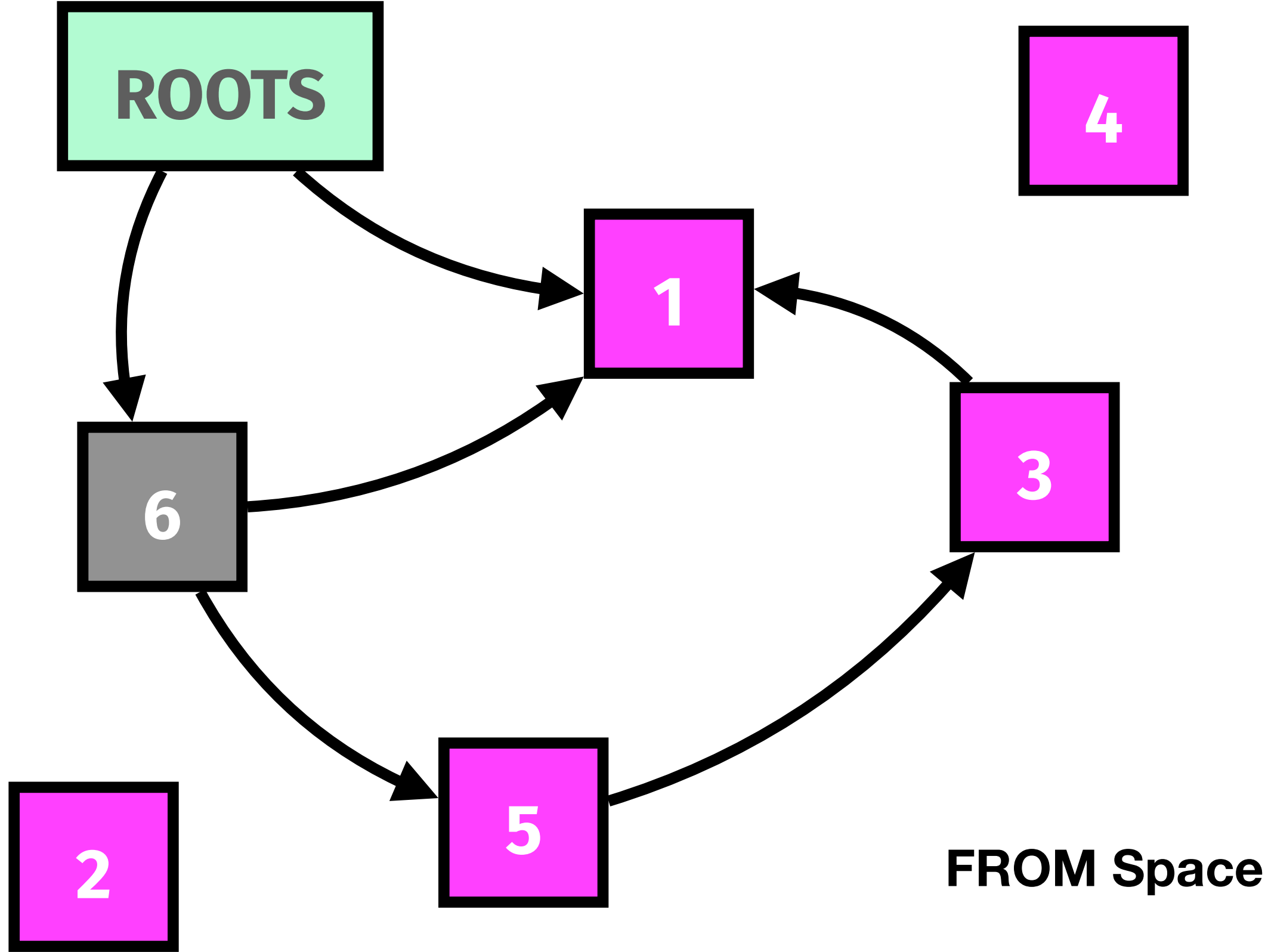
TO Space



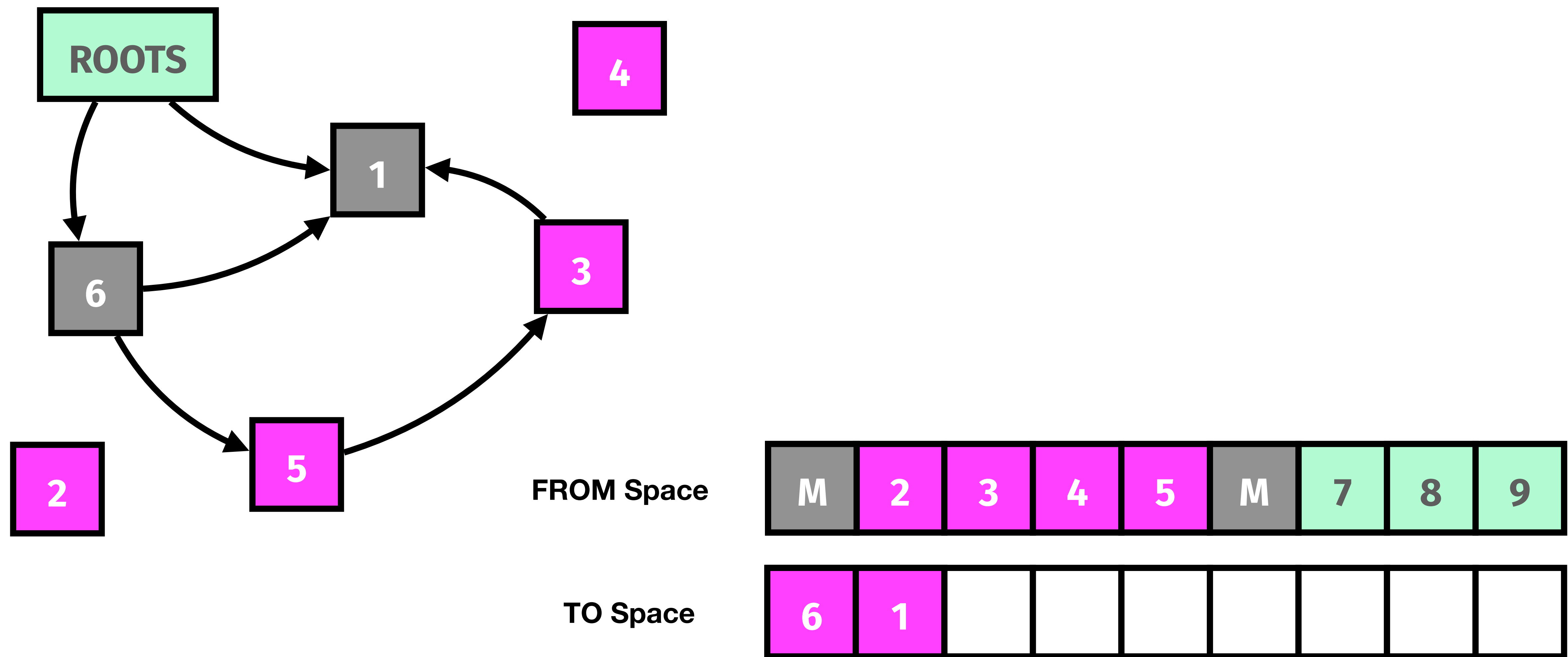
COPYING COLLECTION



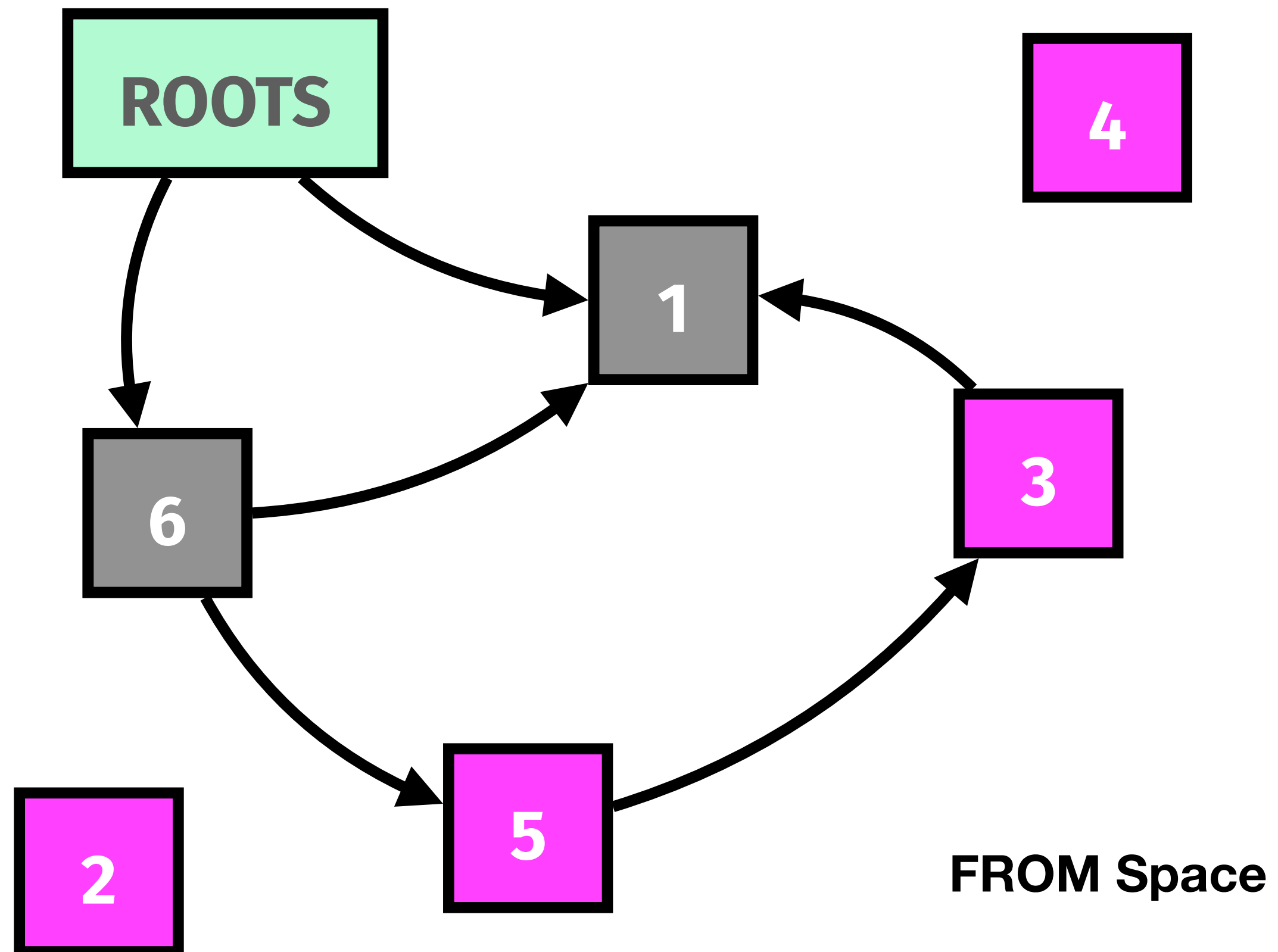
COPYING COLLECTION



COPYING COLLECTION



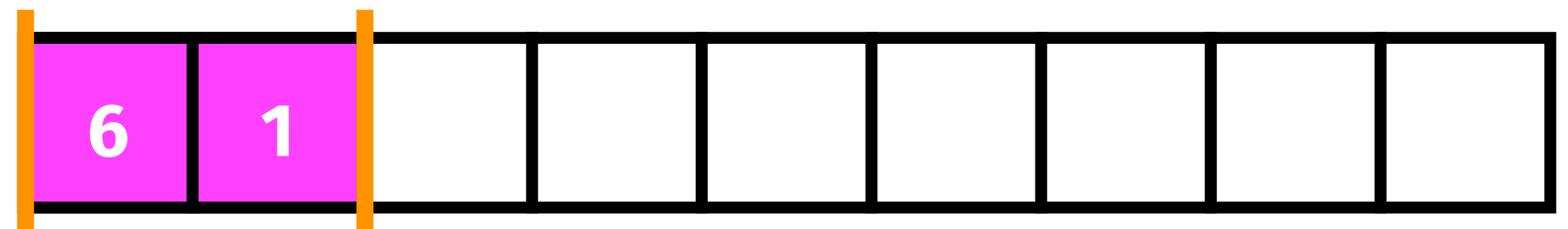
COPYING COLLECTION



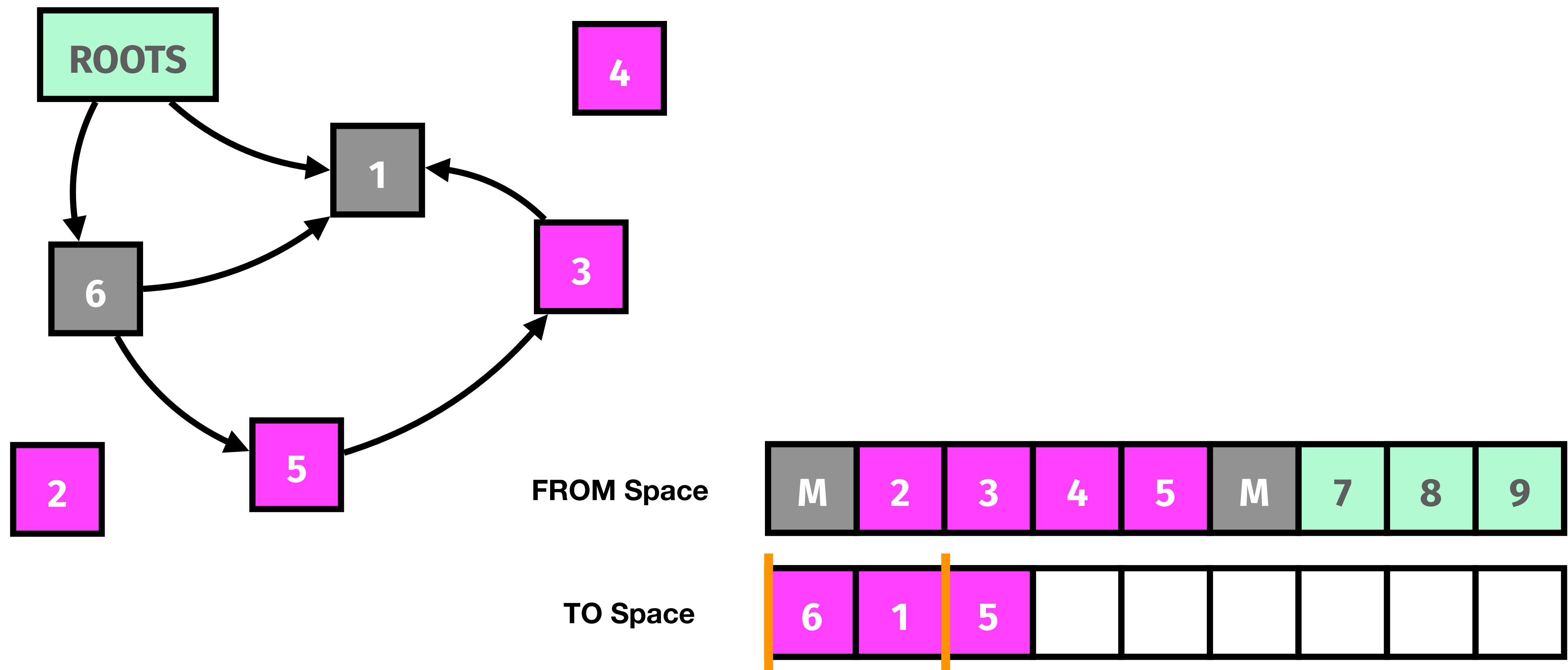
FROM Space



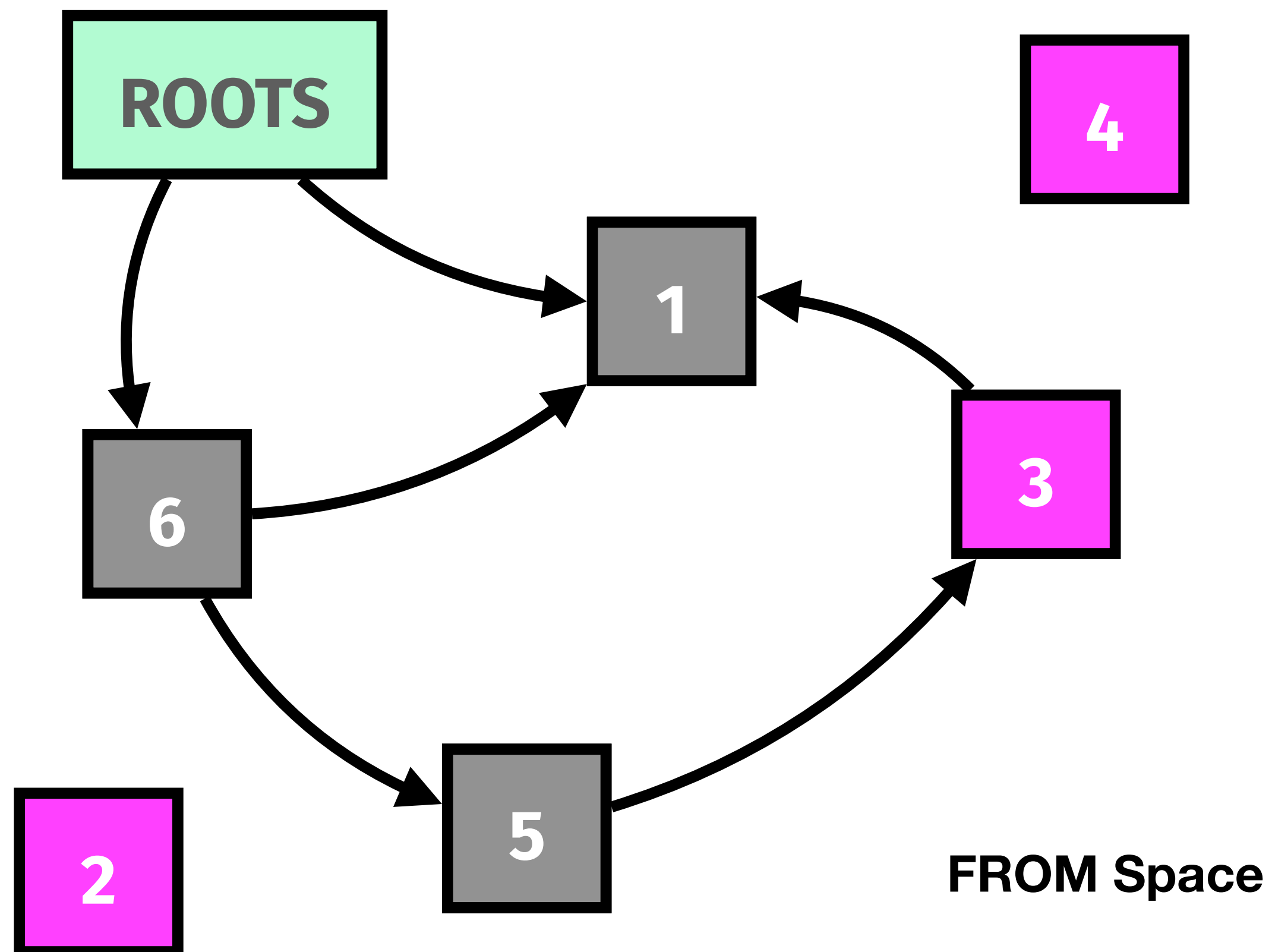
TO Space



COPYING COLLECTION

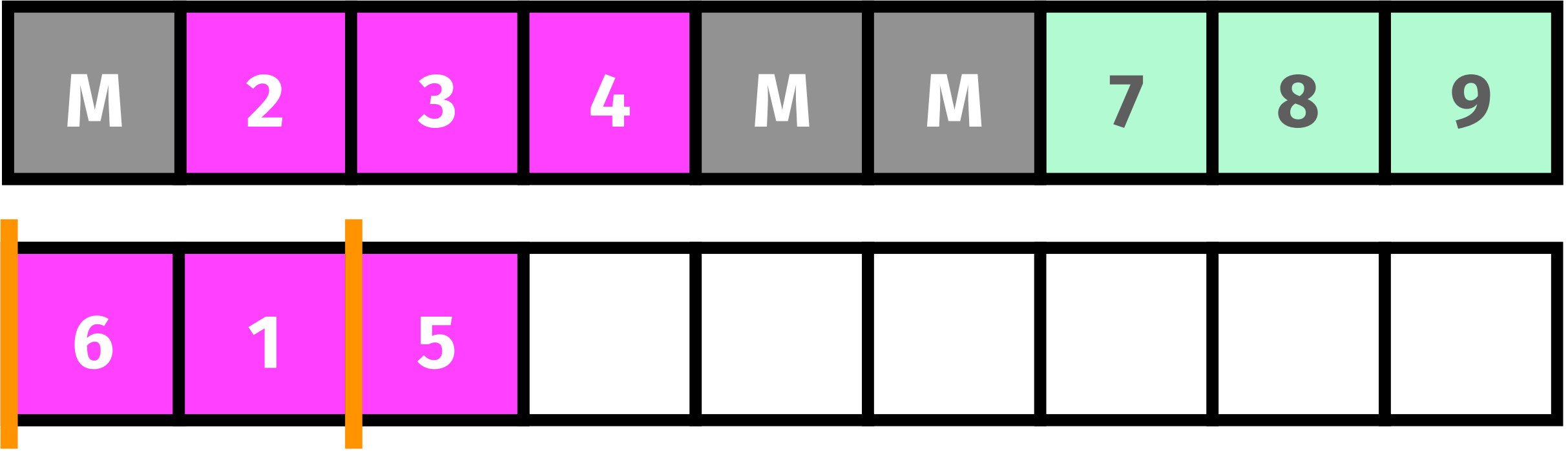


COPYING COLLECTION

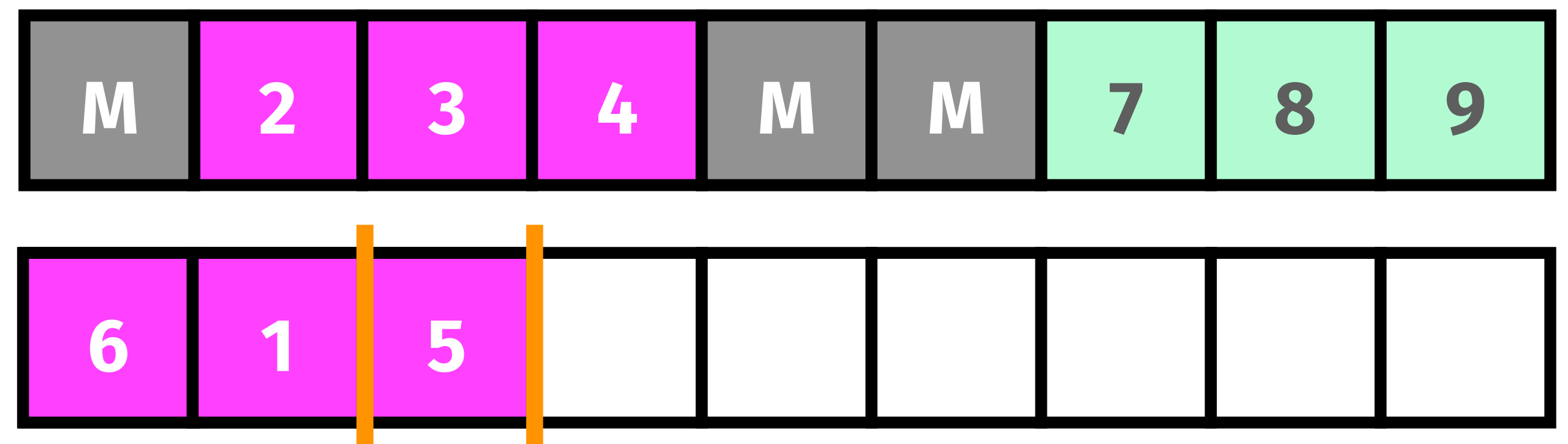
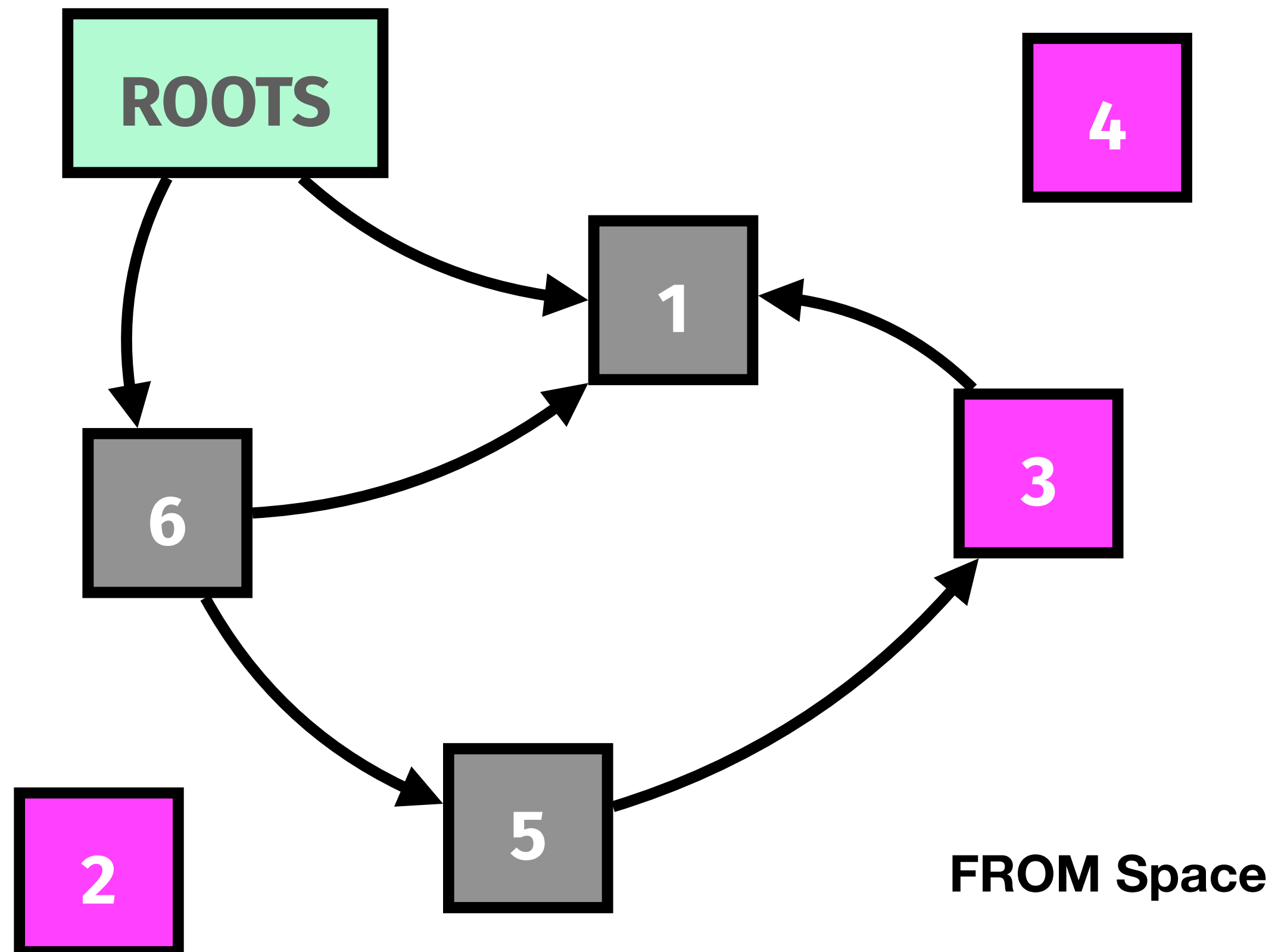


FROM Space

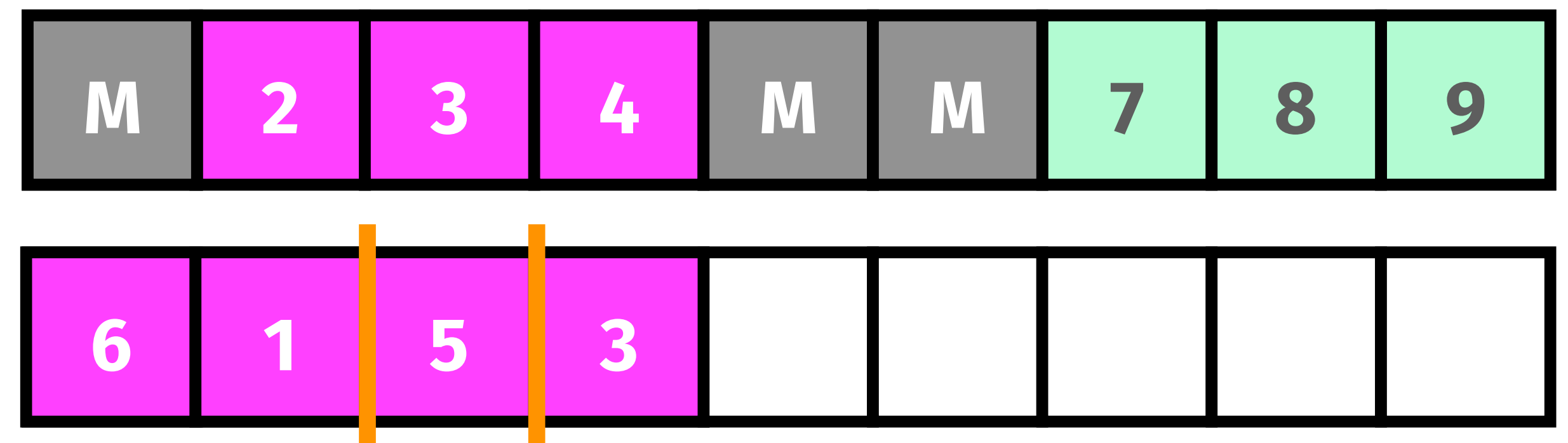
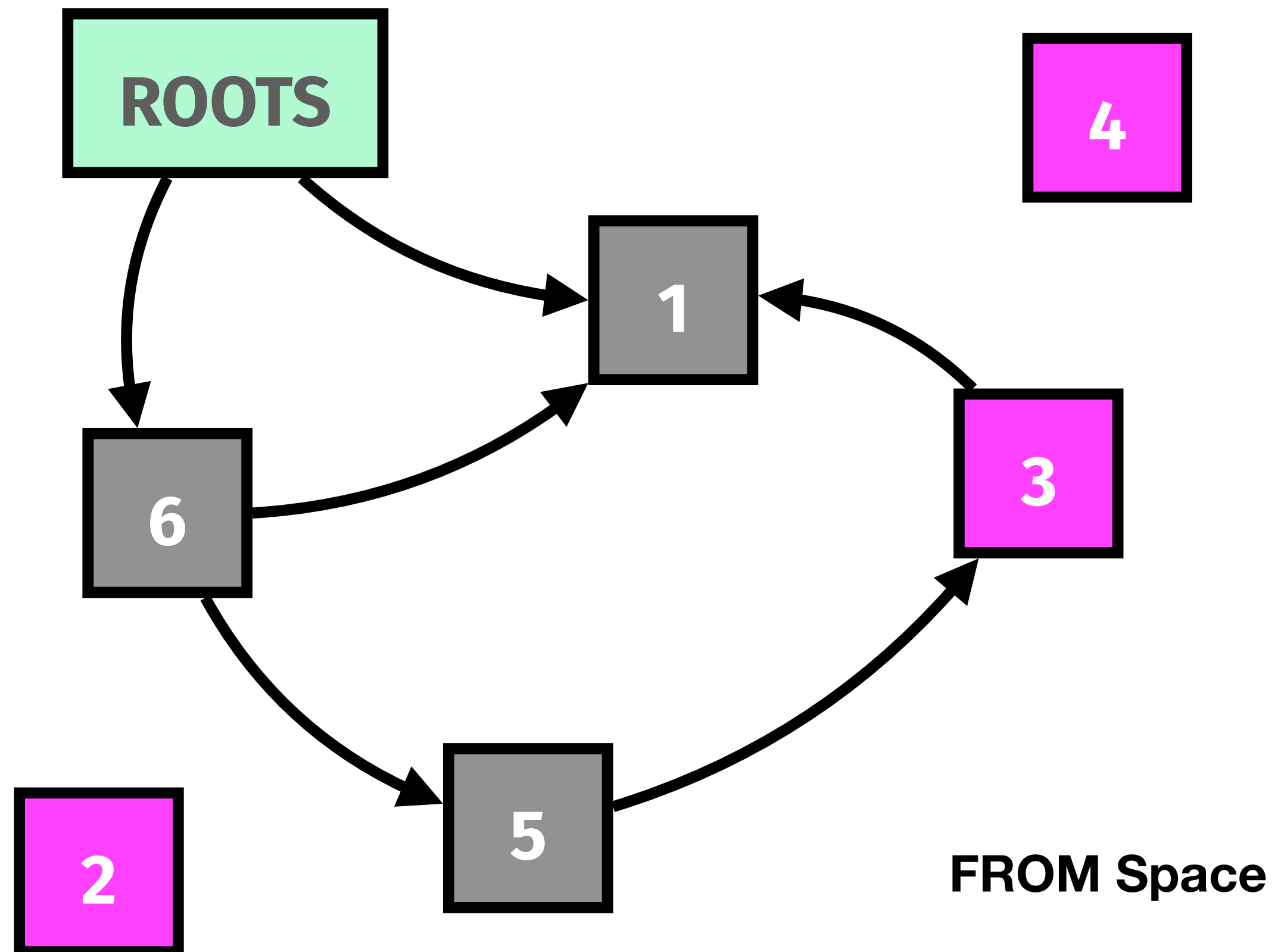
TO Space



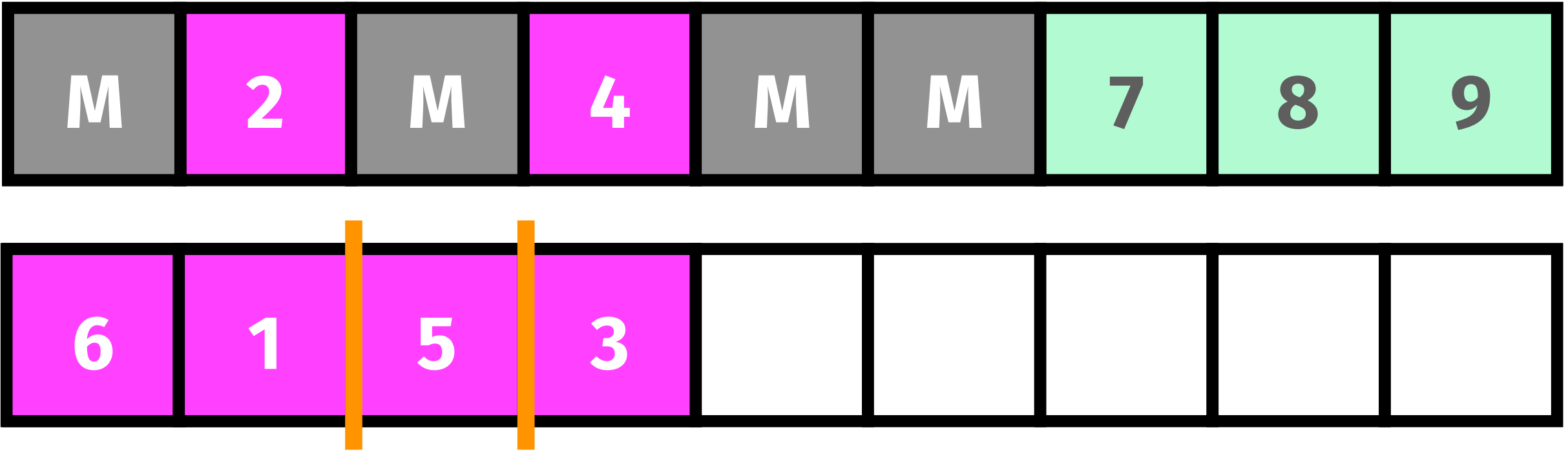
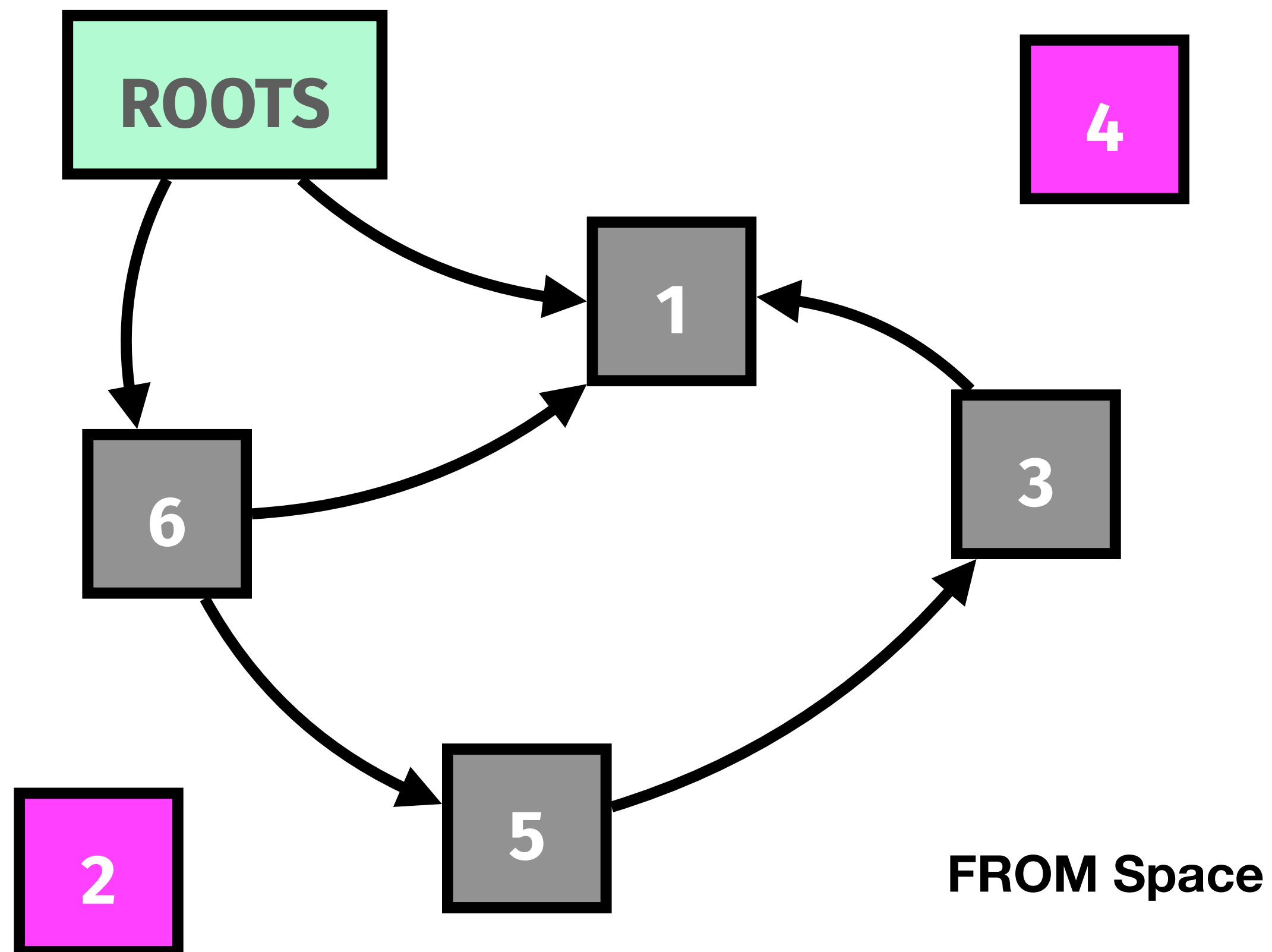
COPYING COLLECTION



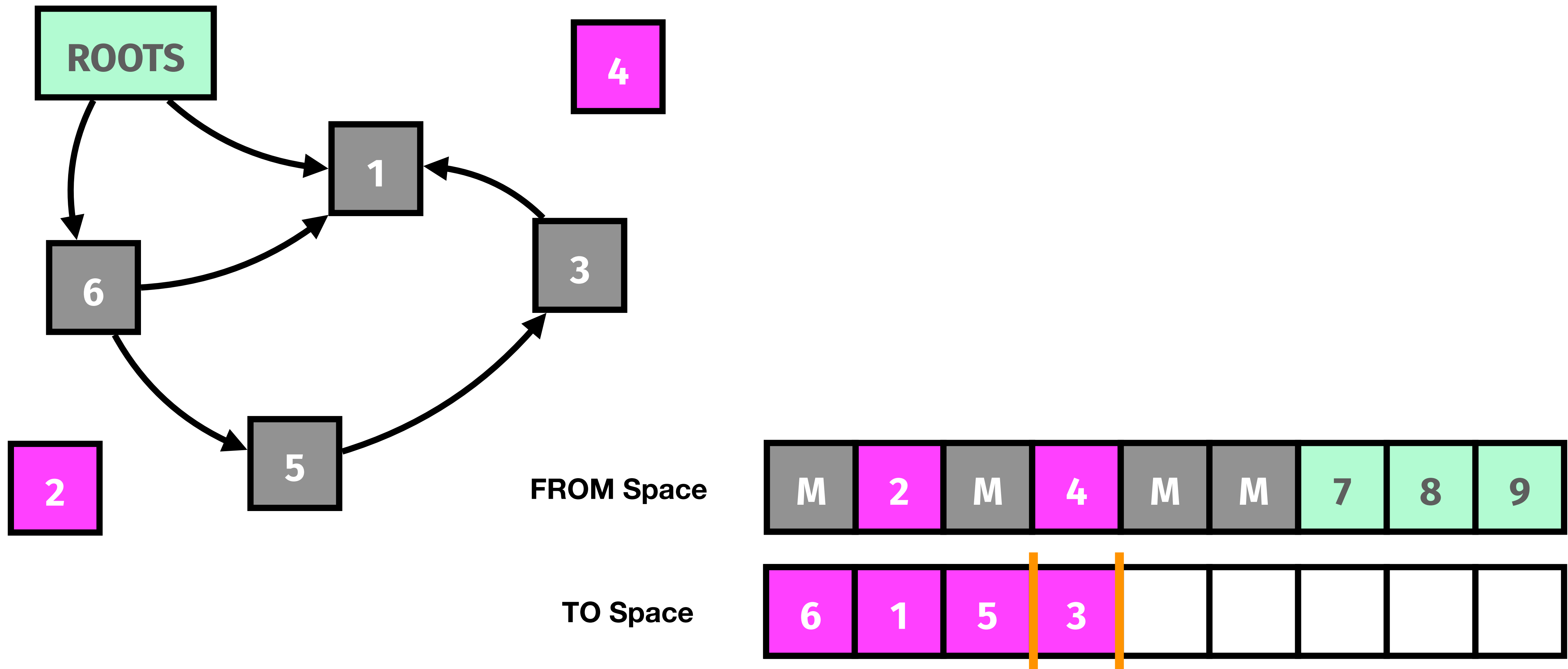
COPYING COLLECTION



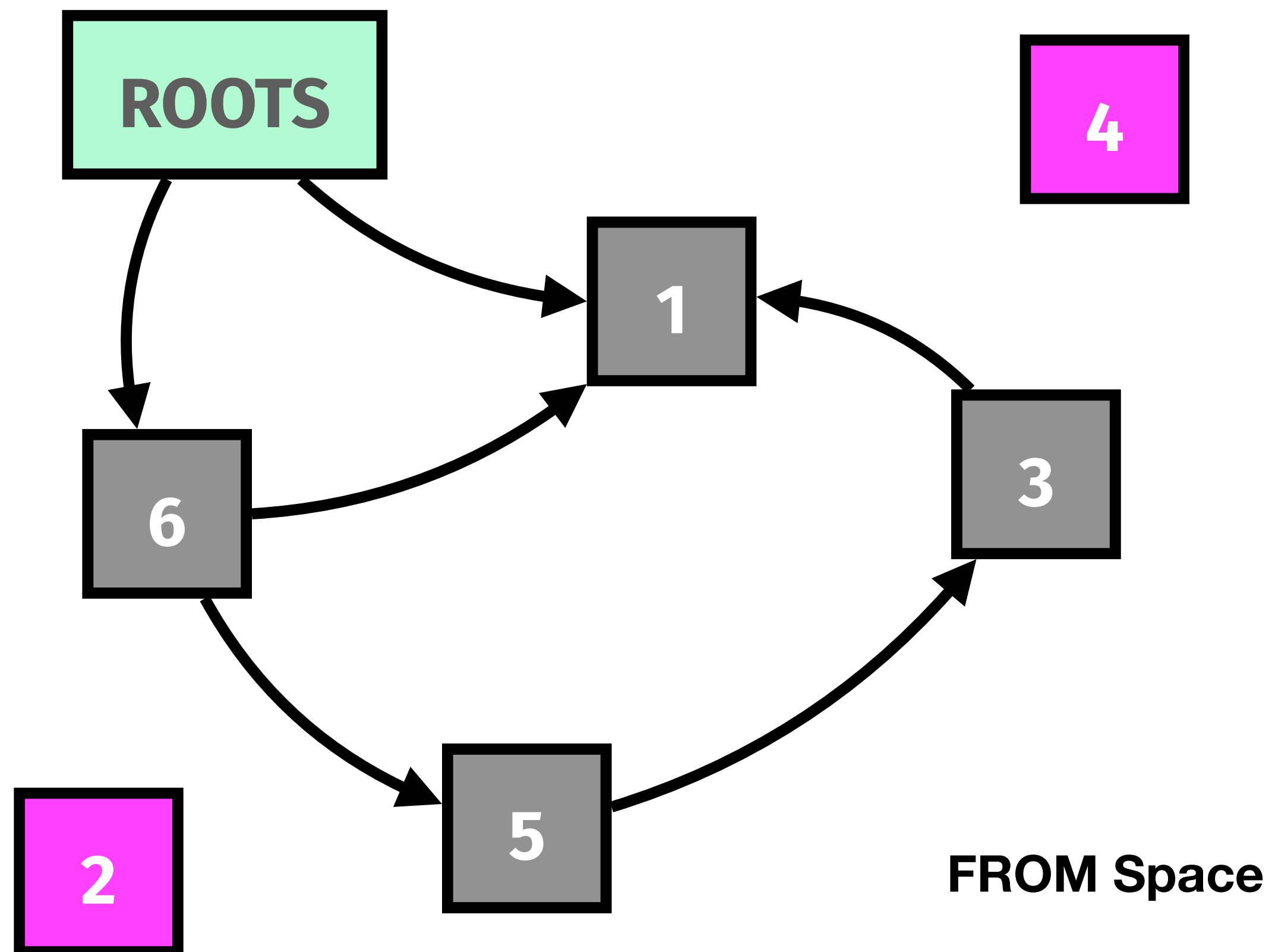
COPYING COLLECTION



COPYING COLLECTION



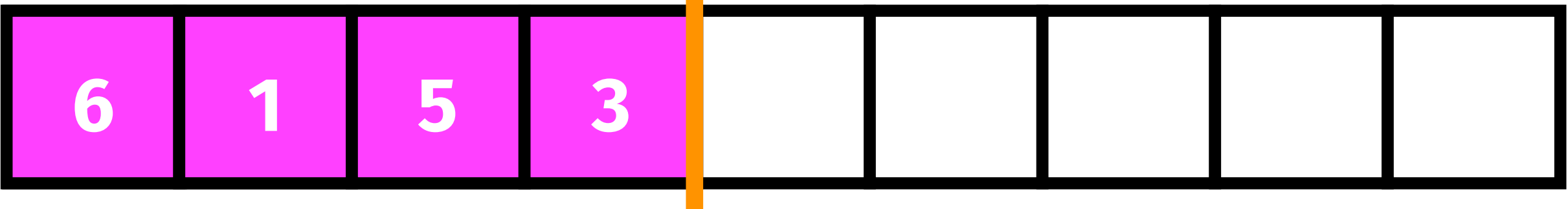
COPYING COLLECTION



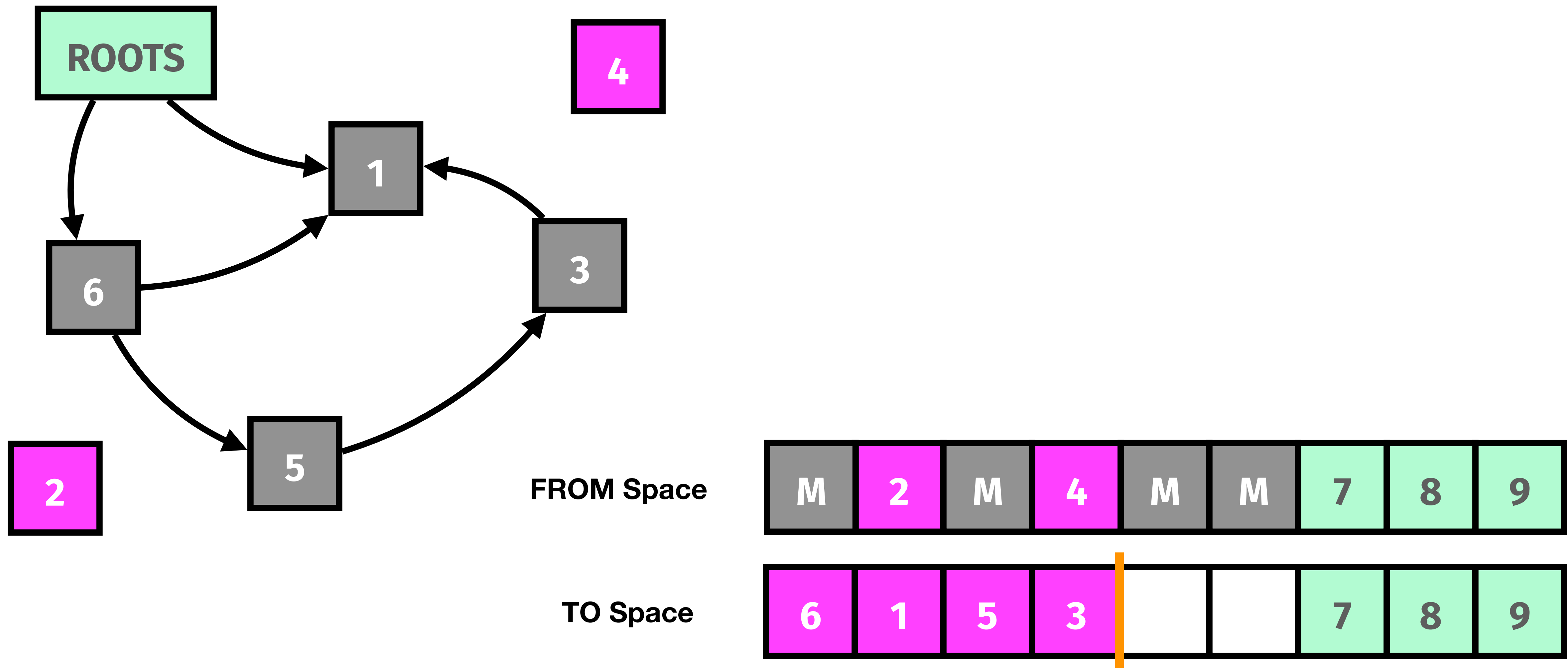
FROM Space



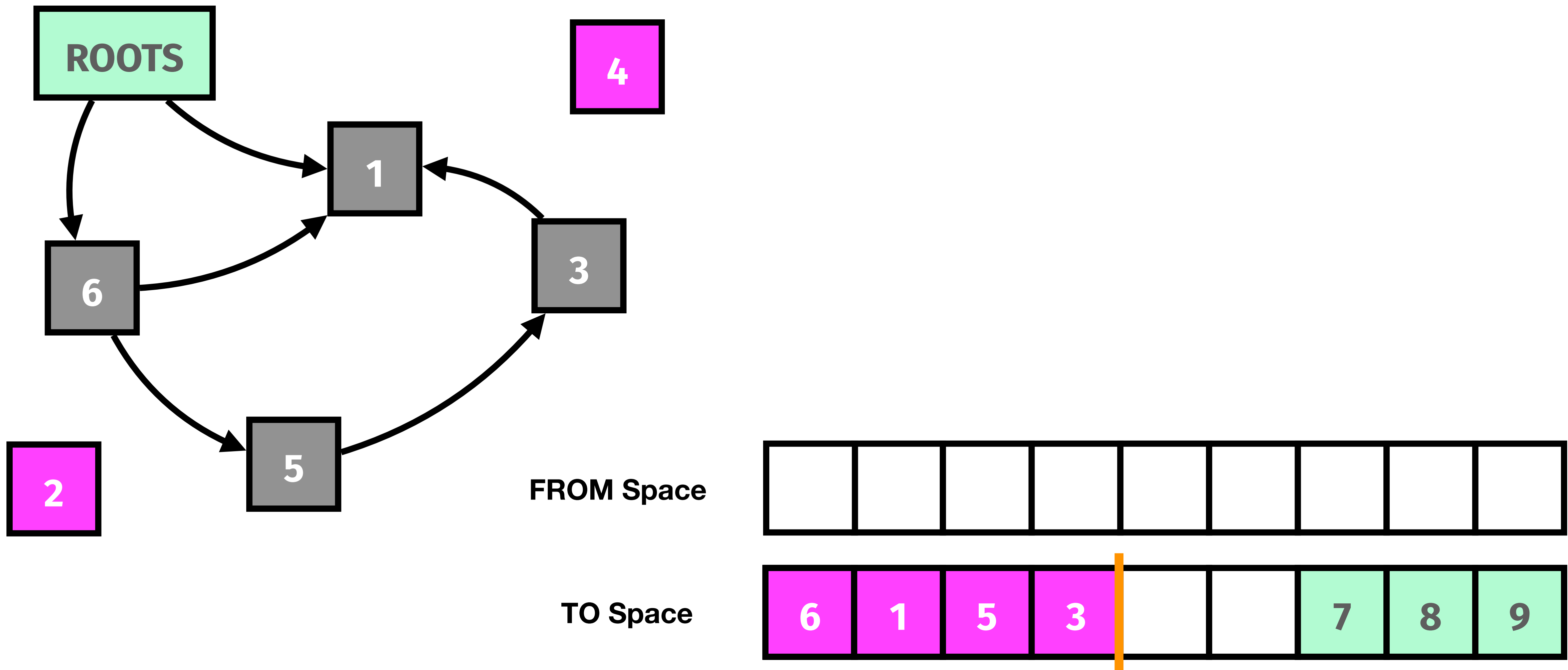
TO Space



COPYING COLLECTION

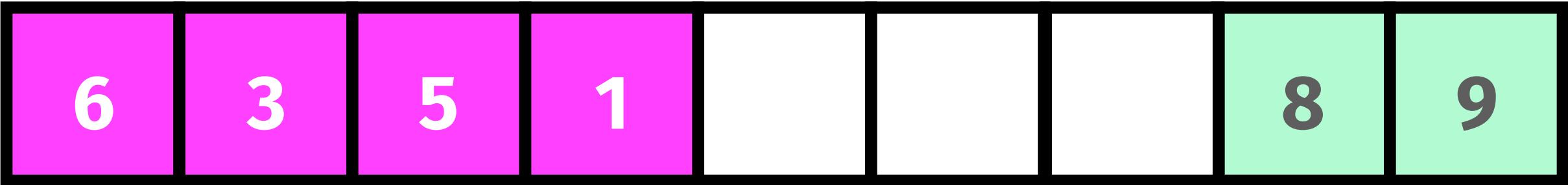


COPYING COLLECTION

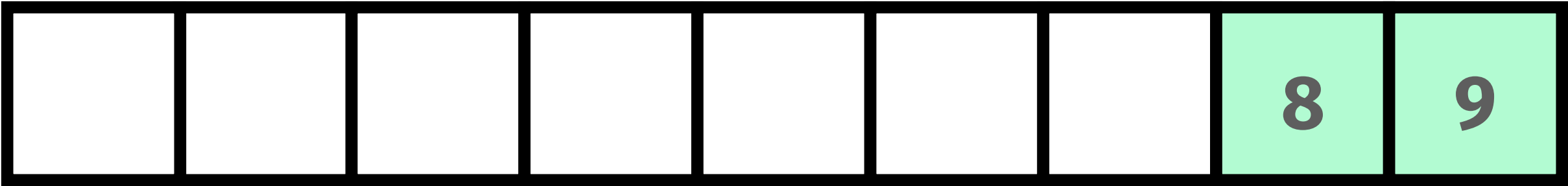


GENERATIONAL COLLECTION

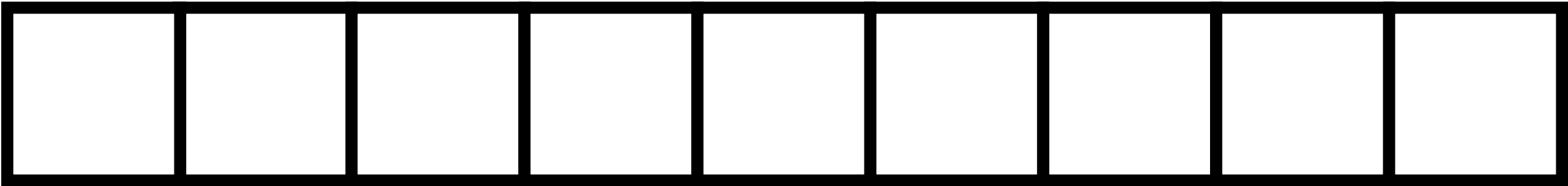
FROM Space



TO Space

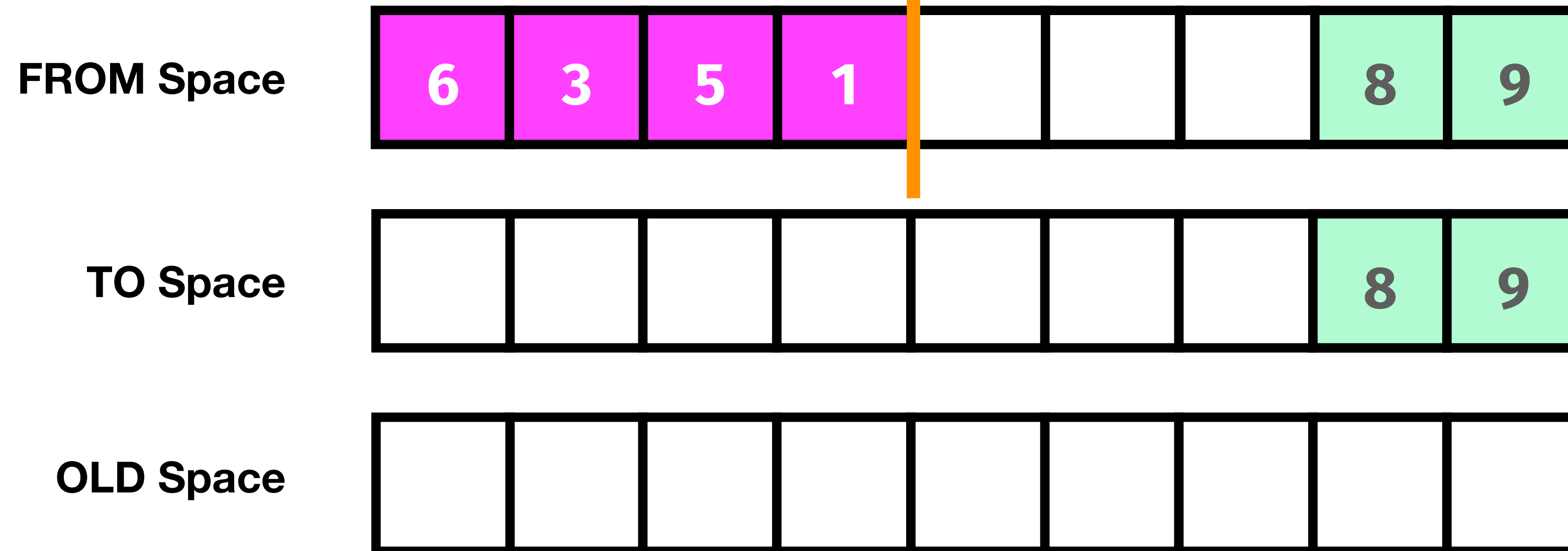


OLD Space

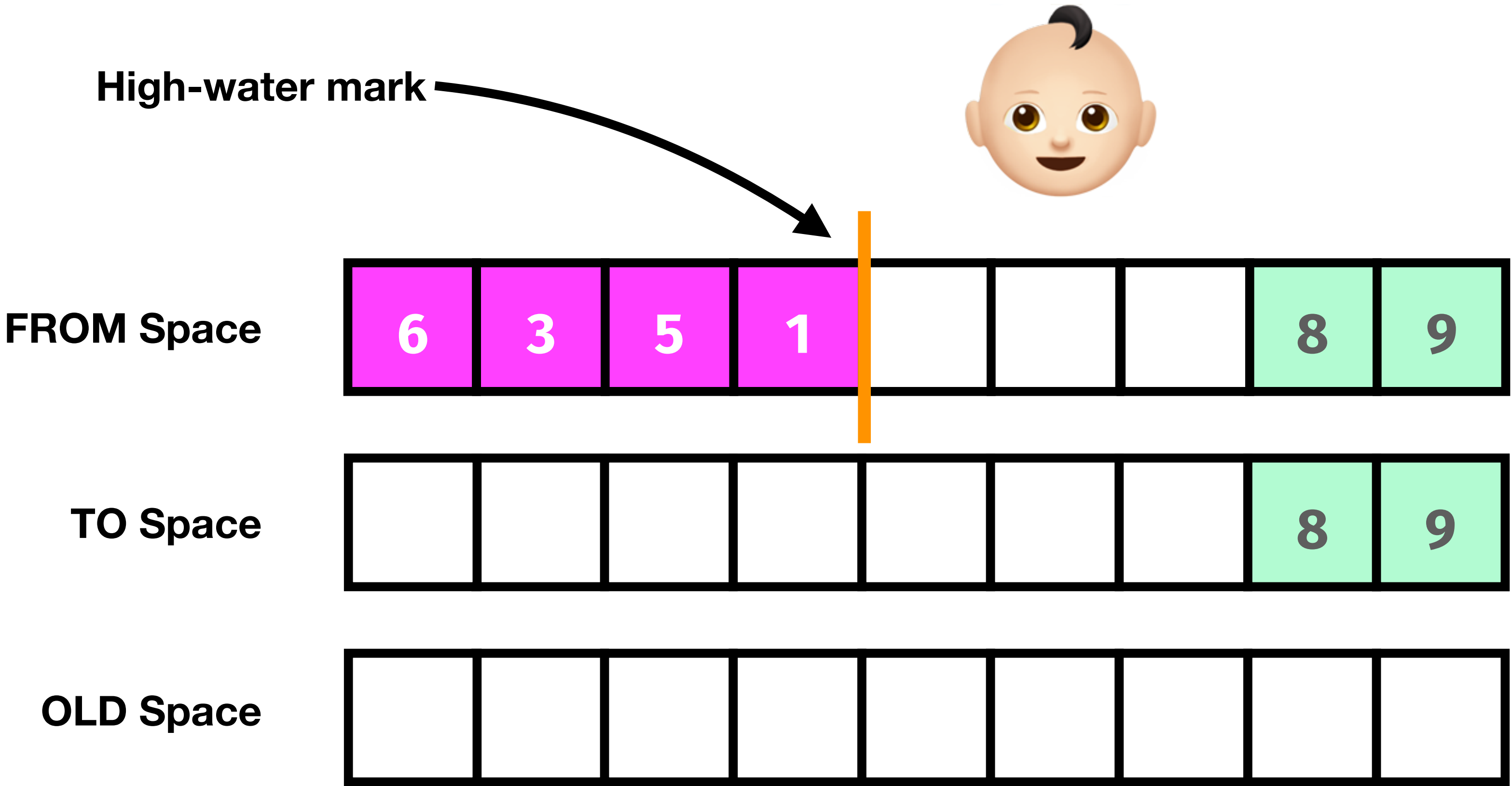


GENERATIONAL COLLECTION

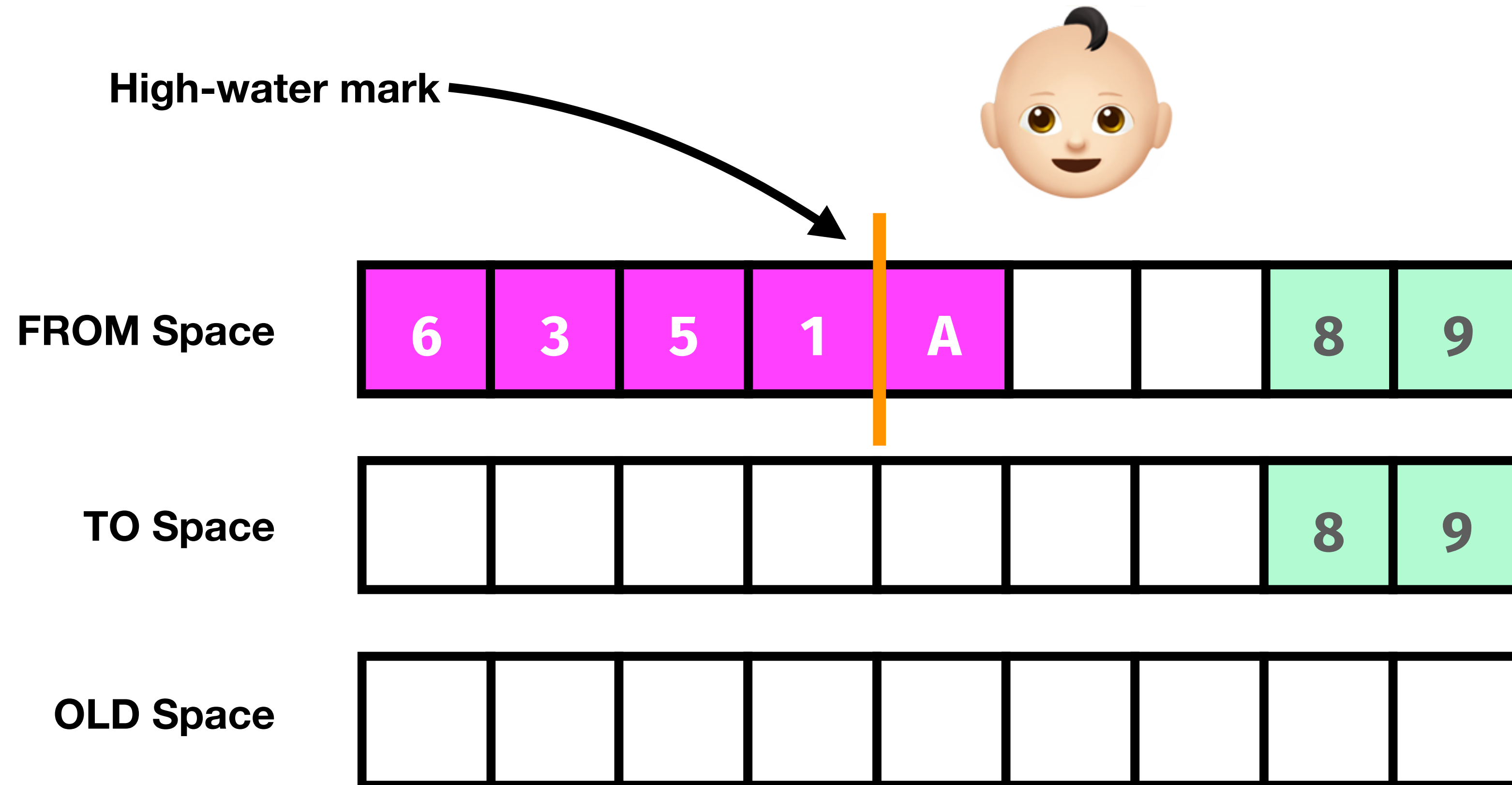
High-water mark



GENERATIONAL COLLECTION



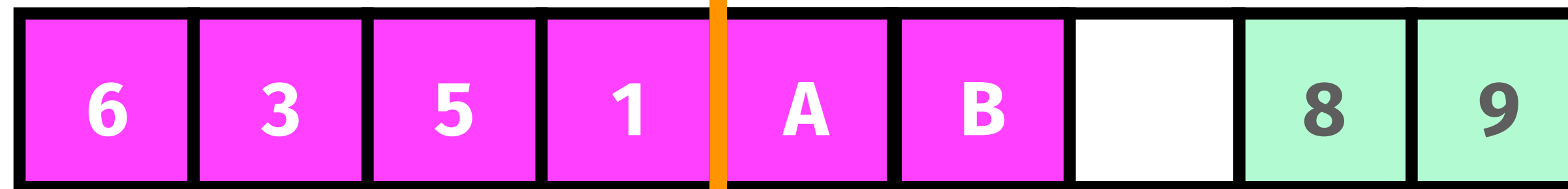
GENERATIONAL COLLECTION



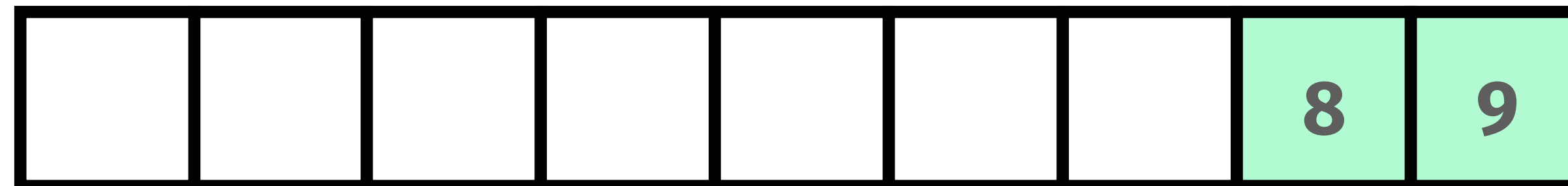
GENERATIONAL COLLECTION



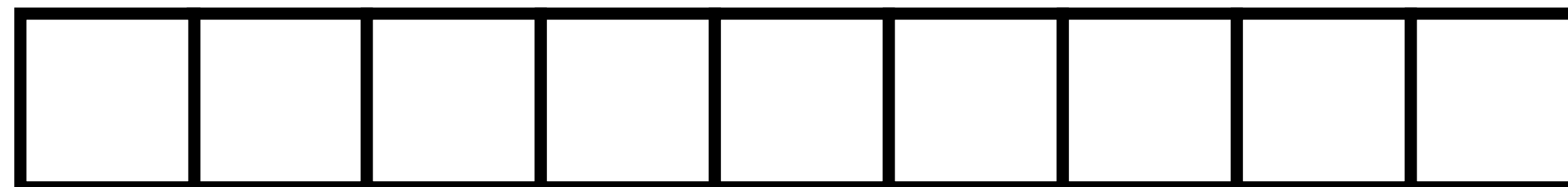
High-water mark



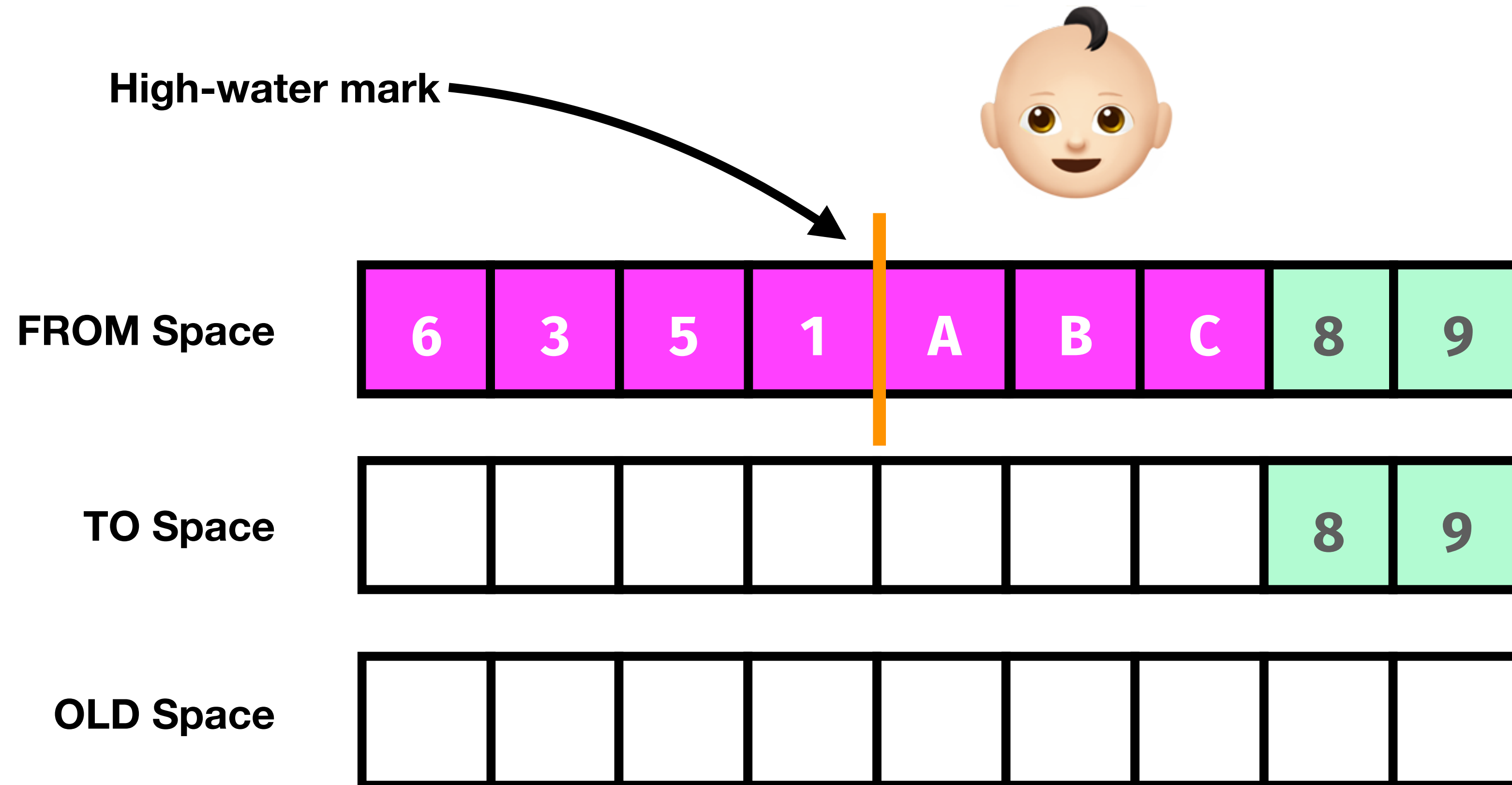
TO Space



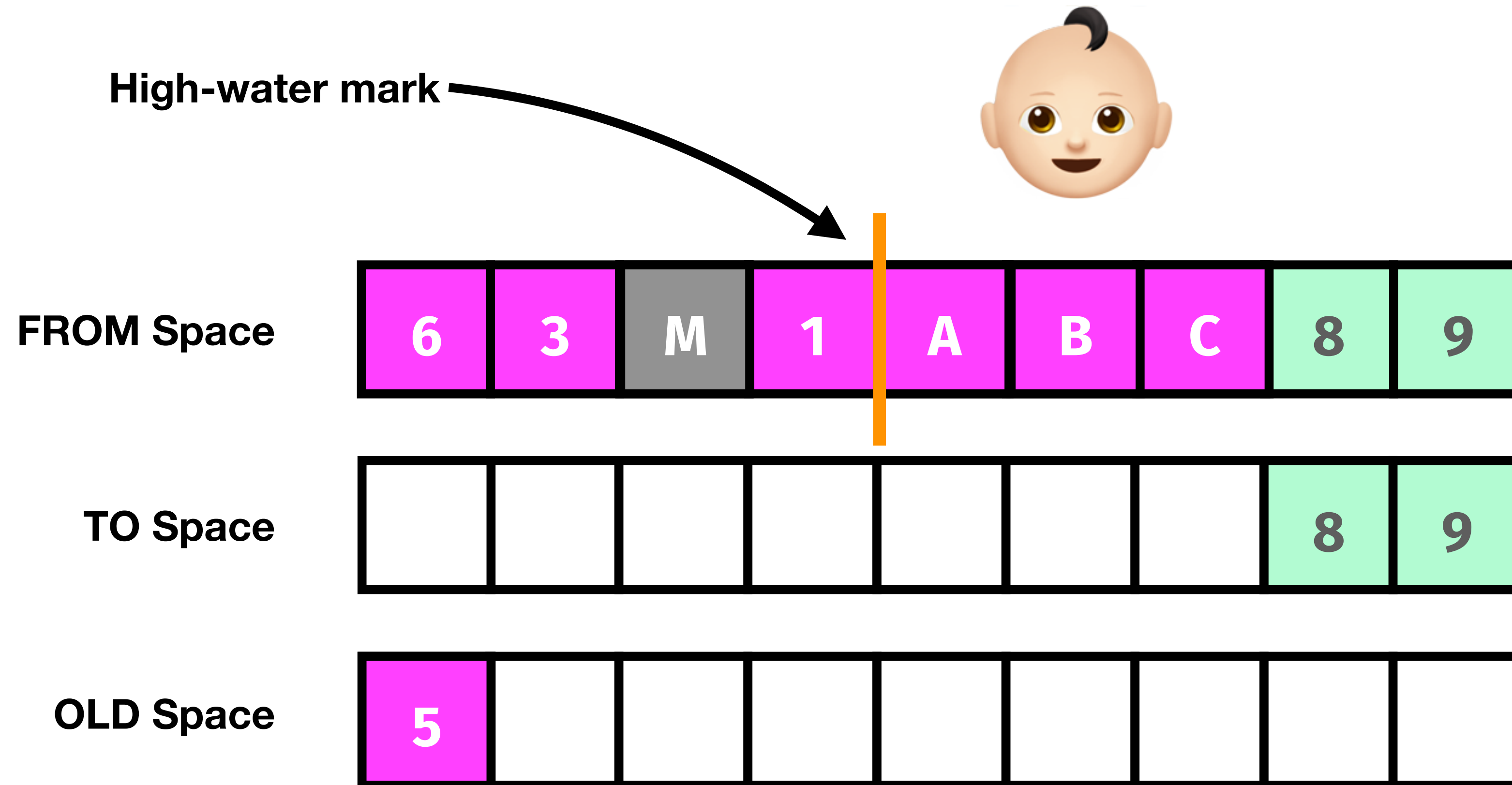
OLD Space



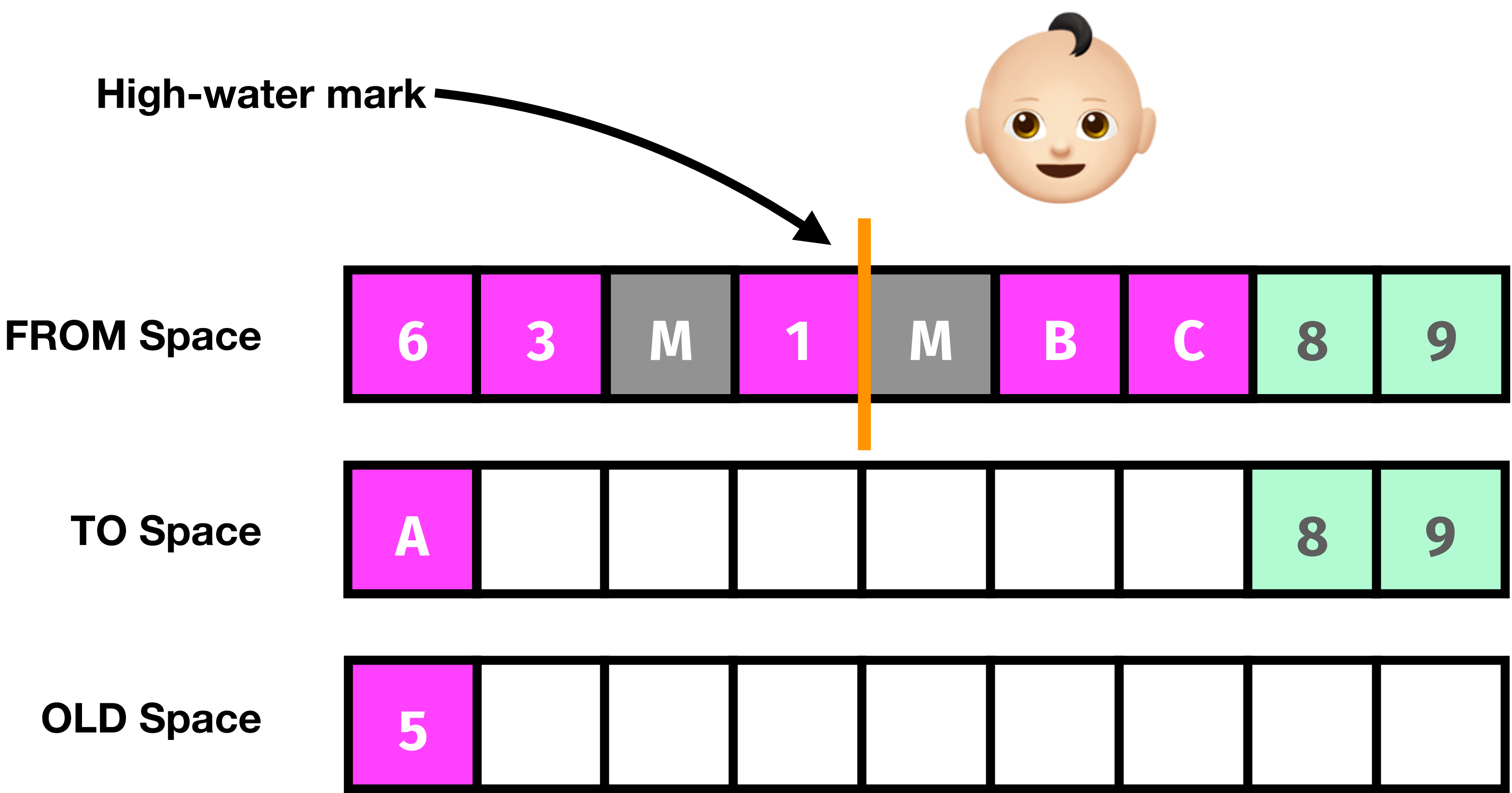
GENERATIONAL COLLECTION



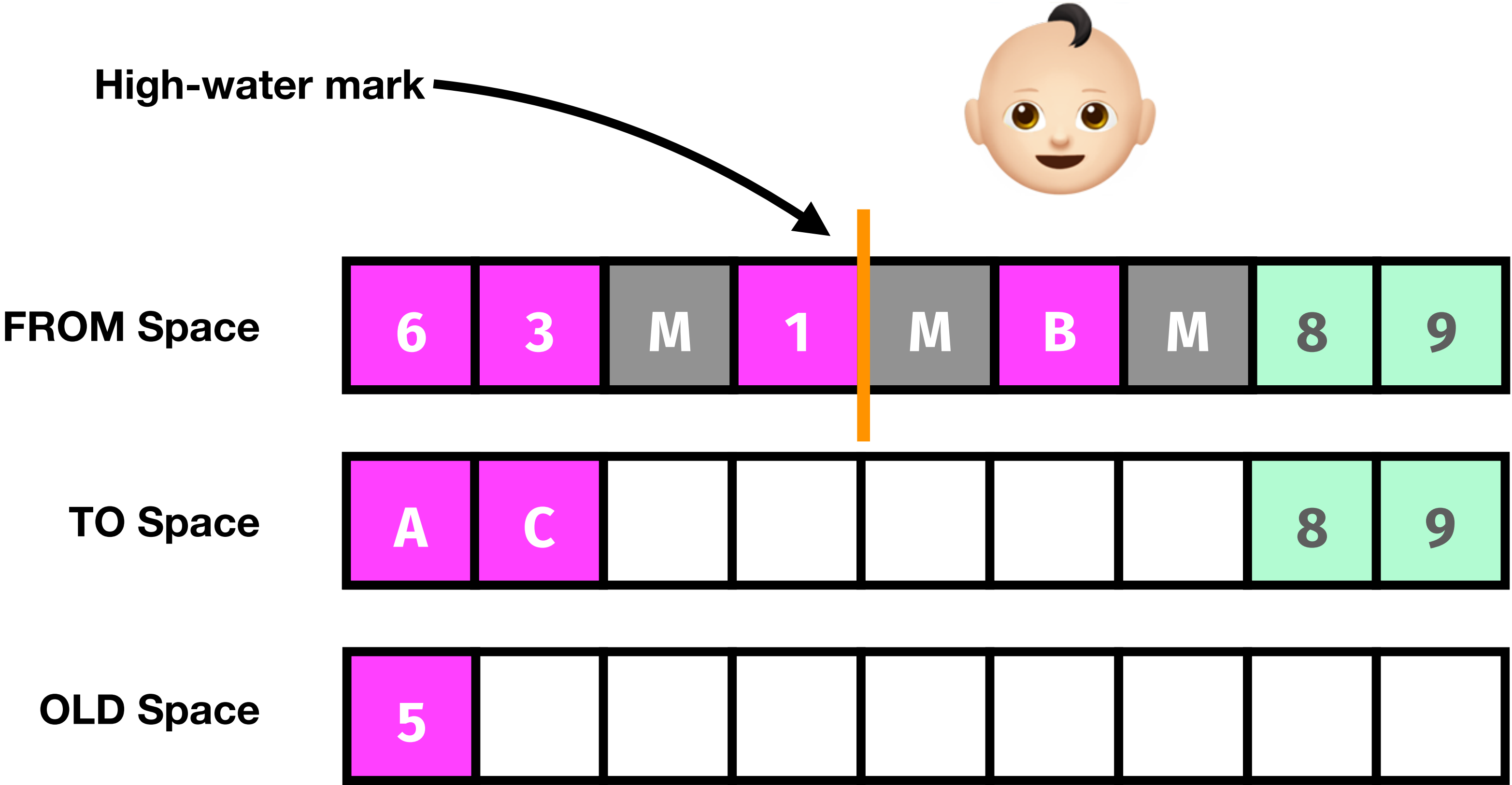
GENERATIONAL COLLECTION



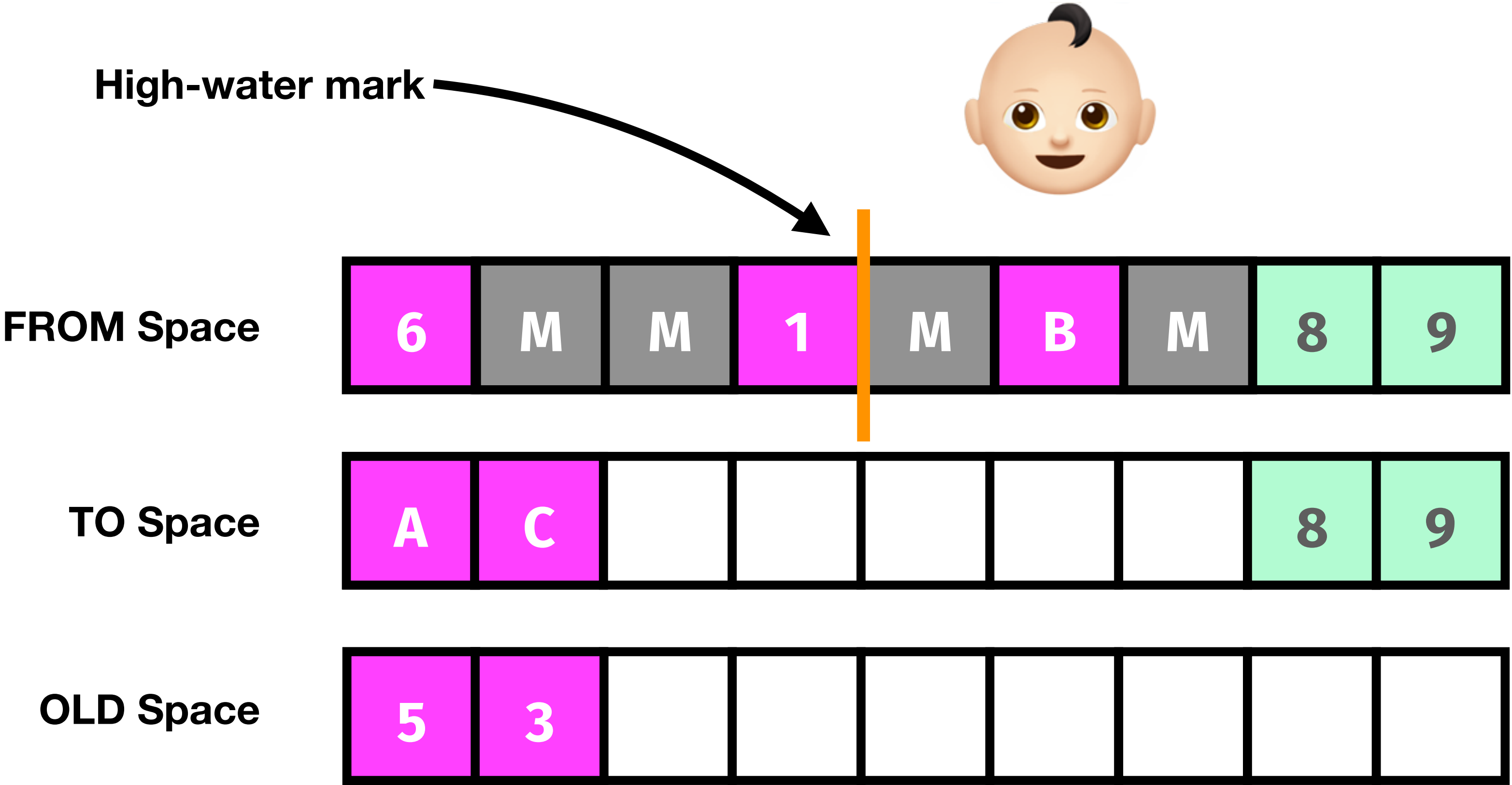
GENERATIONAL COLLECTION



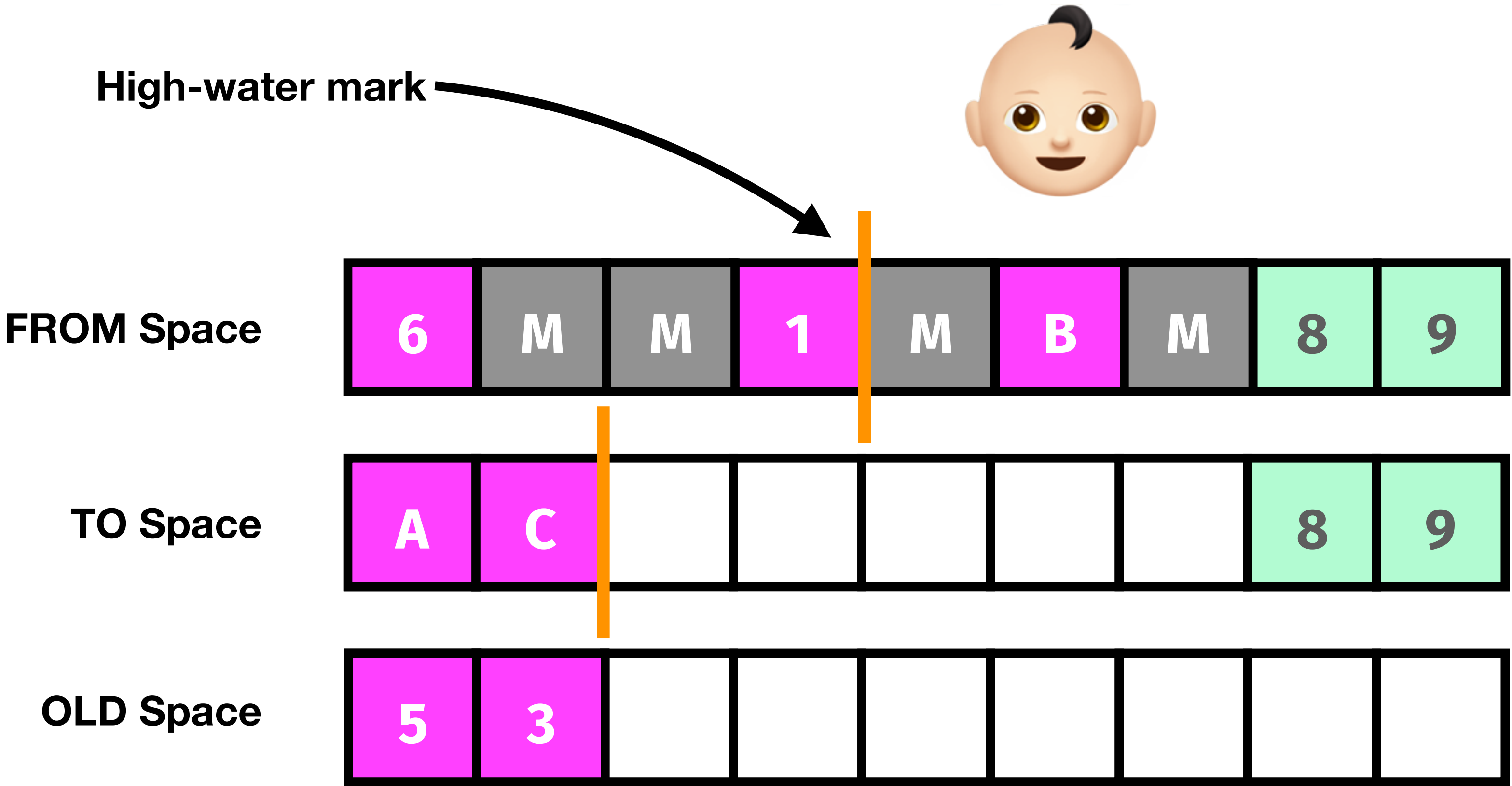
GENERATIONAL COLLECTION



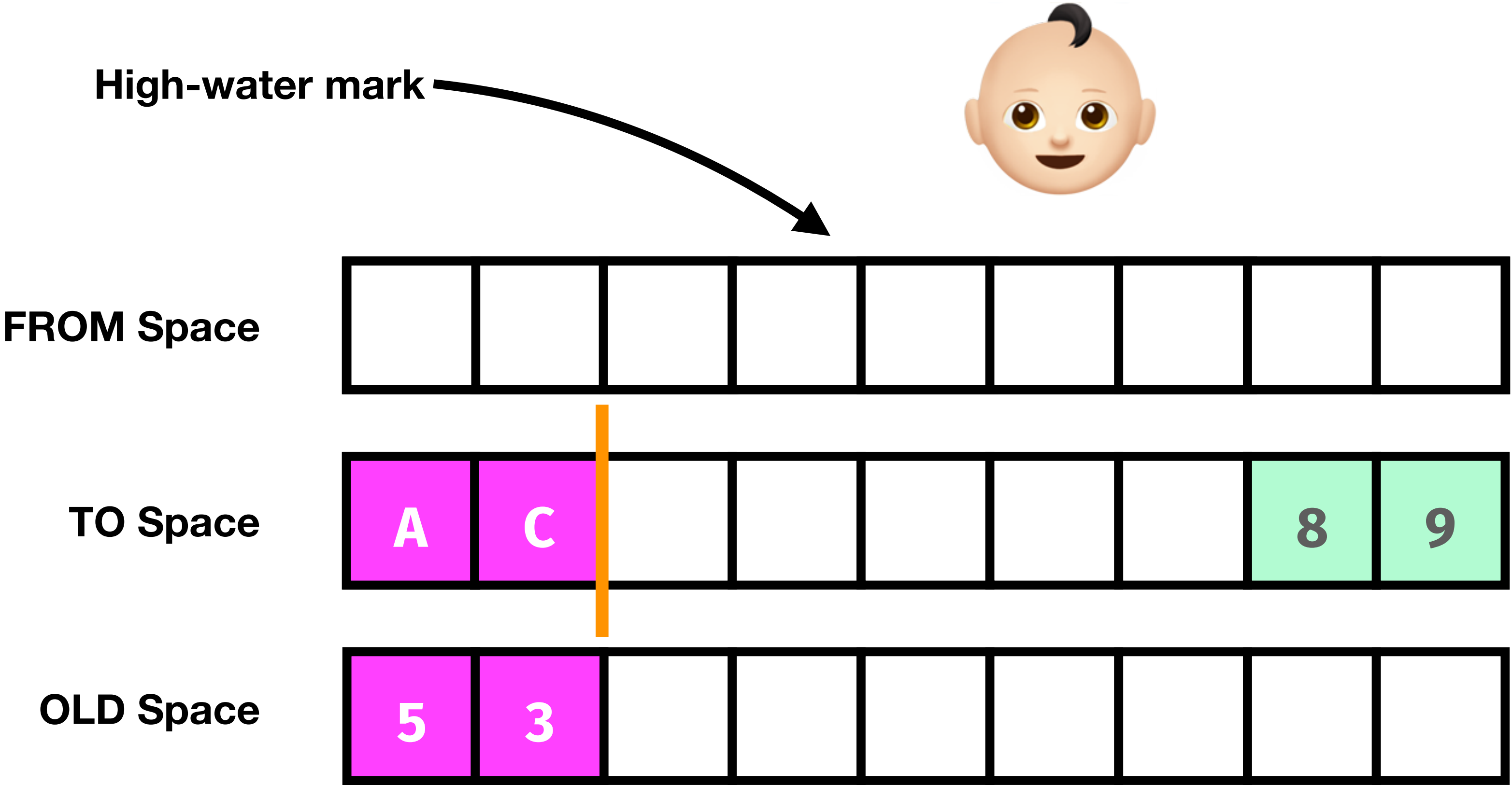
GENERATIONAL COLLECTION



GENERATIONAL COLLECTION



GENERATIONAL COLLECTION



BINARY HEAP

BINARY HEAP



Binaries > 64 bytes

BINARY HEAP



Binaries > 64 bytes



Reference counted



Garbage collection is no longer magic!

LET ME KNOW WHAT YOU THINK!

Tweet me @sannekalkman

Find me in person later!

The Garbage Collection Handbook - Jones, Hoskin & Moss

Erlang Garbage Collector - Lukas Larsson (Erlang Solutions)