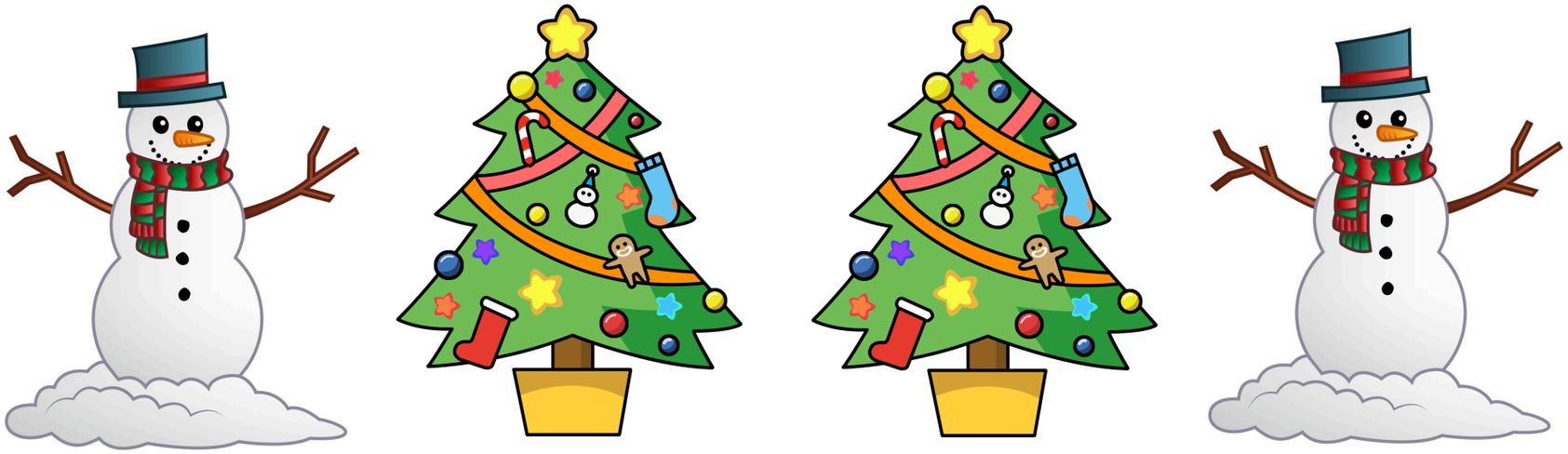
Superpermutations

1)

2)

“



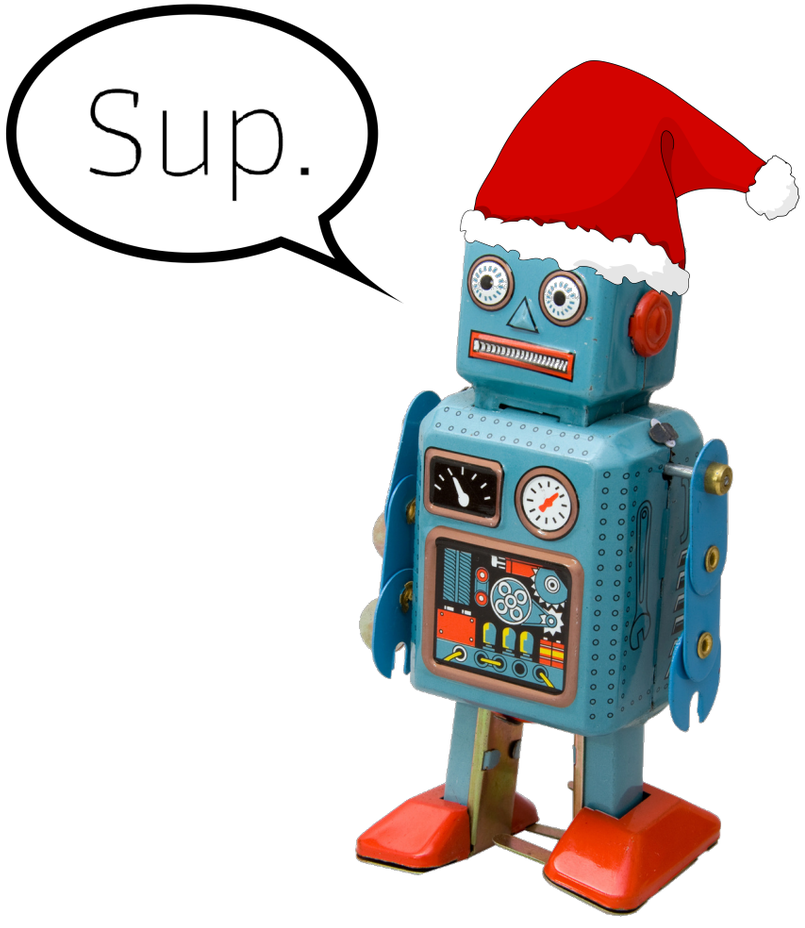
”

“

”

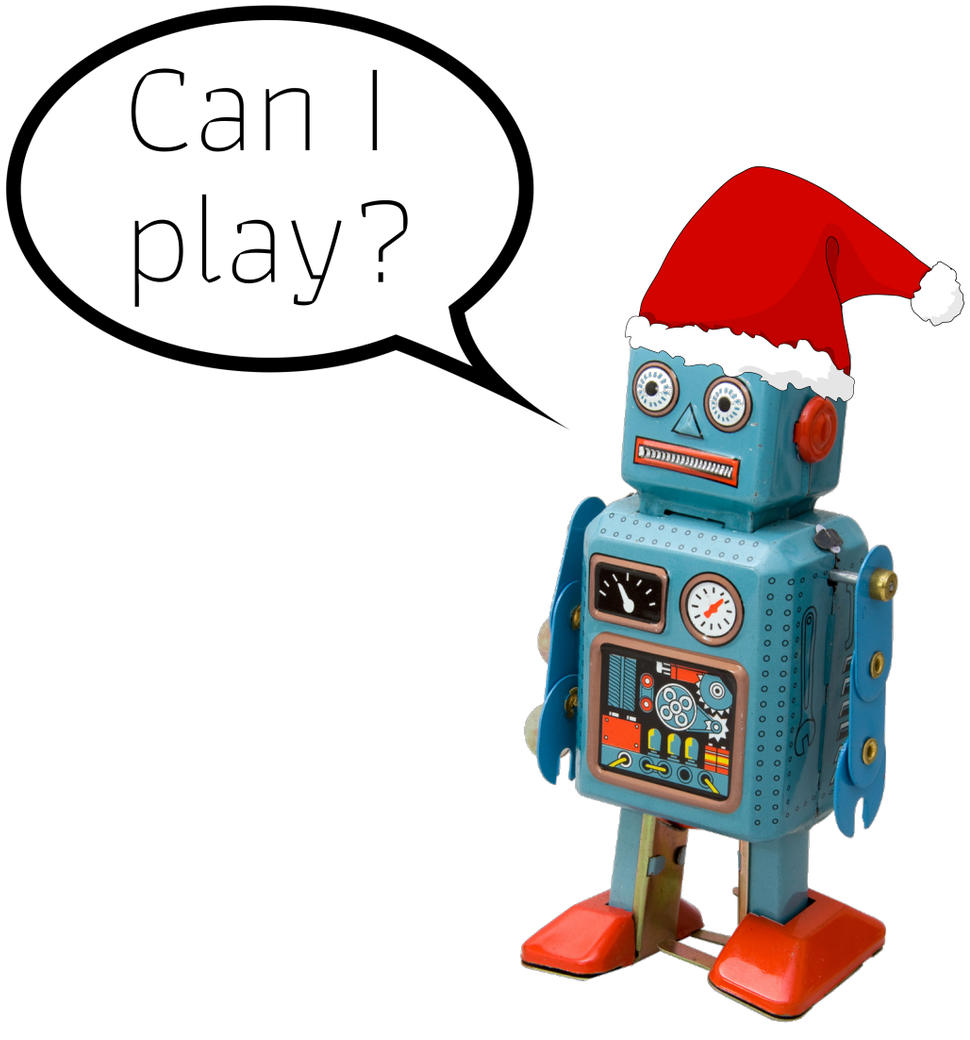
“

”


Sup.

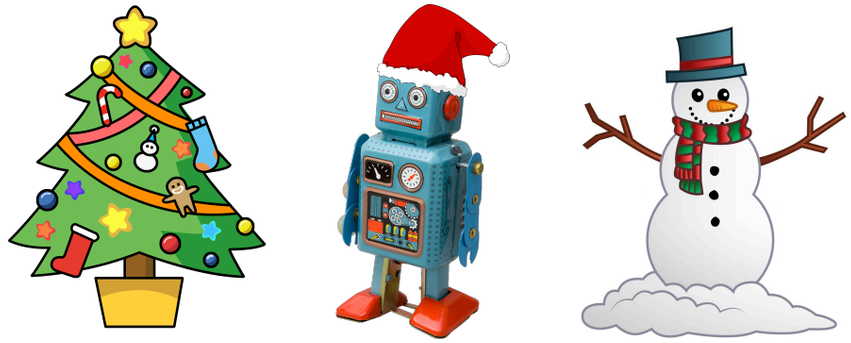
“

”

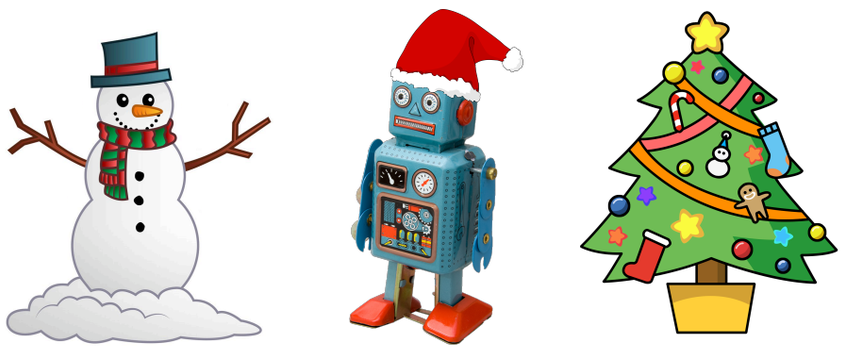

Can I
play?

1)

2)



3)



4)

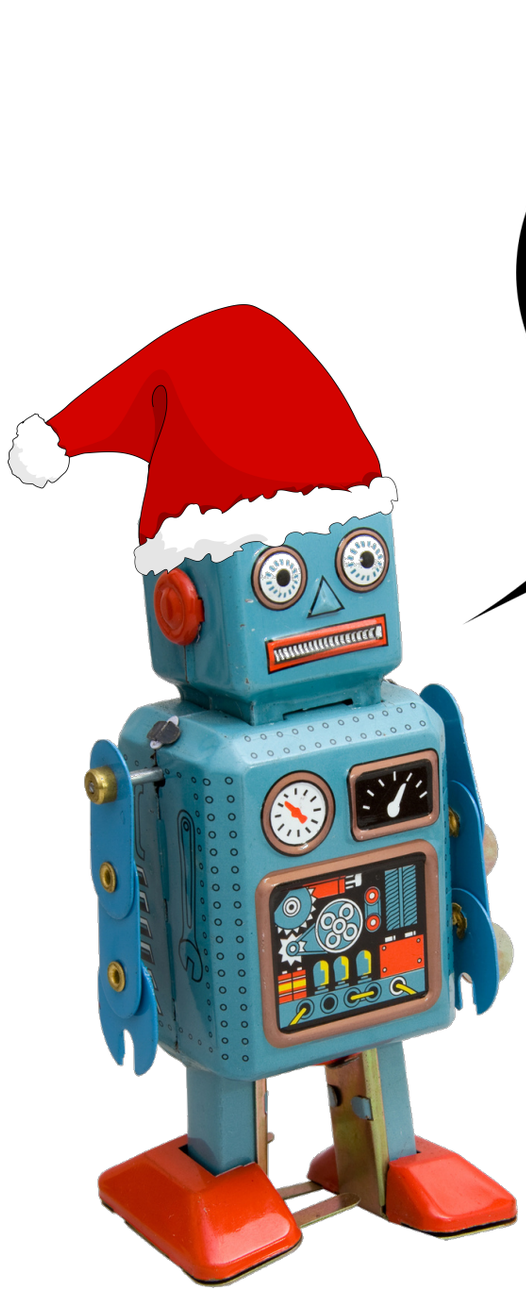
5)

6)

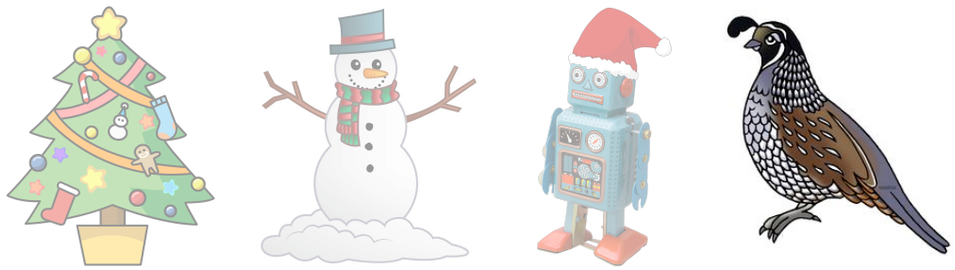
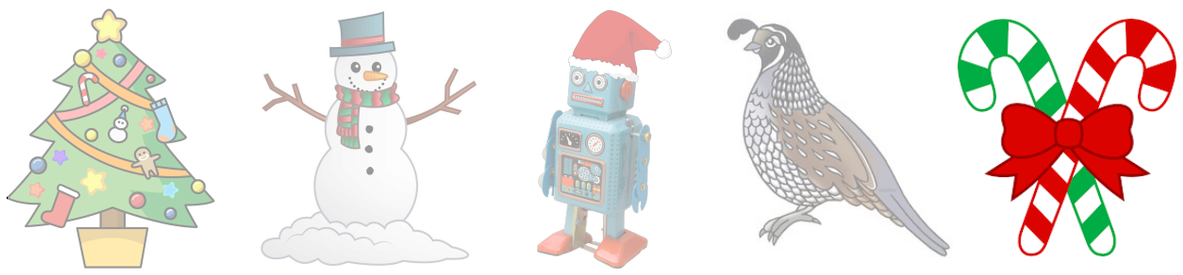
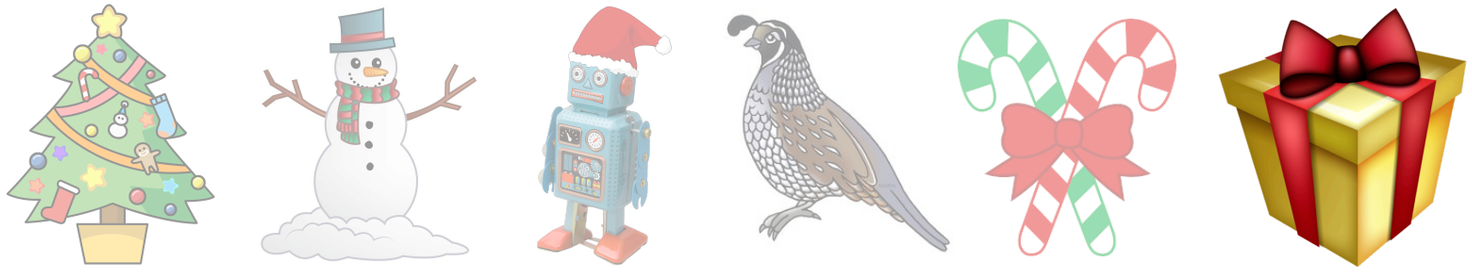
“  ”

(9 symbols)


Squaaawk...
Can I play?



No, 'cause tuzz will get RSI

Number of Symbols	Length of Shortest
	9
	33
	153
	???

The Shortest Known (N=6)

123456123451623451263451236451326451362451364251364521364512346512341562341526341
523641523461523416523412563412536412534612534162534126534123564123546123541623541
263541236541326543126453162435162431562431652431625431624531642531462531426531
425631425361425316452314652314562314526314523614523164532164531264351264315264312
564321564231546231542631542361542316542315642135642153624153621453621543621534621
354621345621346521346251346215364215634216534216354216345216342516342156432516
432561432564132564312654321654326153426135426134526134256134265134261532465132465
312463512463152463125463215463251463254163254613254631245632145632415632451632456
132456312465321465324165324615326415326145326154326514362514365214356214352614
352164352146352143651243615243612543612453612435612436514235614235164235146235142
63514236514326541362541365241356241352641352461352416352413654213654123

(872 symbols)

… but is it the shortest?

5 symbols was proven with ‘branch and bound’

```
(0..Infinity).each do |w|
  p = “What’s the maximum number of
      permutations that can fit into a
      string that ‘wastes’ w symbols?”
  break if p == 5!
  prune_search_space(w, p)
end
```

This takes ~2 minutes

But it won’t scale to 6 symbols

This universe would die first

Can we do better?

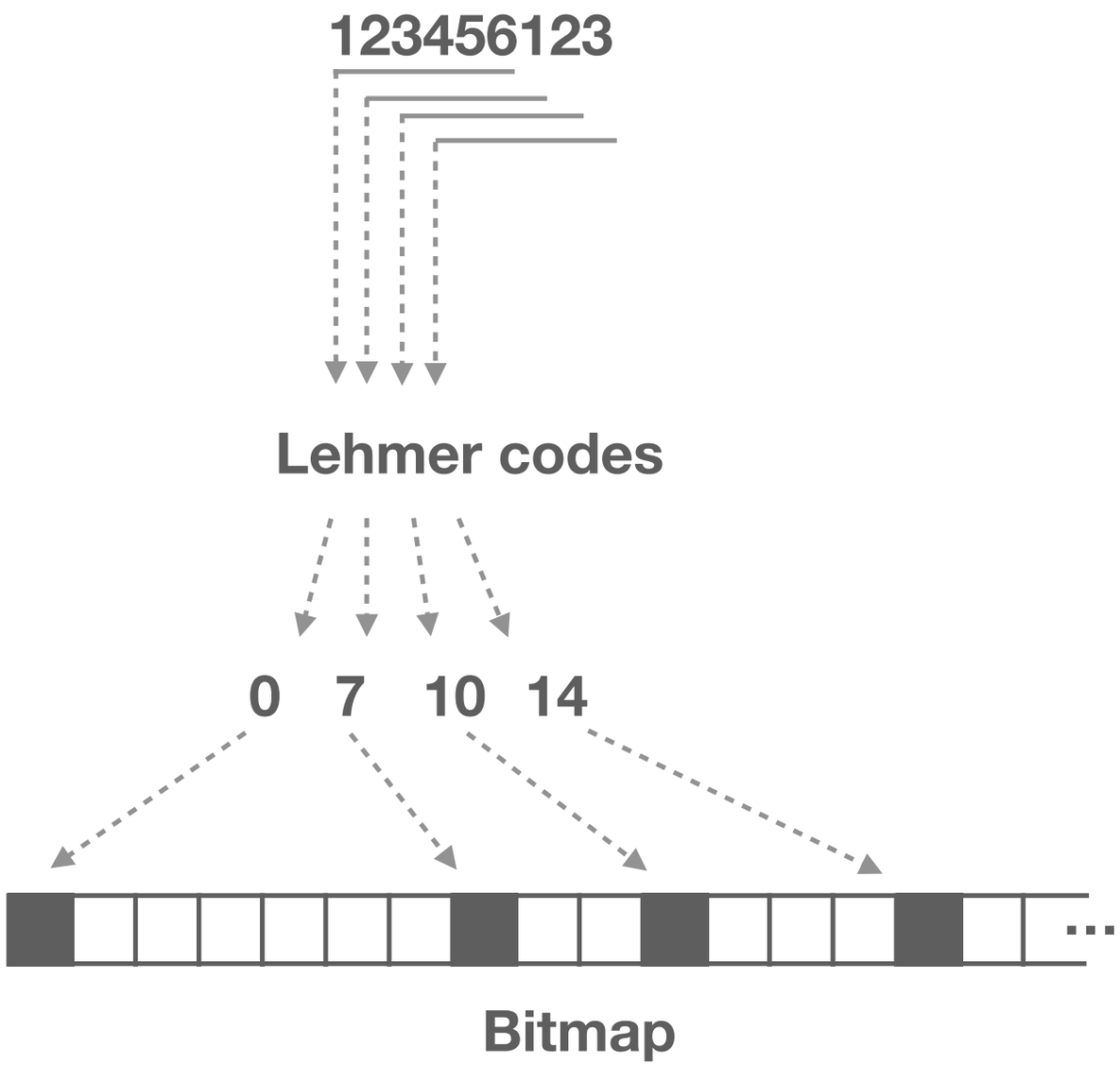
Some clever people found a reduction to the Travelling Salesman Problem

Every permutation is a 'city' to visit

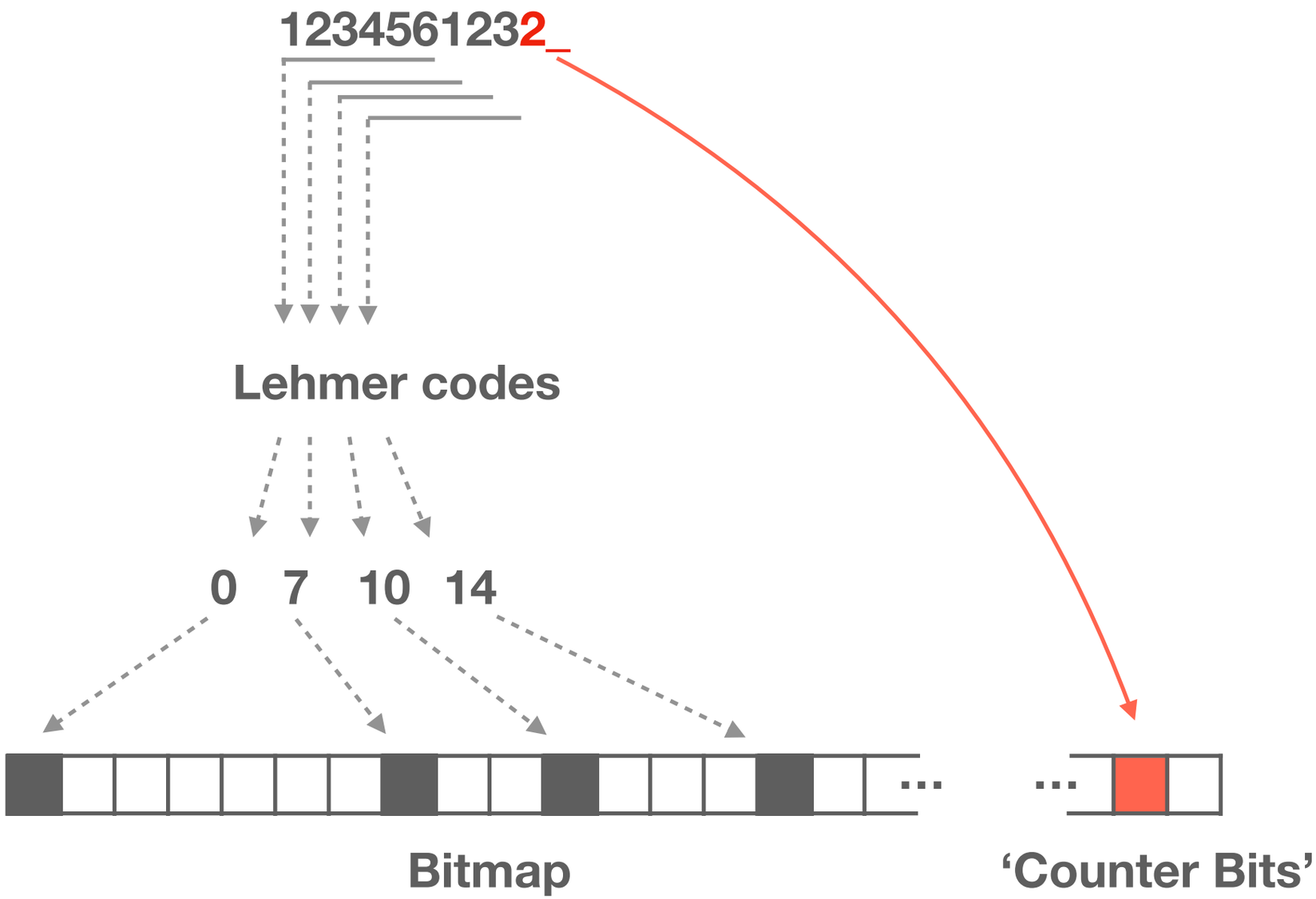
I managed to reduce it to the Boolean Satisfiability Problem, but that doesn't seem too promising

… but I've also found a reduction to the Shortest Path Problem, that seems *more* promising

The reduction:

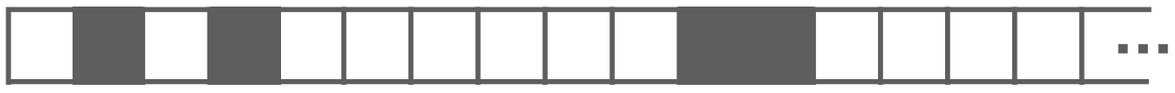


The reduction:



‘Symmetry breaking’

123456123 

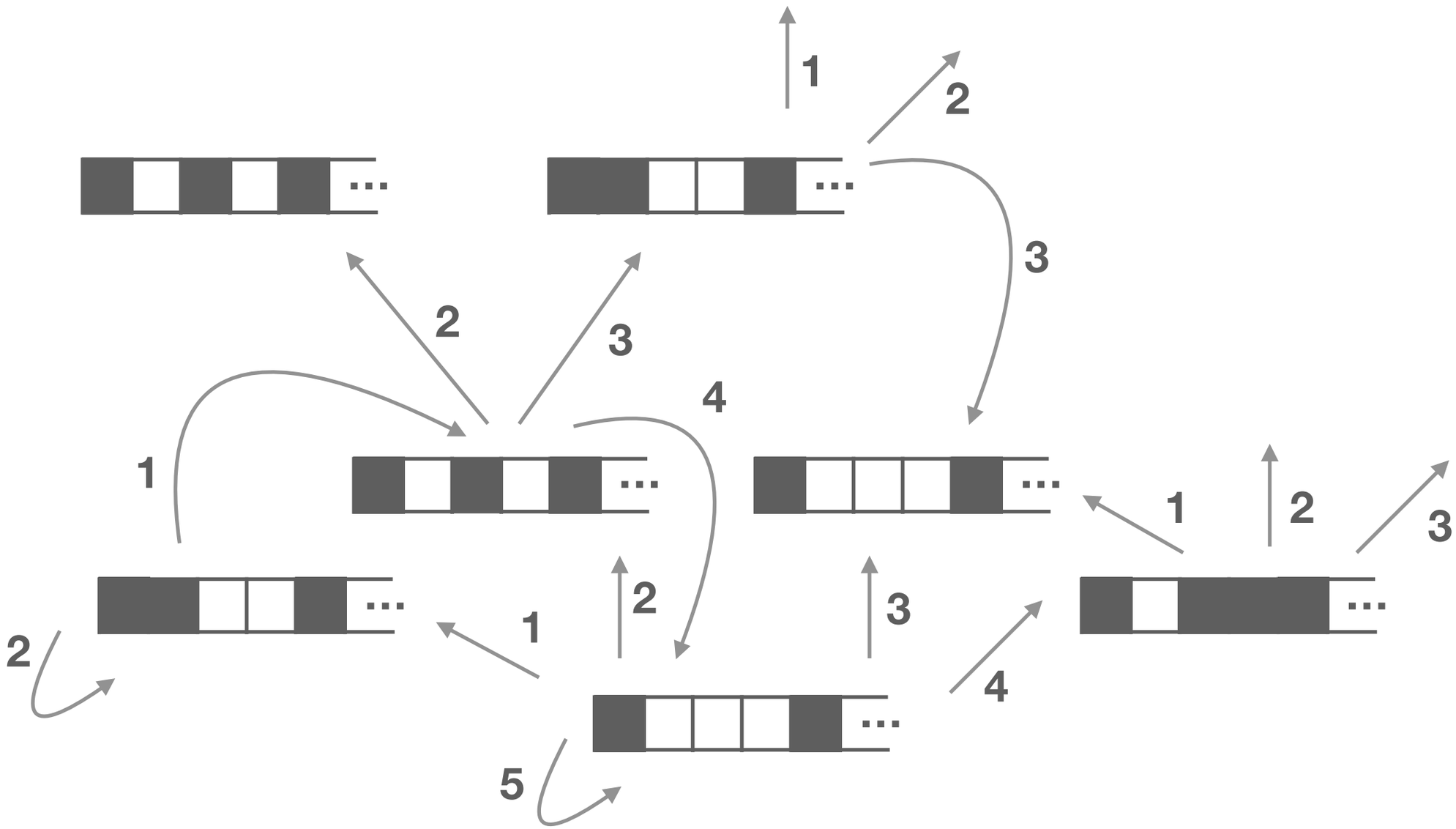
213465213 

321456321 

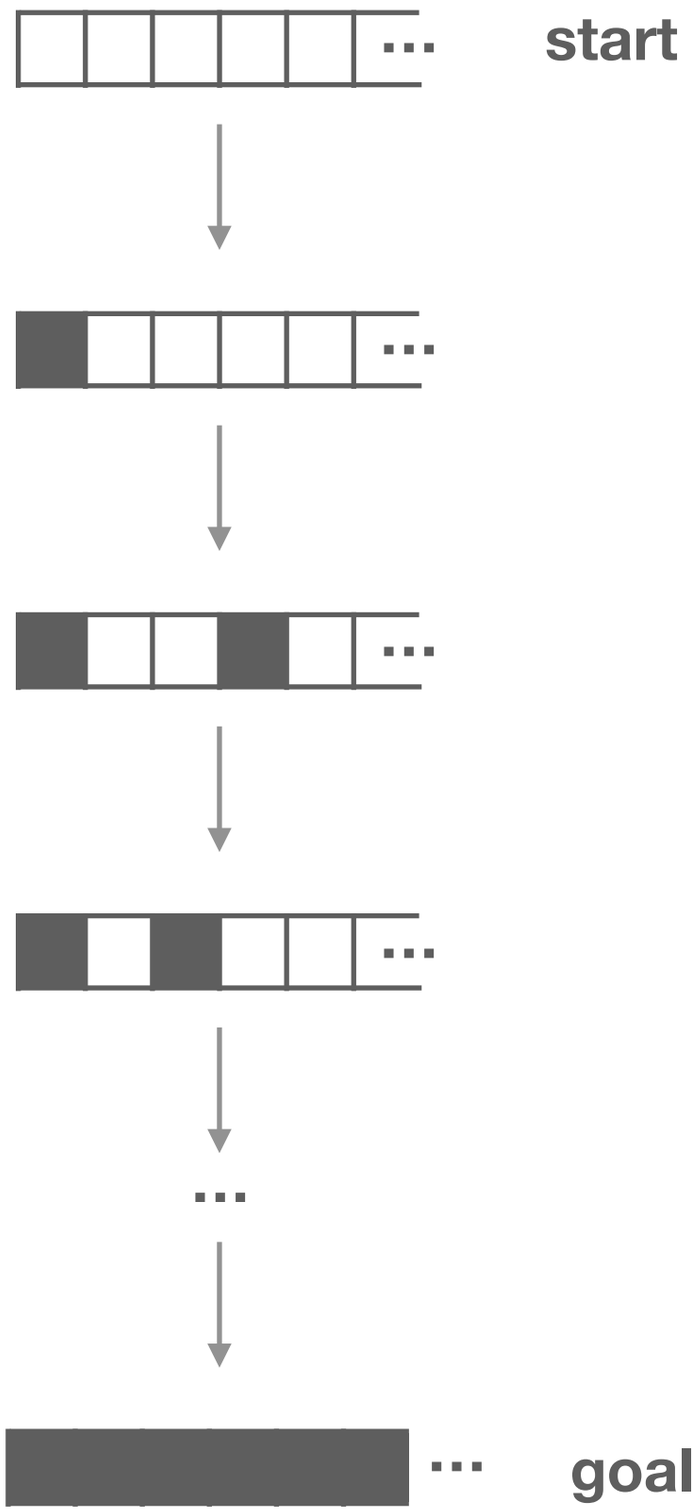
412345412 

… …. (N! possible relabelings)

Directed cyclic graph:



Shortest path:



Shortest path algorithms:

A*

Breadth-First

Frontier Search

Dijkstras

Depth-First

Adaptive A*

Branch and Bound

Best-First

D* Lite

Jump Point

IDA*

Shortest path algorithms:

A*

Breadth-First

Frontier Search

Dijkstras

Depth-First

Adaptive A*

Branch and Bound

Best-First

D* Lite

Jump Point

IDA*

Adaptive A*:

‘Incremental Heuristic Search’ algorithm

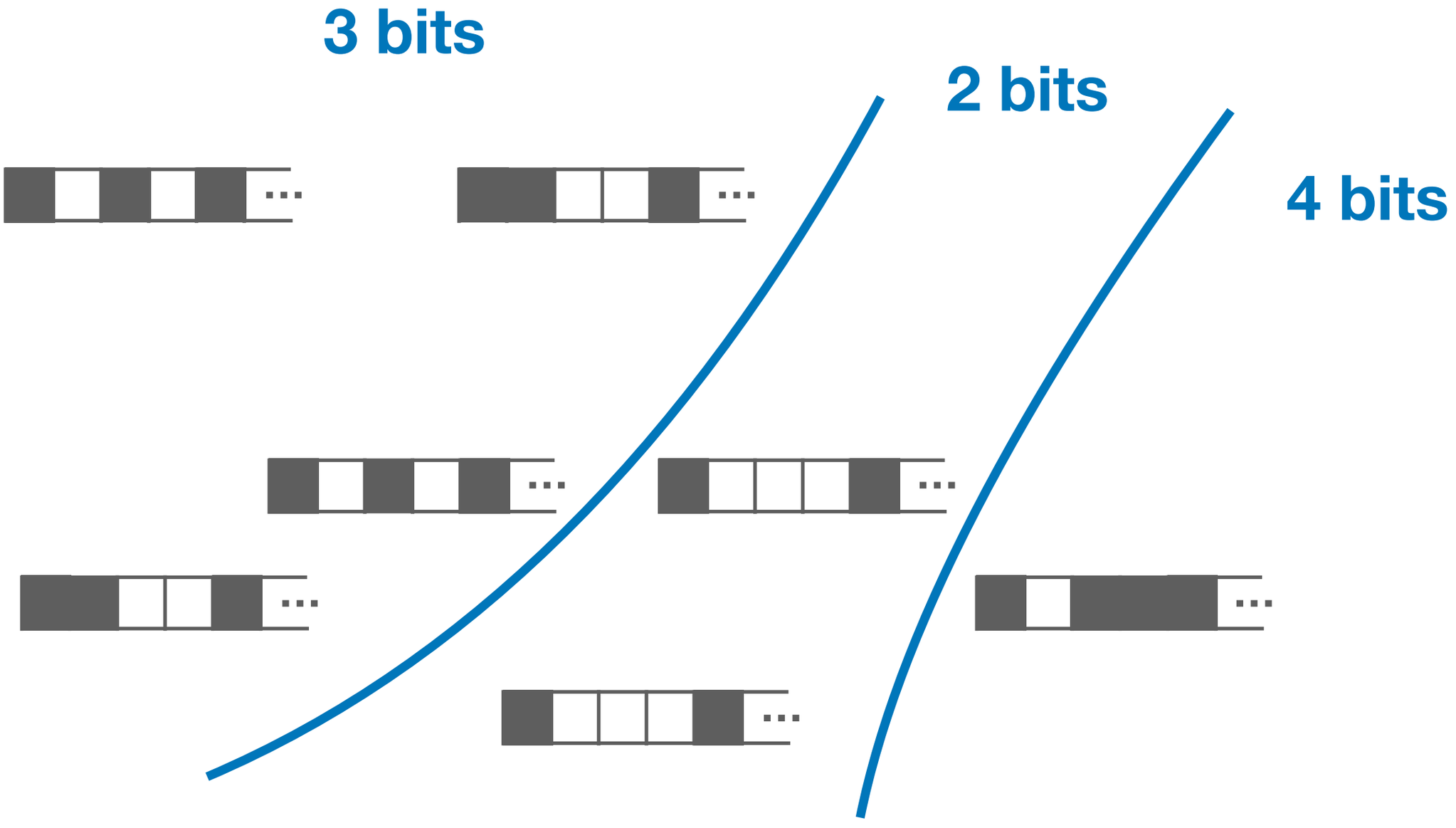
“Repeatedly solves ‘subproblems’ to learn how to solve the overall problem more quickly!”

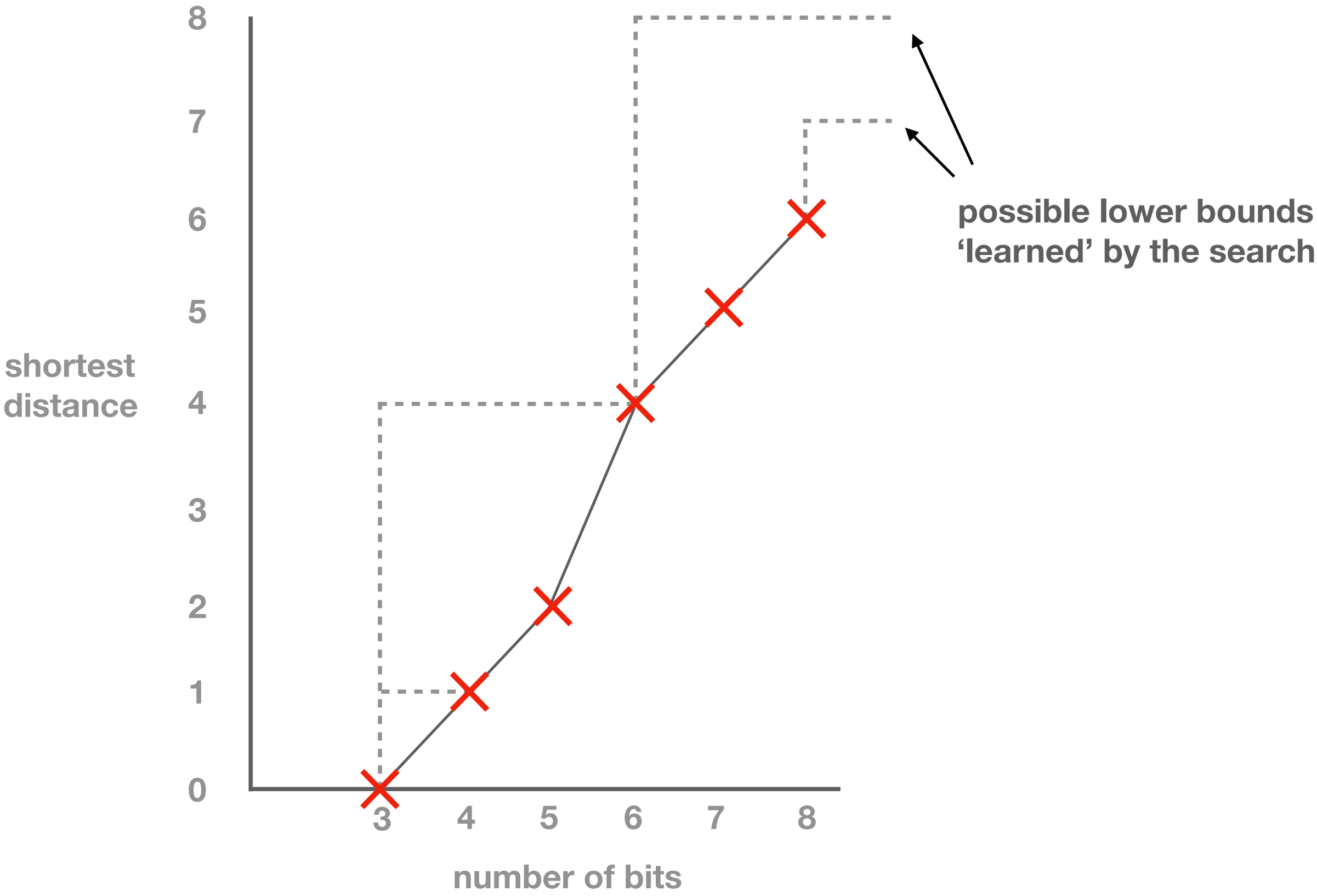
What subproblems?

“What’s the maximum number of permutations that can fit into a string that ‘wastes’ w symbols?”

“What’s the shortest path that adds n bits to the bitmap?”

‘Abstract State Space’




shortest distance
8
7
6
5
4
3
2
1
0
3
4
5
6
7
8
number of bits
possible lower bounds
‘learned’ by the search

Aaaaaanyway…

I've been building some things…

https://github.com/tuzz/bucket_queue

An efficient data structure that can be used as a priority queue

https://github.com/tuzz/leaps-and-bounds

My first attempt at solving this problem using Best-First search

(undocumented)

https://github.com/tuzz/supermutation

My current attempt at solving this problem using Adaptive A* search

(undocumented, unfinished)

I've no idea if it'll work!

(…probably not)

Thanks!