

Backtracking

Similar to brute force, but instead of
naively(sp) trying each option in a "slot",
Skip the ones that are doomed to fail!

(1) K Divisibility

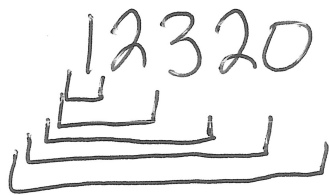
(2) 8 Queens

(3) Sudoku

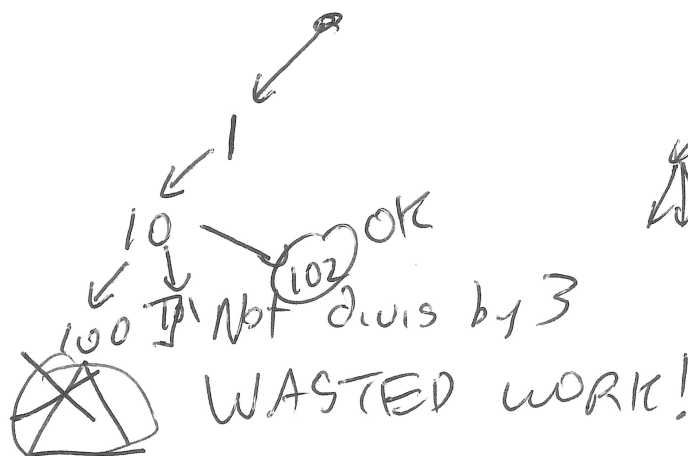
(4) Maze

★ (5) Magic Square

K Div

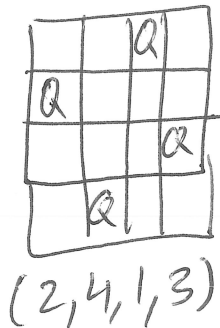


first k digits of each prefix
are divisible by k .



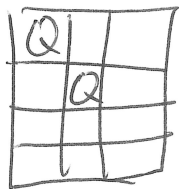
IDEA: Only
place a digit if
doing so preserves
divisibility rule!

8 Queens



~~brute~~ brute force:

try each perm of 1..8
check which ones work.



Doomed
to fail.



1
↓
1,2

```
for (int i = 0; i < n; i++) {  
    // Reg perm  
    if (used[i]) continue;  
    if (conflict(board, k, i)) continue;  
    // Backtracking step placing  
    // if doing so conflicts w/  
    // another queen
```

RECURSE

}

Sudoku

⚡ (4 is 1st item tried) 9x9 grid

3	2							
	1							
			7					

Given some #s

etc.

Fill in Rest.

Each Row 1..9

COL 1..9

BOX 1..9

When you place a number in "SLOT k"
 (0..80, $SLOT/9 \rightarrow row$, $SLOT \% 9 \rightarrow col$)
 Do not it if it's the 2nd copy of # in R,
 C or Box.

Magic Square

Program did this!

2	7	6
9	5	1
4	3	8

n by n Square
 Oops! Ints 1 through n^2 , exactly once.
 each row, col, diag (main)
 must add to same value.

we'll let
 the program do it!!

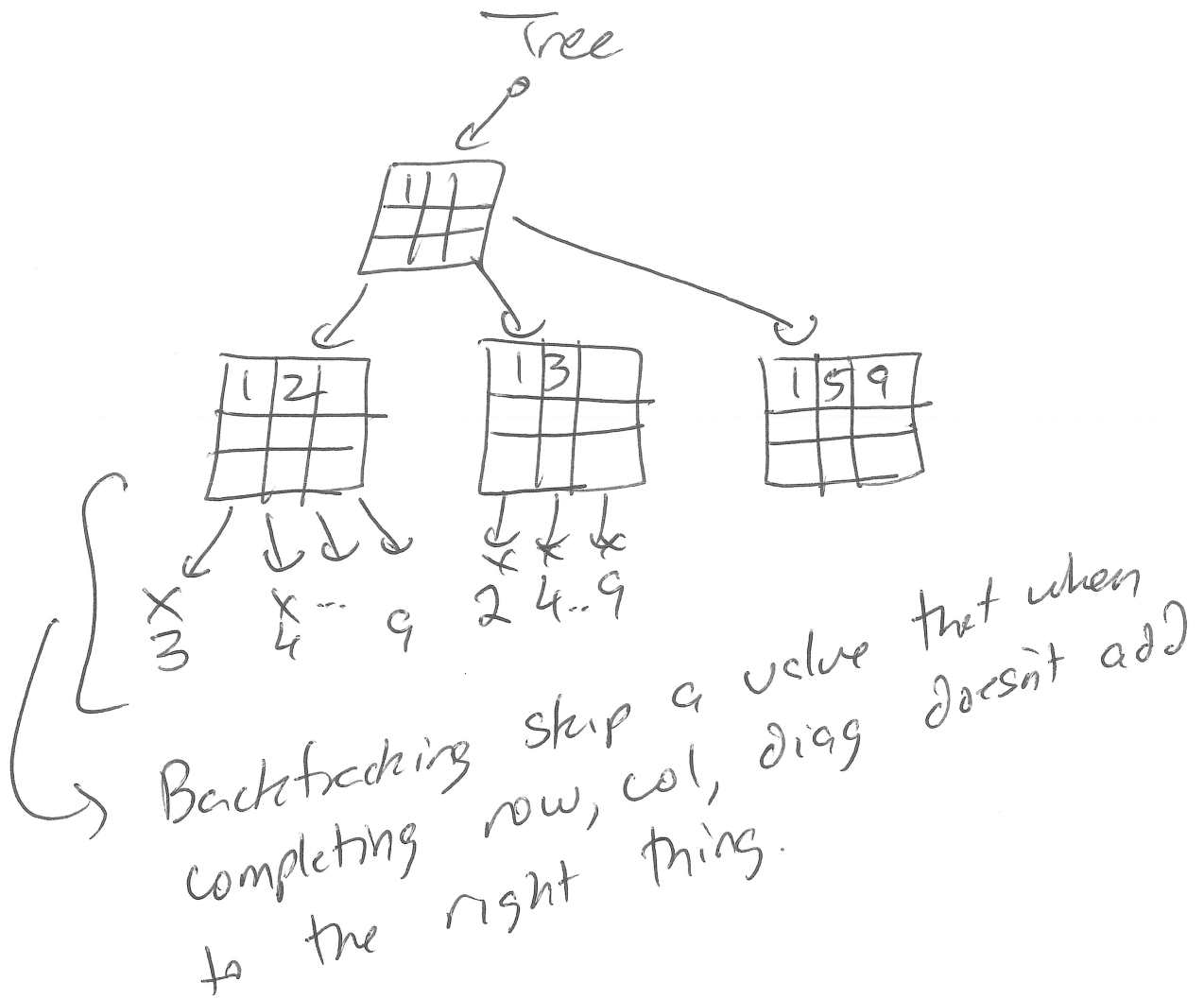
$$\text{Sum of } 1 \text{ to } n^2 = \frac{n^2(n^2+1)}{2}$$

$$n=3$$

$$\text{Sum} = \frac{3(3^2+1)}{2} = 15$$

Sum of one row :

$$\frac{n^2(n^2+1)}{2n} = \boxed{\frac{n(n^2+1)}{2}}$$



Magic Square Code

SLOW: all rows, cols, diags are checked EVERY TIME. But adding 1 number only affects 1 row, 1 col and maybe 1 or 2 diag.

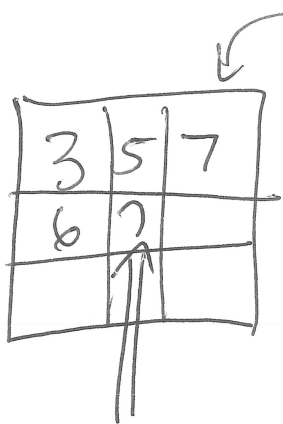
V2: MagicSquare2.java

IMPROVEMENTS: ONLY check MY row.

WRONG: TOO BIG (easy)

Completed but wrong sum.

Filled in all but one $\Rightarrow$ CALC LAST $\Rightarrow$ used or too big or 1 too small



3	5	7
6	2	

Consider 2 (2nd last slot)

$$6 + 2 = 8$$

$$15 - 8 = 7$$

what last slot would
need to be

→ BUT 7 is used!

So we skip
putting 2 there!