Rubik's Clock

(See http://www.rubiks.com/Games/)

Consists of 18 clocks

Front

Back (flipwise)

4 pegs, each of which can be up or down

i.e. 4 gears, each can be moved 1 tick.

A basic move consists of:

1. Configuring the pegs 16 ways

2. Selecting a gear
   - Turns one gear near up peg,
   - Same result

3. Moving the gear one gear near down
   - Peg always
   - Same result

Goal: Get each of the dials to read 12 o'clock
Ex of move

0 = down
1 = up

• any dial move clock 1-9 1 unit cw
  ccw choices 1, 3, 7, 9 1 unit ccw.

Record as a vector

\[ V_1 = \begin{bmatrix} 1 \\ 0 \\ -1 \\ 0 \\ 0 \end{bmatrix} \]

11 any dial move 1, 3, 7, 9 cw
11 all 10-18 ccw

\[ V_2 = \begin{bmatrix} 1 \\ 0 \\ 0 \\ 0 \end{bmatrix} \]
DO

moves 13 CW

cogwheel

all but 16 ccw

\[ \gamma_3 = \begin{pmatrix}
0 \\
0 \\
0 \\
0 \\
1 \\
1 \\
1 \\
0 \\
-1 \\
-1 \\
-1 \\
0 \\
0
\end{pmatrix} \]

moves 56 CW

89

ex

just 16 ccw on back

\[ \begin{pmatrix}
0 \\
0 \\
0 \\
1 \\
1 \\
1 \\
1 \\
0 \\
0 \\
0 \\
-1 \\
0
\end{pmatrix} \]
Turns out there are 30 different basic moves.

Form

\[ A = \begin{bmatrix} v_1 & \cdots & v_{30} \end{bmatrix} \]

30 \times 18

\[ b = \text{initial readings of clocks} \]

\[ x \text{ integer vector} = \begin{bmatrix} x_1 \\ \vdots \\ x_{30} \end{bmatrix} \]

\[ Ax + b = \text{reading of clocks after} \]

\[ \text{done x_1, if more} \]

\[ x_2 \text{ of move 2} \]

\[ \vdots \]

\[ x_{30} \text{ of move 30} \]

We want to find \( x \) so that

\[ Ax + b = 12k \]

for some vector \( k \).

I.e. we want to find integer vectors \( x, y \)

s.t.

\[ Ax = 12k - b \]
Find SNF of $A$.

$$UAV = S$$  where $$S = \begin{bmatrix} 1 & 0 \\ \vdots & \ddots & \ddots \\ 0 & \cdots & 1 \end{bmatrix}$$

$Ax = y \oplus b_i$ becomes

$$yAV \begin{bmatrix} x \\ y \end{bmatrix} = yw - Ub$$

$$y_i = \begin{cases} 0 & (ub)_i, (ub)_i, \\ \text{Can always solve} \end{cases}$$

$$y_{ii} = 12(ub)_i - (ub)_2$$

$$y = - (ub)_2 + \text{null of 12}$$

If

$$y = - (ub)_0 + \text{null of 12}$$

$$(ub)_2, (ub)_4, (ub)_7, (ub)_8$$

are null of 12

**Update:** If $b \begin{bmatrix} u_{i1} \\ u_{i2} \end{bmatrix} = b_3 + b_{16}$

or

$$12 + u_{16} \begin{bmatrix} 1 \\ b \end{bmatrix} = b_1 + b_{12}$$

then no sol.;

or

$$12 + u_{16} \begin{bmatrix} 1 \\ b \end{bmatrix} = b_9 + b_{16}$$

Otherwise $y = \begin{bmatrix} (ub)_1 \\ \vdots \\ (ub)_0 \end{bmatrix}$ is a solution.
\[ x = \sqrt{b_1} \]

\[ b_0, b_1 \]

\[ a_0, a_1 \]

\[ \text{So initial configuration can be solved if for each corner time on front + time on back is a multiple of 12.} \]