A - Word Encoding

Let the function *f(left,x,y)* be number of ways to extend a word ending in *xy* with *left* more letters.

$$f(left,x,y)=\begin{cases} 1 & left=0 \\ \sum_{c} f(left-1,y,c) & left>0 \end{cases}$$

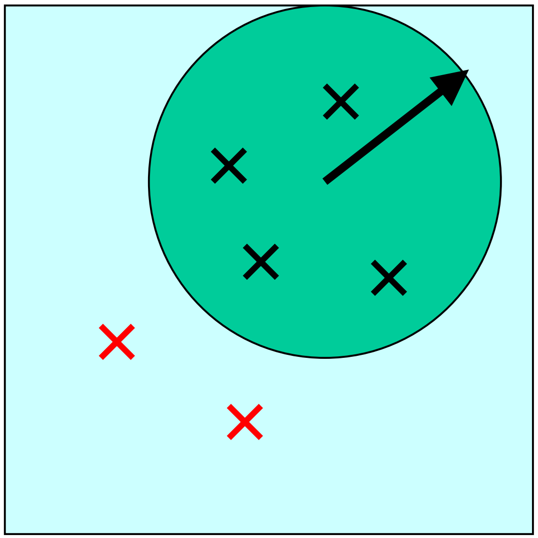
for all *c* so that *c*, *yc* and *xyc* are legal combinations.

Using this…
Determine the length of the word.
Start with an empty word, and check for each character whether skipping it would skip more words than desired.

The other direction can be implemented with a binary search.

B - Watchdog

Use brute force (≈ 1500 places to try on a 40x40 roof)



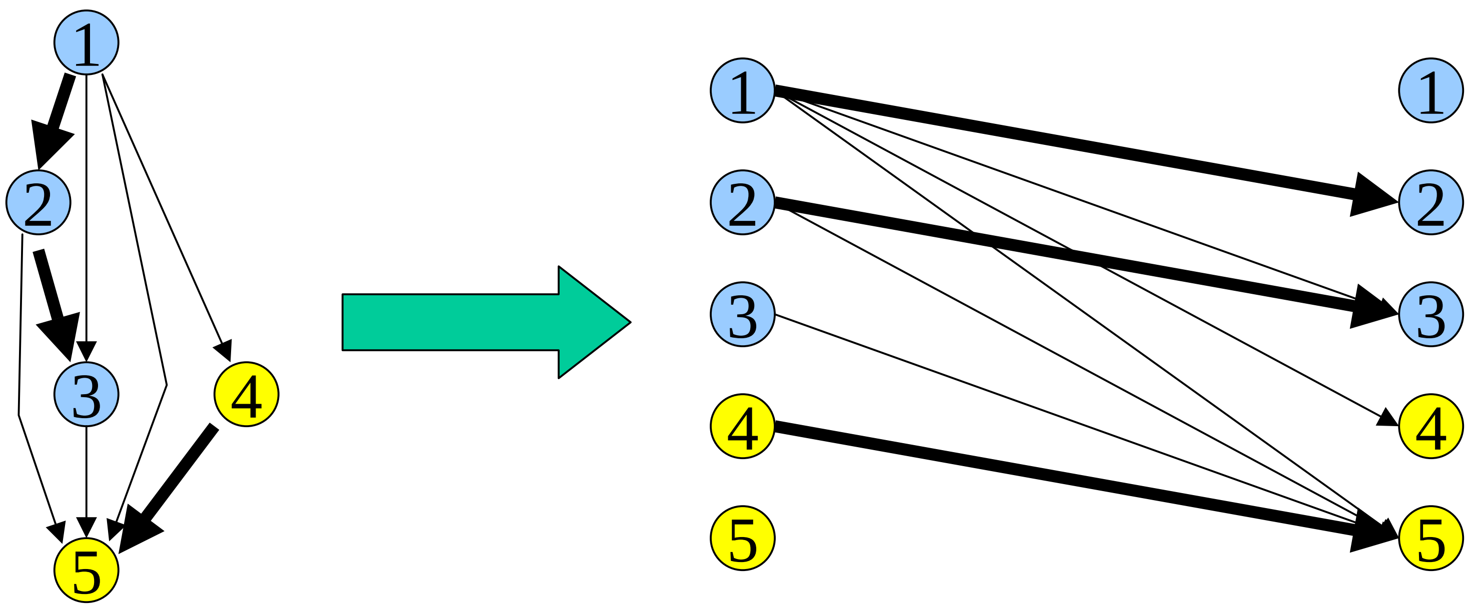
For each point (x,y) with integer coordinates:

Make leash as long as possible.

Check if all hatches are within reach.

C – Taxi Cab Scheme

Every taxi ride is represented by a node in a directed graph. There is an edge from *u* to *v* if and only if the same taxi can carry out the rides represented by *u* and *v* in that order.



Graph representing the rides

Corresponding matching problem

D - Pseudo-random Numbers

Step 1:

Calculate the seed backwards from the sequence.

5		2 3 4 5
9		5 7 9
6	$\Rightarrow$	2 1 6
7		3 7
0		0

Step 2:

Calculate forward from the seed, this is easy, in theory.
Problem: The numbers get very long.
Solution: Throw away everything but the last few digits.

E - Card Game Cheater

Brute force not possible (26! is to large)

Greedy strategy:

1. If Eve’s weakest card beats Adam’s weakest card, then pair up these cards.
2. Otherwise pair up Eve’s weakest card with Adam’s strongest.
3. Repeat until all cards have been paired.

F - Investment

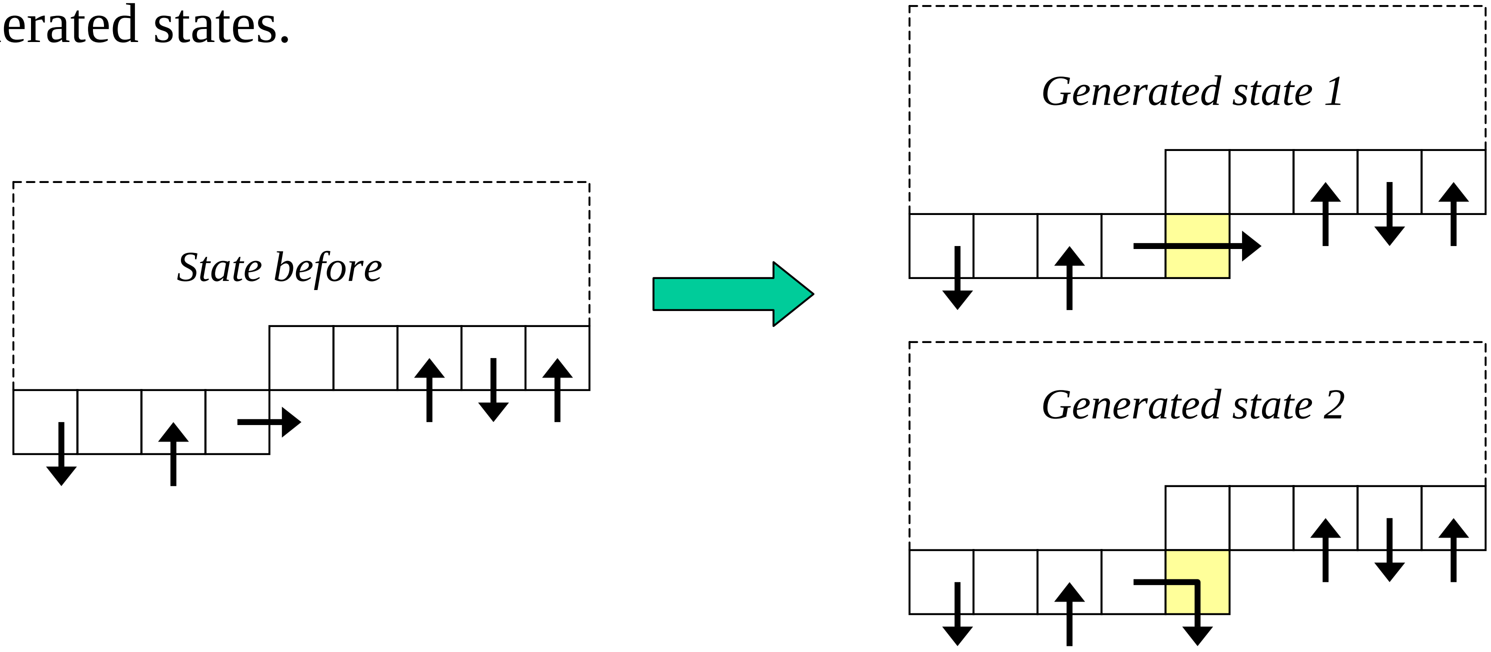
- The famous Knapsack problem!
- From the maximum start capital of € 1,000,000, the maximum interest 10% and the longest time of 40 years, an upper bound of ≈ € 46,000,000 can be calculated.
- For every multiple of € 1000 tabulate by dynamic programming the best possible interest.
- Iterate over the years and accumulate the interest to the amount.

G - Pipes

• Iterate over all rooms, from the upper-left to the lower-right in row order.

• Keep track of the optimal cost for all possible boundary states

• For each new room, loop over the possible boundary states. A boundary state can generate zero, one or two new states. Keep track of the optimal generated states.



H - SETI

• Solve the Vandermonde linear equation system by Gaussian elimination (it can be proved that there is always a unique solution).

• Note that since p is prime, all non-zero elements have a multiplicative inverse.

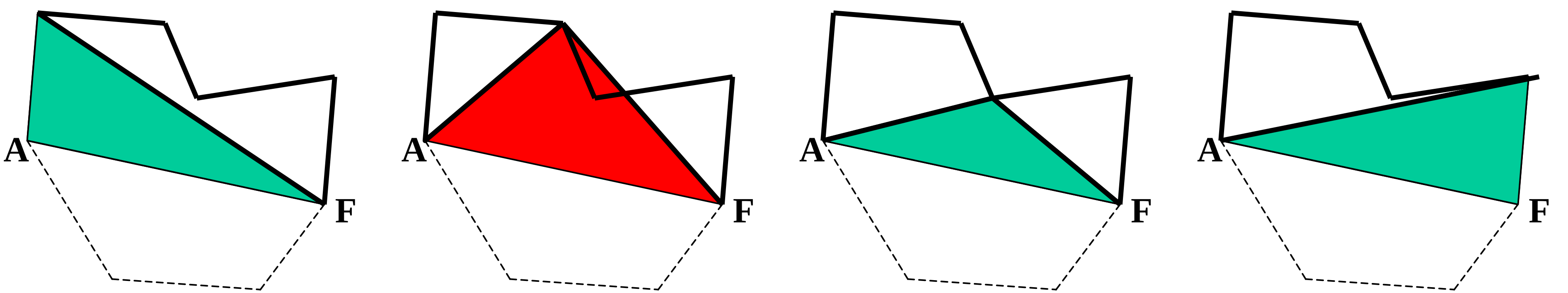
• Since p<30000 the inverse may be calculated by brute force (a faster solution may be achieved using Euclid's algorithm or using Fermat's (small) theorem).

$$\begin{pmatrix} 1^0 & 1^1 & & 1^{n-1} \\ 2^0 & 2^1 & & 2^{n-1} \\ & & \ddots & \vdots \\ n^0 & n^1 & \cdots & n^{n-1} \end{pmatrix} \cdot \begin{pmatrix} a_0 \\ a_1 \\ \vdots \\ a_{n-1} \end{pmatrix} \equiv \begin{pmatrix} f(1) \\ f(2) \\ \vdots \\ f(n) \end{pmatrix} \pmod{p}$$

I - Minimax Triangulation

• In order of increasing number of vertices, enumerate sub-polygons consisting of adjacent corners along the border.

• A new, larger sub-polygon is constructed by a triangle and two smaller (previously constructed) sub-polygons. Consider all possible constructions and select the optimal.



Example: The four possible triangulations of the A-F chord. The second construction is invalid. Tabulate the best valid sub-solution.