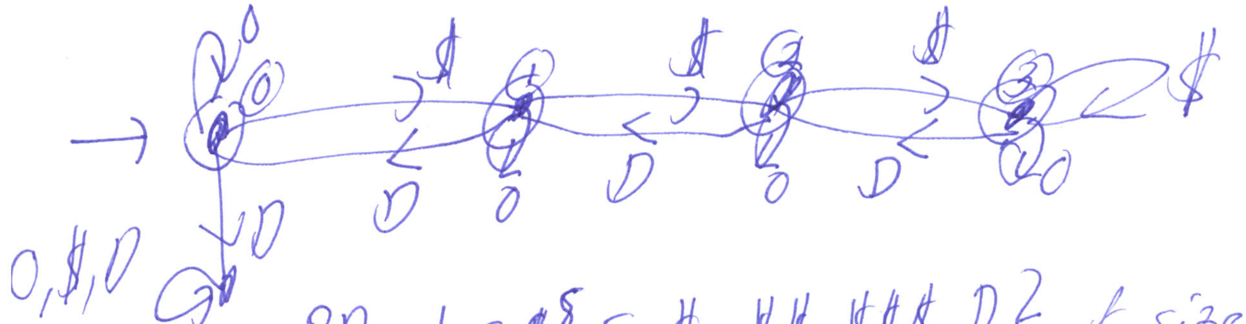
$L_K = \{ x \in \{\$, D, 0\}^* :$ x is a "survivable dungeon" when the player is allowed to hold up to K spears \$ }.

K = 3 label = # spears \$



PD set = $\{\epsilon, \$, \$\$, \$\$\$, D\}$ of size 5 for L_3.

- $D \notin L_3$, the other strings <u>are</u> in L_3, so $D \not\sim_{L_3} \epsilon$, ditto the others.
- Let any $m \neq n$, $0 \leq m < n \leq 3$ (without loss of generality) be given. Then $\$^m \not\sim_{L_3} \$^n$ because with $z = D^n$, $\$^m D^n \notin L_3$ since $m < n$, while $\$^n D^n \in L_3$.

Define $L_* = \{ x \in \{\$, D, 0\}^* :$ x is survivable with unlimited \$ saving }.
Same argument with "...<u>wlog</u> $0 \leq m < n < \infty$" shows that the <u>infinite</u> set
$S = \{ \$^i : i \geq 0 \} = \$^*$ is PD for L_*.

Note: S is not maximal — we could add "D" to it but this is enough.

Take $\Sigma = \{ (,) \}$ and $L = \{ x \in \Sigma^* :$ x can be "closed" into a balanced parentheses string }.

Example: $x = (()(\in L$ closed by $w =))$ But $x' = (()()))(\notin L$ because the marked right paren is overkill!

L is the same as L_* with $(= \$$, $) = D$, no 0. Infinite PD set: $S = (^*$.

Recall D = the doubleword language $\{uu : u \in \{0,1\}^*\}$.

$= \{uv : u = v,\ u, v \in \{0,1\}^*\}$

Some critical cases:

u	v
01	01
001	001
0001	0001

⚠ If $u = 000$ then nothing forces $v = 000$ too. $v = $ "0" is fine: $uv = 000 \cdot 0 = 00 \cdot 00 \in D$.

Tip: Anytime you have a compound expression like $u \cdot v$ on the LHS of set, rewrite in form $D = \{w : w$ can be broken as $w =: u \cdot v$ such that: $u = v,\ u, v \in \{0,1\}^*\}$

Hence the "1" is important. Let's define S to use it:

$$S = \{0^m 1 : m \geq 0\} = \underline{0^* 1}.$$

Or take $S = \{0^m 1 : m \geq 1\} = 0^+ 1$. still clearly infinite and "tighter."

S is PD: Let any $x, y \in S$, $x \neq y$ be given. Then there are $m, n \in \mathbb{N}$ with $m \neq n$ (we could say "wlog $m < n$" but we won't need it) such that $m, n \geq 1$ and $x = 0^m 1$, $y = 0^n 1$. Take $z = 0^m 1$. Then $xz = 0^m 1 0^m 1 \in D$ but $yz = 0^n 1 0^m 1 \notin D$ since $n \neq m$.

$\therefore$ S is PD for D and infinite, so D is non-regular by MNT. ☒

Can do PAL = {palindromes} similarly $S = 0^+ 1$ etc.

We could take $S = 0^*$ anyway: Let any $x = 0^m$, $y = 0^n$ be given with $m \neq n$. Take $z = \underline{10^m}$.

Then $xz = 0^m 1 0^m \in PAL$

but $yz = 0^n 1 0^m \notin PAL$ since $m \neq n$.

Since S is infinite, PAL is not regular. ☒

But $S = 0^+ 1$ or $S = 0^* 1$ are "proactive" an'l

One more example that starts regular but "blows up":

$L_K = \{ x \in \{0,1\}^* :$ the Kth bit from the right is a 1 $\} = (0 \cup 1)^* \cdot 1 \cdot (0 \cup 1)^{K-1}$

14 symbols counting · ; $K-1$ ≈ $\lceil \log_2 K \rceil$ symbols

$\approx 14 + \log_2 K = \Theta(\log K)$ chars

Strictly according to the regexp induction,

$r_K = (0 \cup 1)^* \cdot 1 \cdot (0 \cup 1) \cdot (0 \cup 1) \cdot (0 \cup 1) \cdots (0 \cup 1)$

9 symbols; $K-1$ times, giving $\approx 9 + (K-1)6 = \Theta(K)$ chars

NFA N_K =

arcs on 0 and 1 go to dead.

K−1 progressions using K states.

$K+1 = \Theta(K)$ states

$2K+1 = \Theta(K)$ instructions

How many states does a DFA M_K such that $L(M_K) = L_K$ need?

Define $S = \{0,1\}^K$. Then S is finite but $|S| = 2^K$.

Claim: S is PD for L_K, which $\Rightarrow$ M_K needs (at least) 2^K states!

Let any $x, y \in S$, $x \neq y$ be given. Then x and y differ as binary strings in at least one place $j \leq K$.

Take $z = 0^{j-1}$. For convenience, $j \geq 1$ — or vice-versa.

Say "x" refers to the string that has the 1 in place j, "y" to the string with 0 there.

Then $xz = x0^{j-1}$ now has that 1 in position K from the end, so $xz \in L_K$.

Whereas $yz = y0^{j-1}$ has a 0 in place K from end, so $yz \notin L_K$. ∴ S is PD.

By MNT, M_K needs (at least) 2^K states. $K = 270$ makes $|M_K| >$ |observed universe|

Brief Intro to Context Free Grammars

which can express many nonregular languages succinctly.

Examples in Advance.

1. $G = S \to 0S1 \mid \varepsilon$. Can produce

$S \Rightarrow \varepsilon$ directly.

$S \Rightarrow 0S1 \Rightarrow 01$

$S \Rightarrow 0S1 \Rightarrow 00S11 \Rightarrow 0011$ ($S \to \varepsilon$)

G generates the language $\{0^n 1^n : n \geq 0\}$.

2: $G = S \to \varepsilon \mid SS \mid (S)$ generates all balanced paren strings.

$G' = S \to \varepsilon \mid (S)S \mid (S)$ ditto.

3. $G = S \to \varepsilon \mid (S)S \mid (S$ generates all closable strings.

4. $G = S \to 0S0 \mid 1S1 \mid 0 \mid 1 \mid \varepsilon$ generates PAL.

¿Can we generate the doubleword language D??