# Termination

Tree Orderings

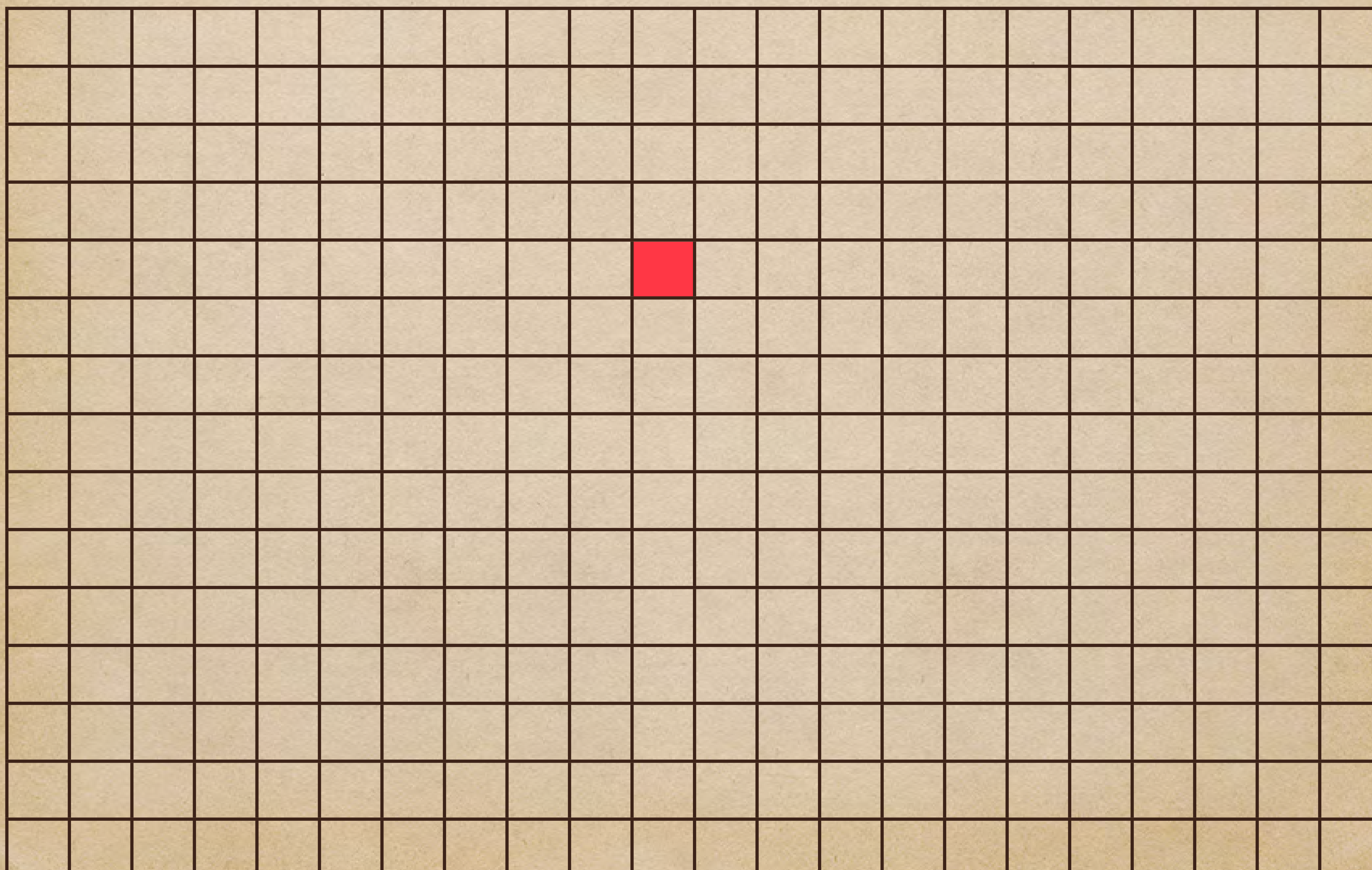
# Grid Game

- Given (upper-right) grid coordinates $(x0,y0)$
- Choose $(xj,yj)$ to prolong game s.t.
    - $xj < xi$ for all $i<j$ OR $yj < yi$ for all $i<j$

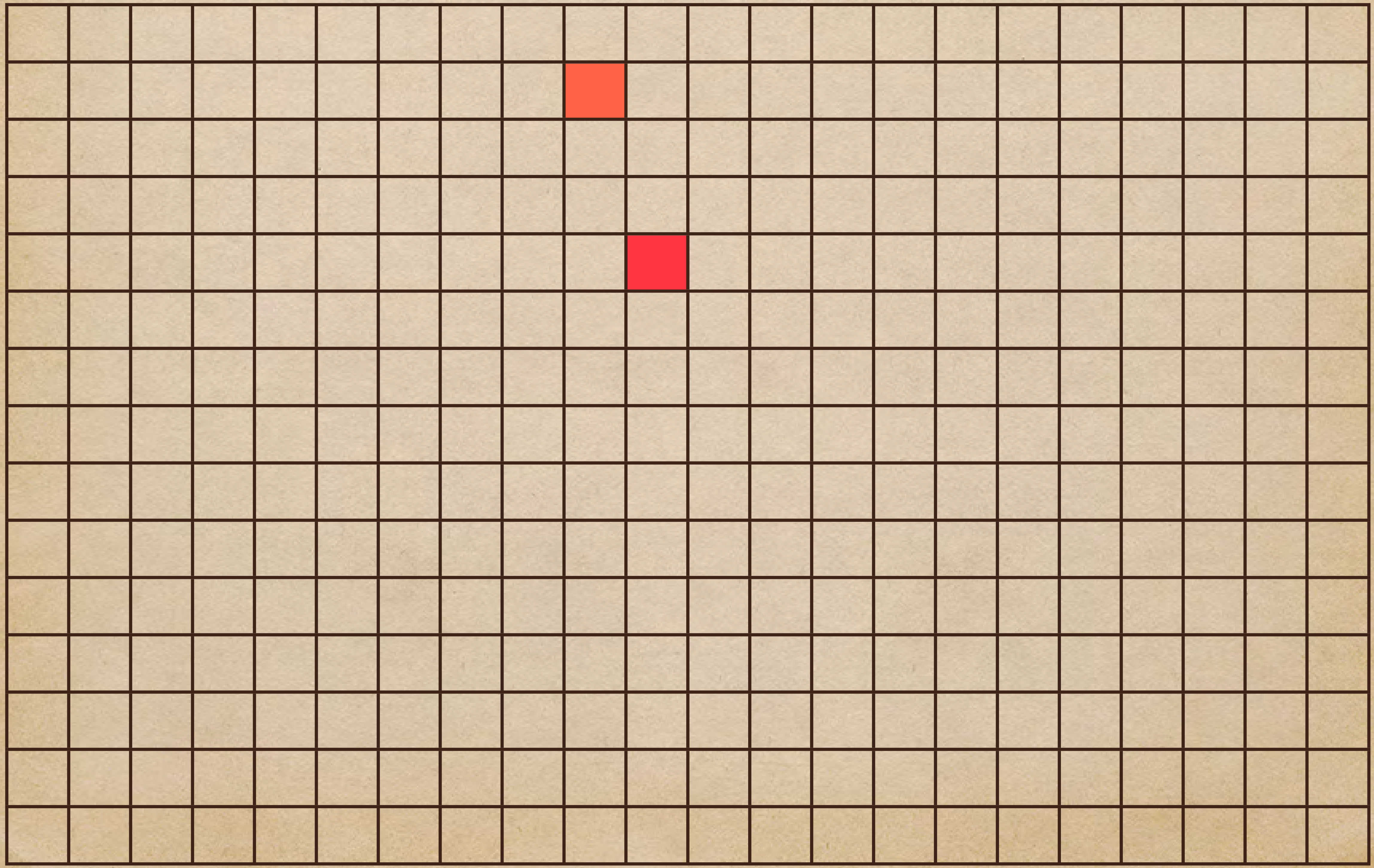
# Better Grid Game

- Given (upper-right) grid coordinates (x0,y0)
- Choose (xj,yj) to prolong game s.t.
  - xj < xi **OR** yj < yi     for all i<j

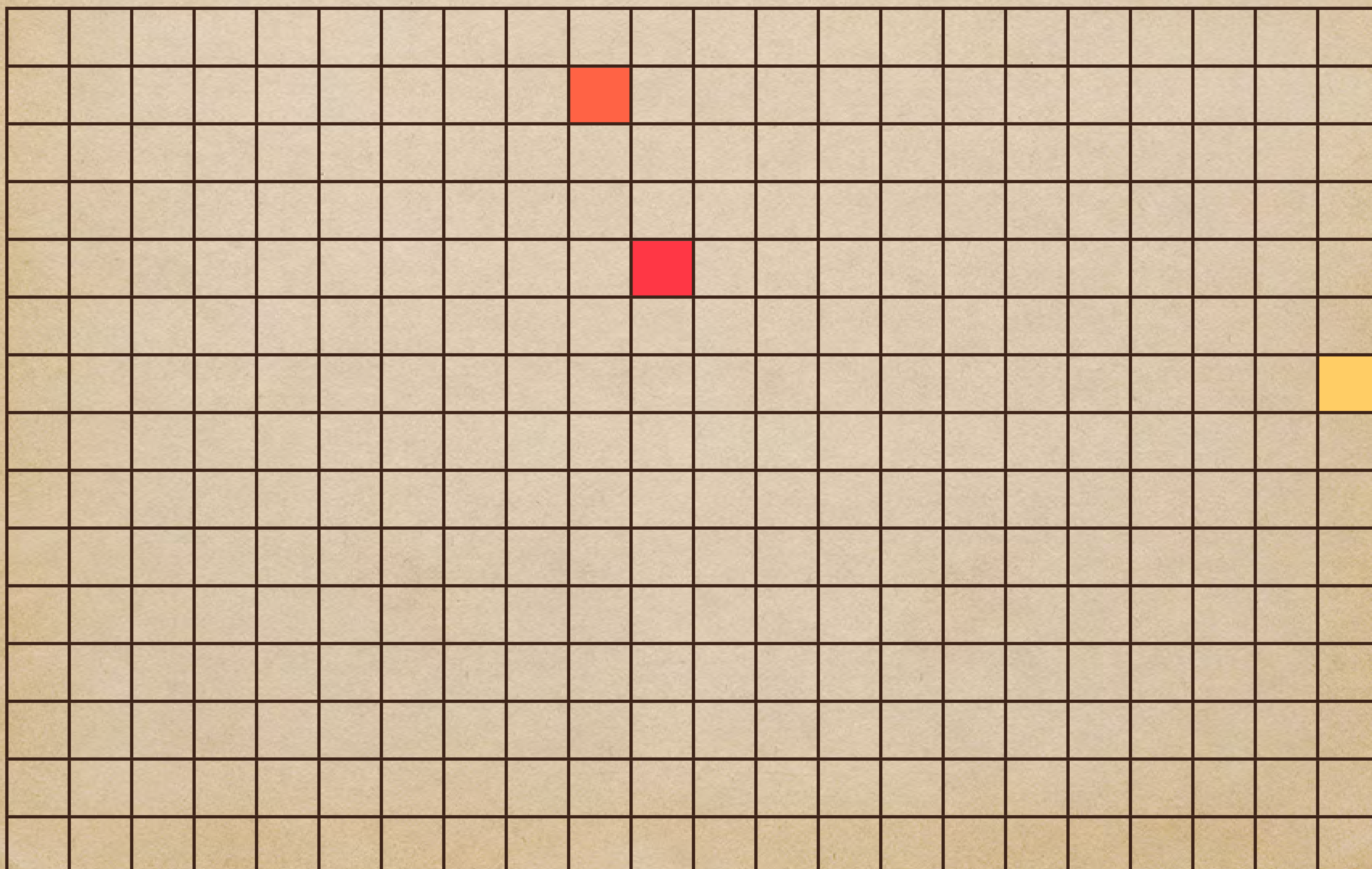
# Grid Game



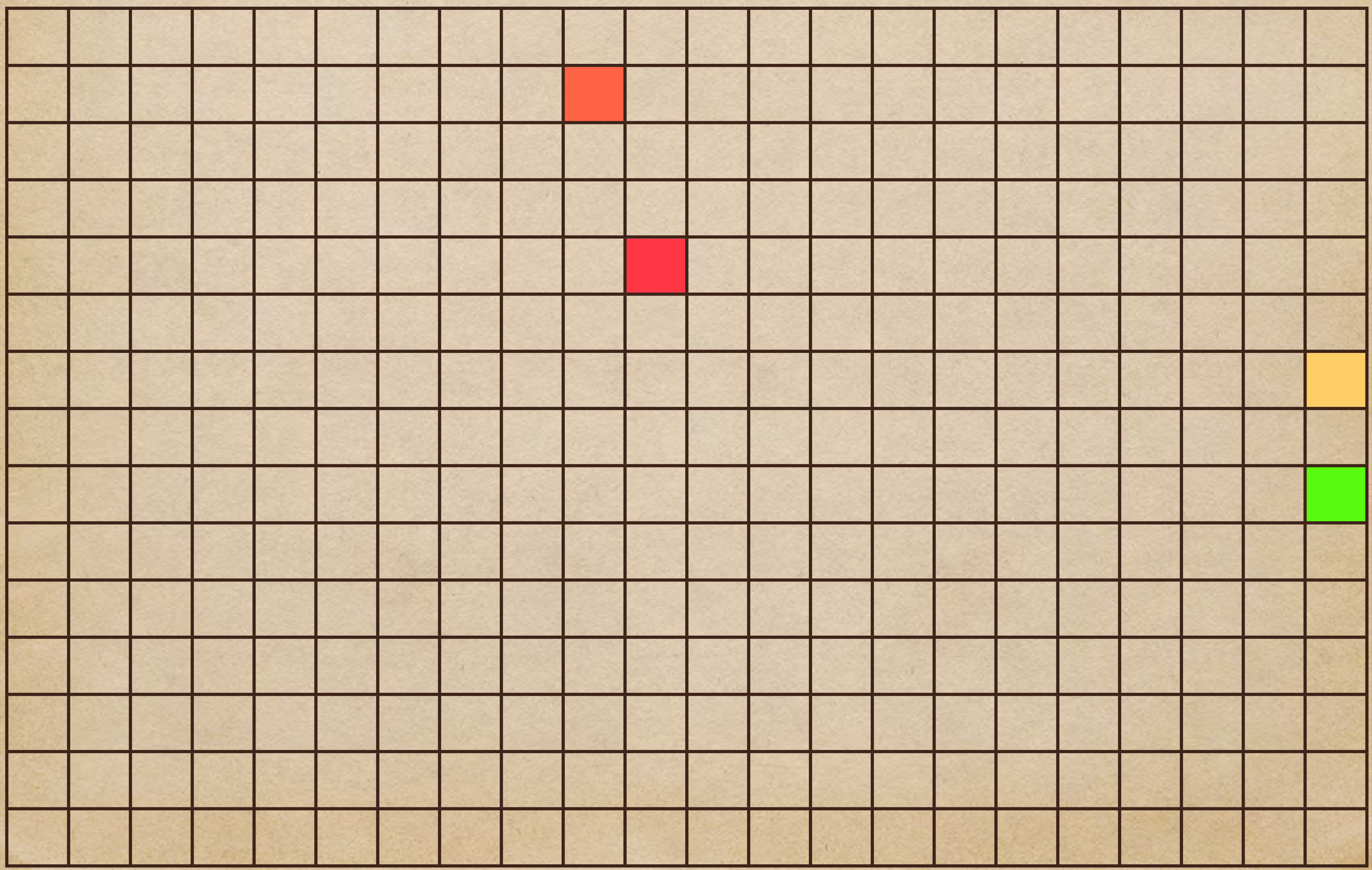
# Grid Game



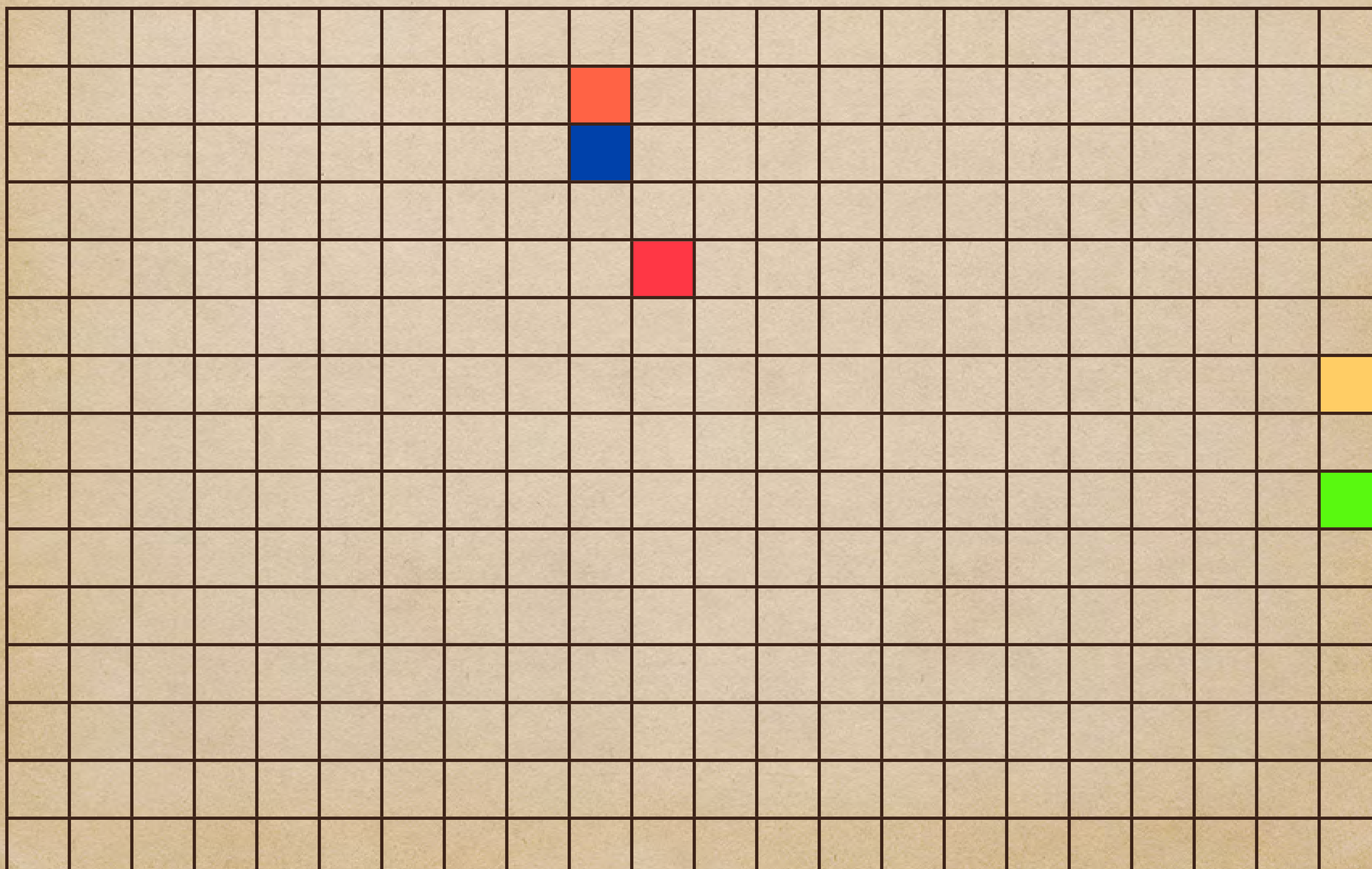
# Grid Game



# Grid Game



# Grid Game



# Tricolor

- Color pairs $i < j$ of points
  - Purple if $x_i > x_j$ and $y_i > y_j$
  - Blue if only $x_i > x_j$
  - Red if only $y_i > y_j$
- Consider sequence of points
  - Ramsey contradicts well-foundedness

# Symbolic Computation

- $Dt = 1$
- $Dc = 0$
- $D(x+y) = Dx + Dy$
- $D(xy) = xDy + yDx$
- ...

# Exponential Interpretation

- $[Dx] = 3^{[x]}$
- $[t] = [c] = 3$
- $[x+y] = \ldots = [xy] = [x] + [y]$

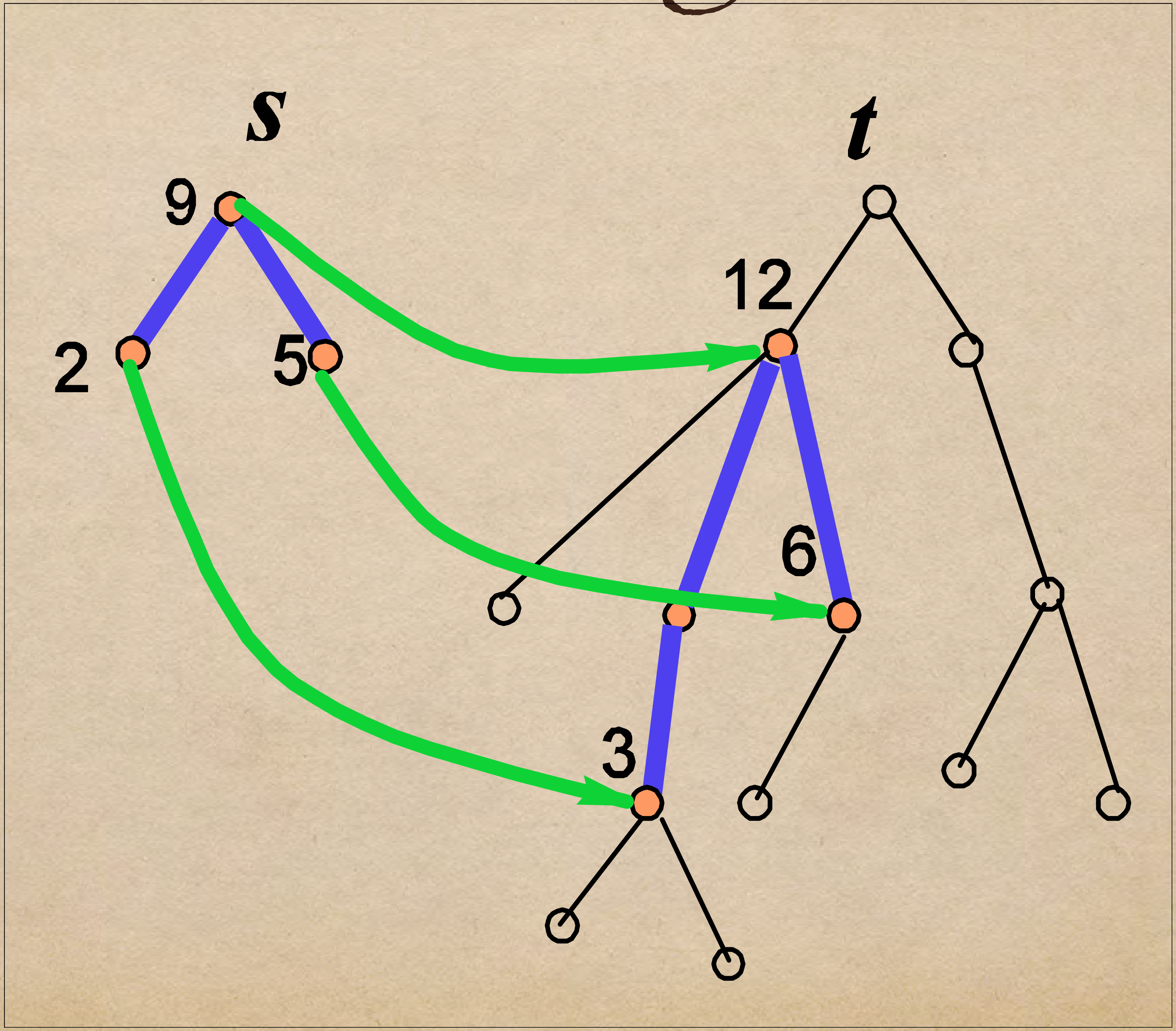
# WQO

- Standard: wf and no inf antichain
- Simple: Every infinite sequence has an ordered pair
- Useful: Every infinite sequence contains an infinite non-decreasing chain
  - Why? -- Ramsey

# Corollary

- Multiset ordering
- Bounded-arity tree ordering

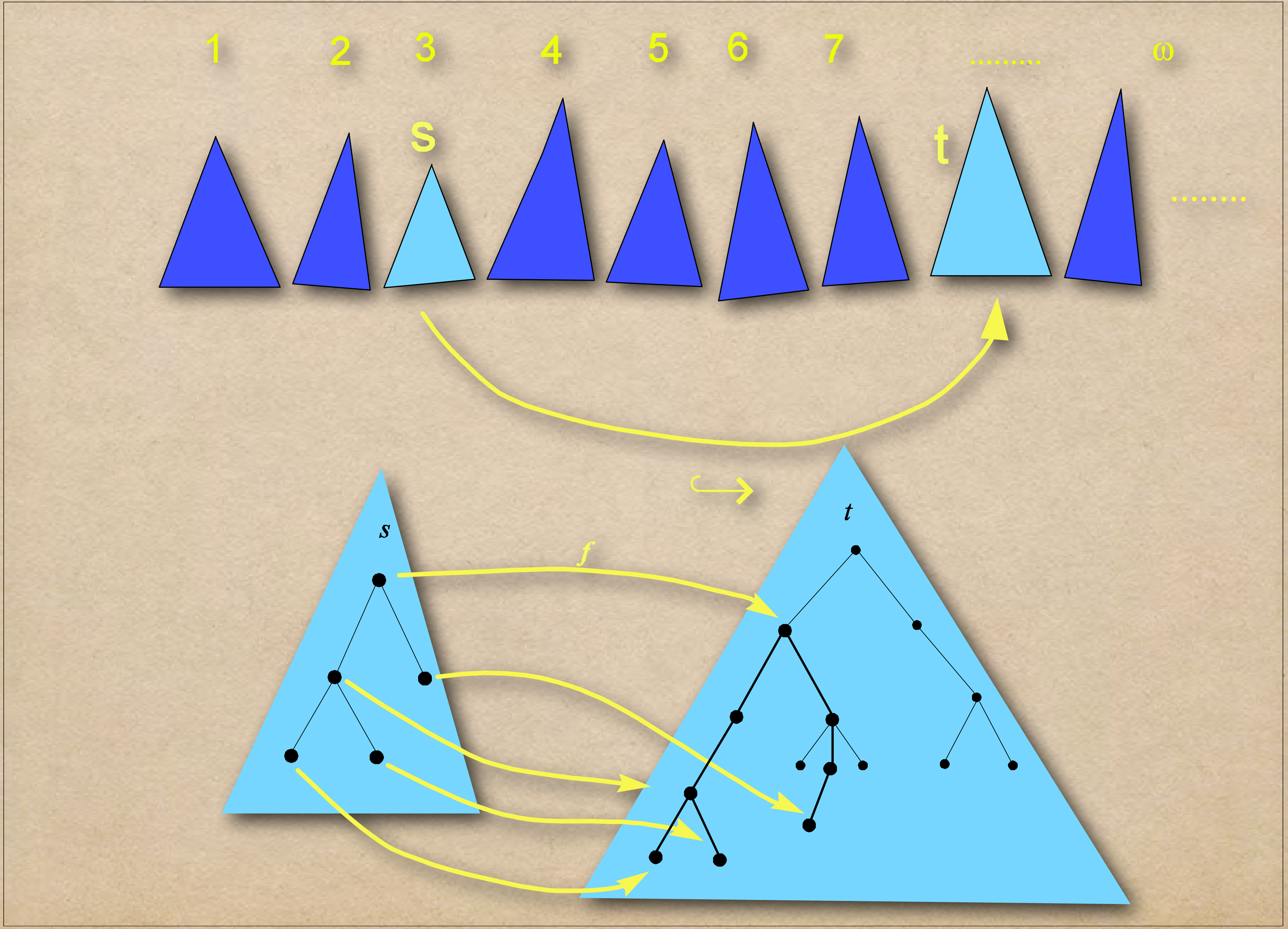
# Tree Embedding



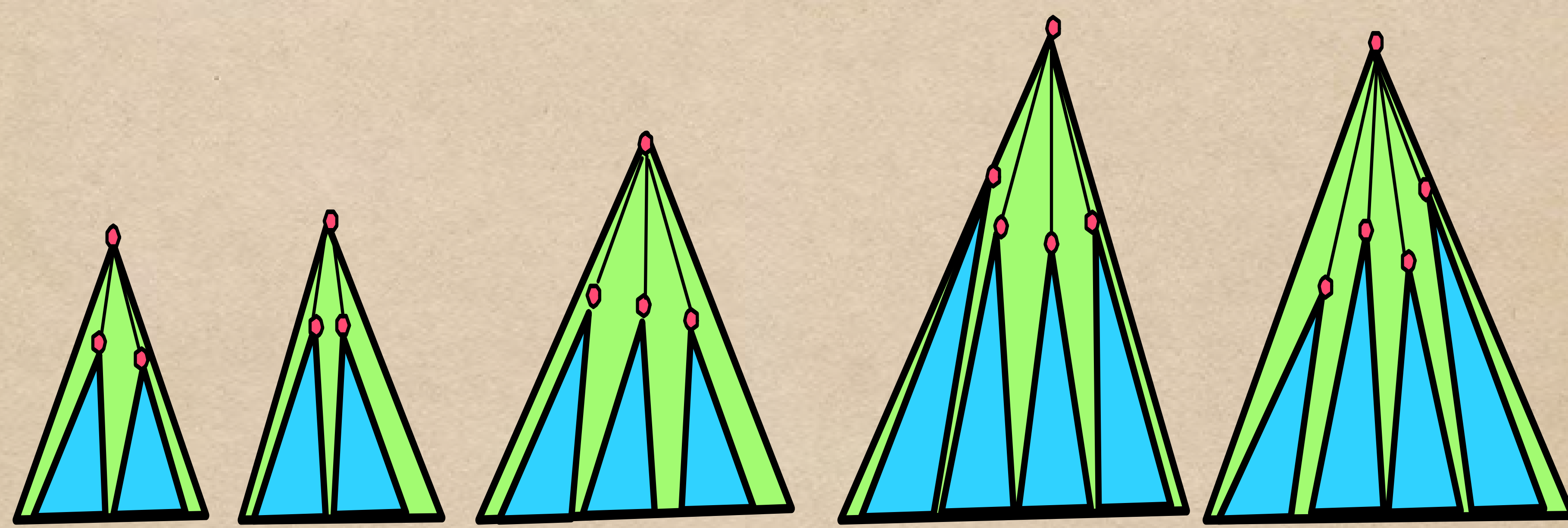
# Kruskal's Tree Theorem

- Every infinite sequence of trees (over a wqo alphabet) includes an embedding.

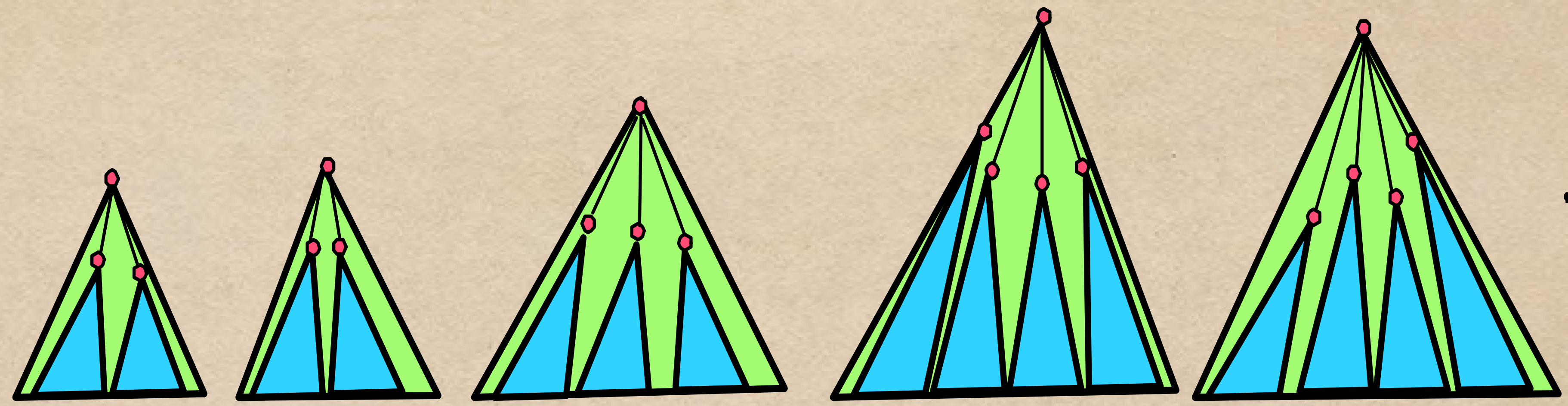
# Good Sequence

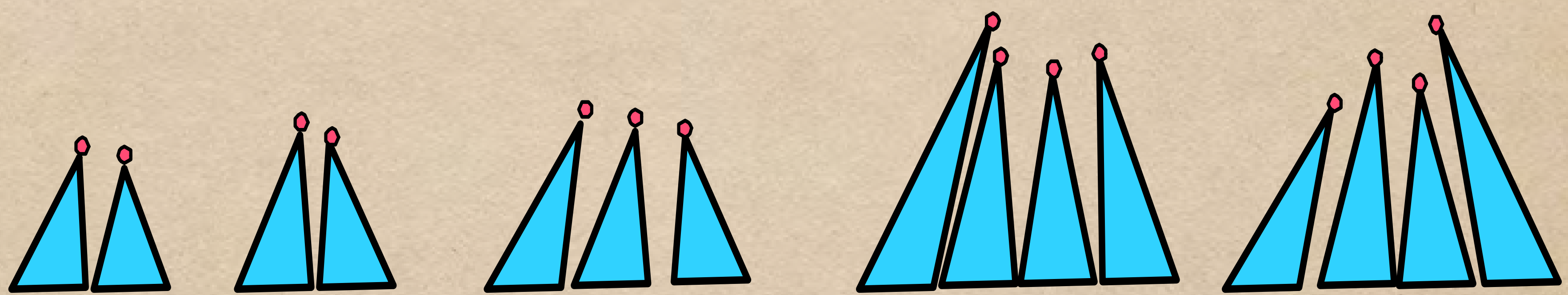


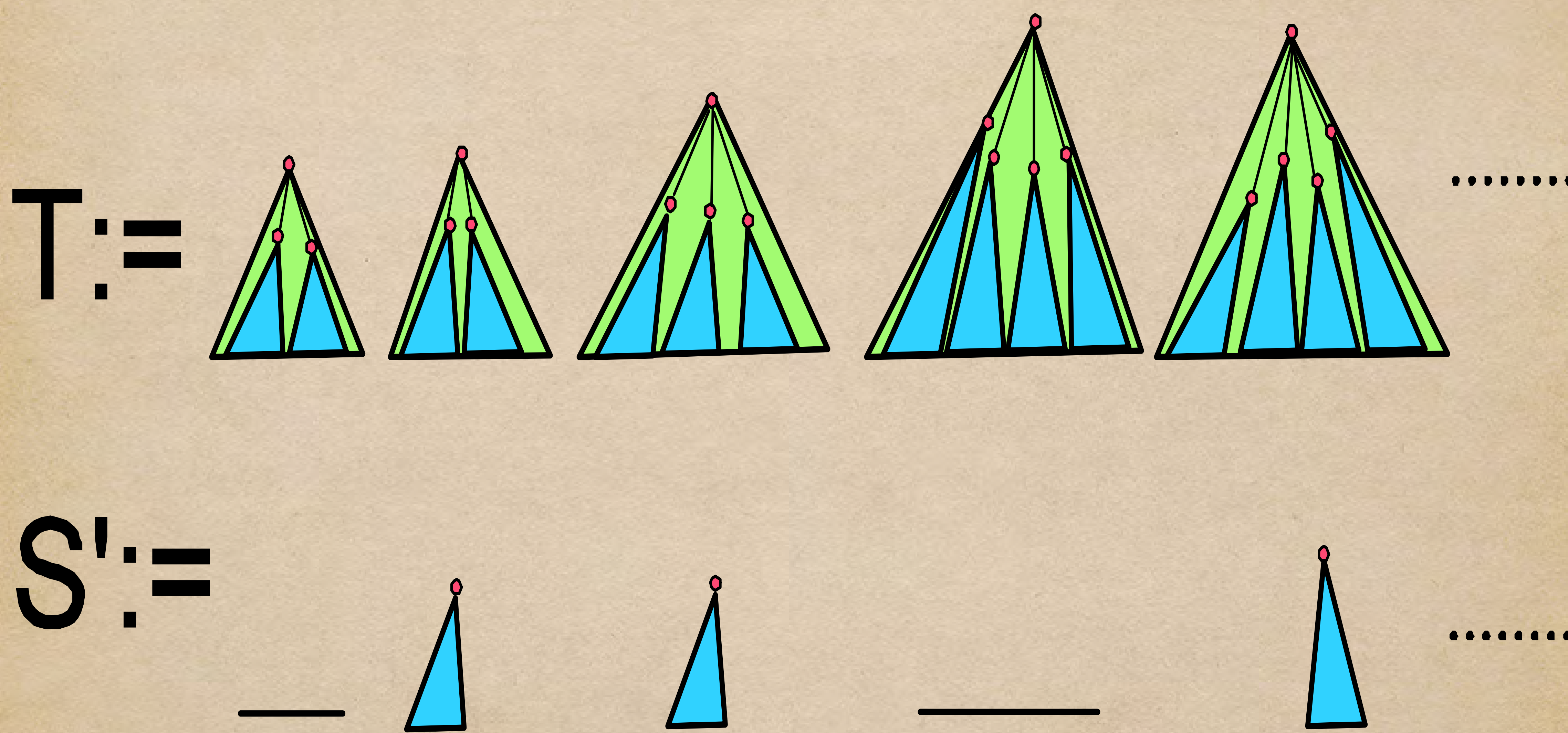
T:=

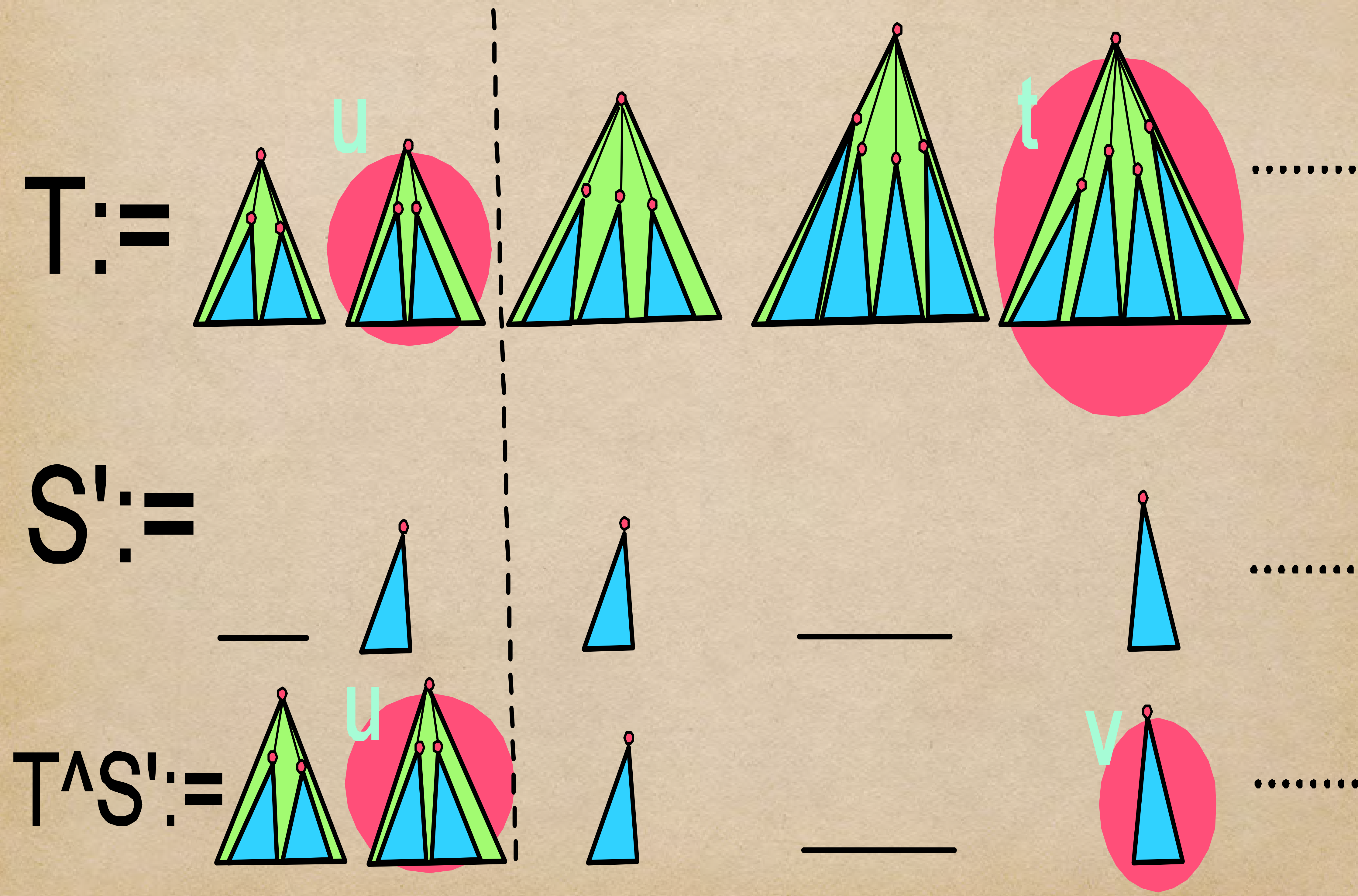


,,,,,,,

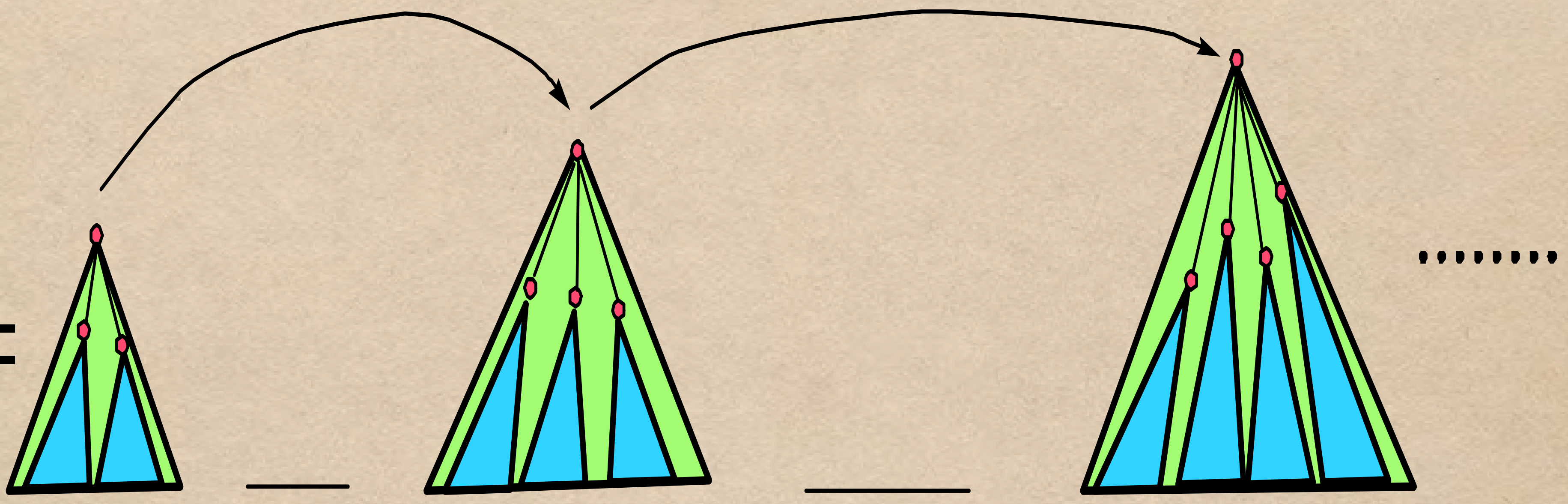
T:=  ........

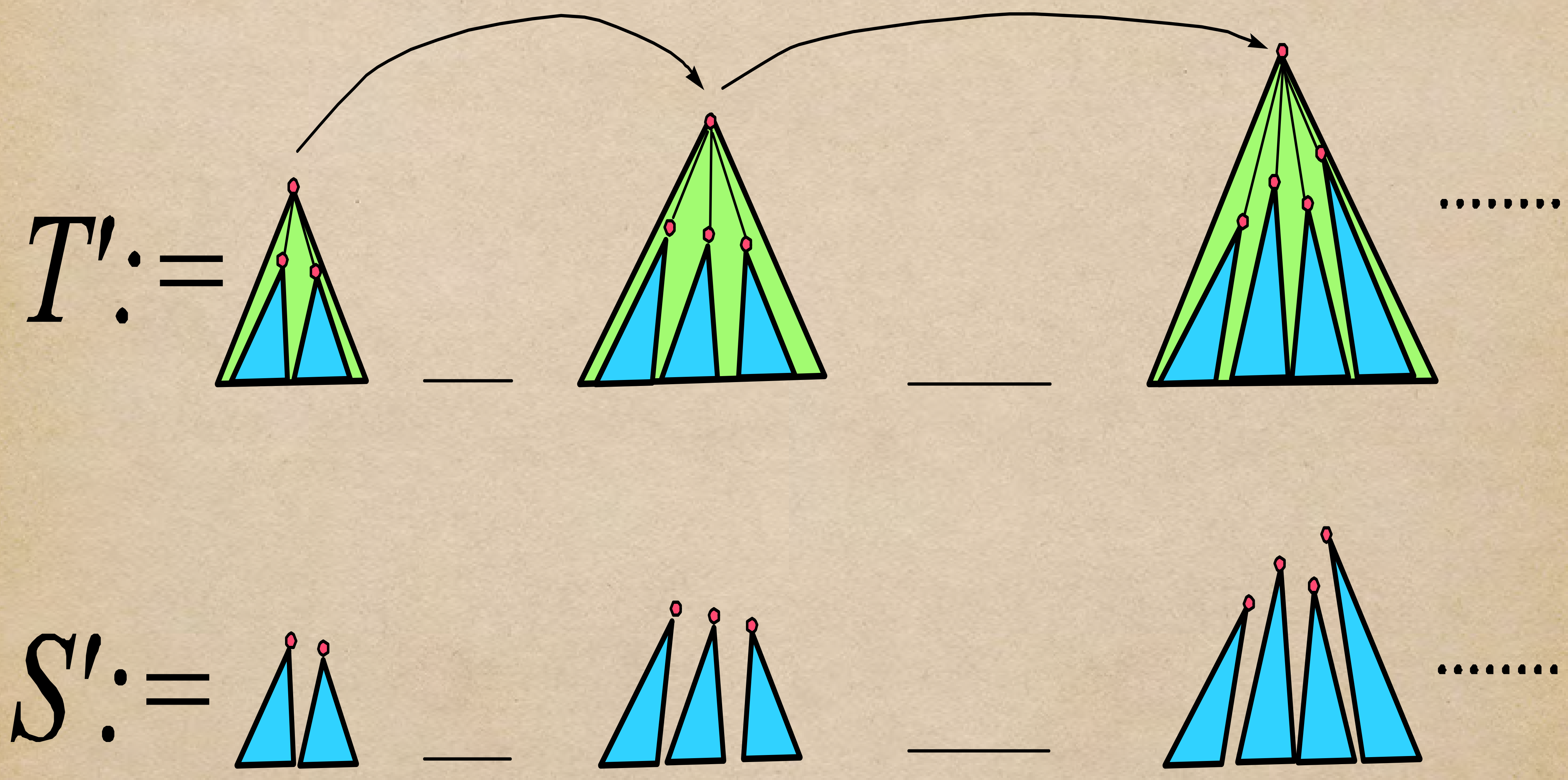
S:=  ........

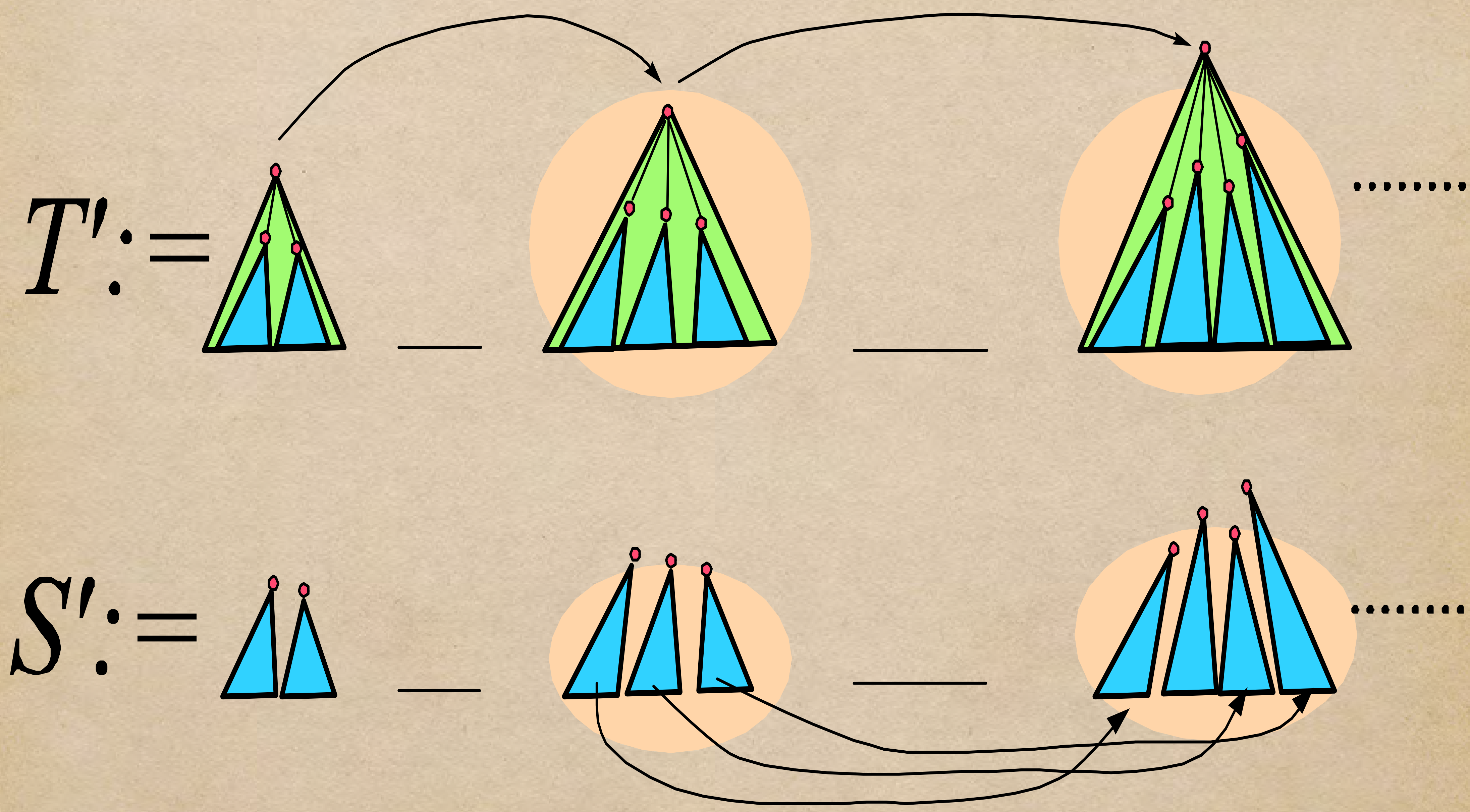

T:=
S':=



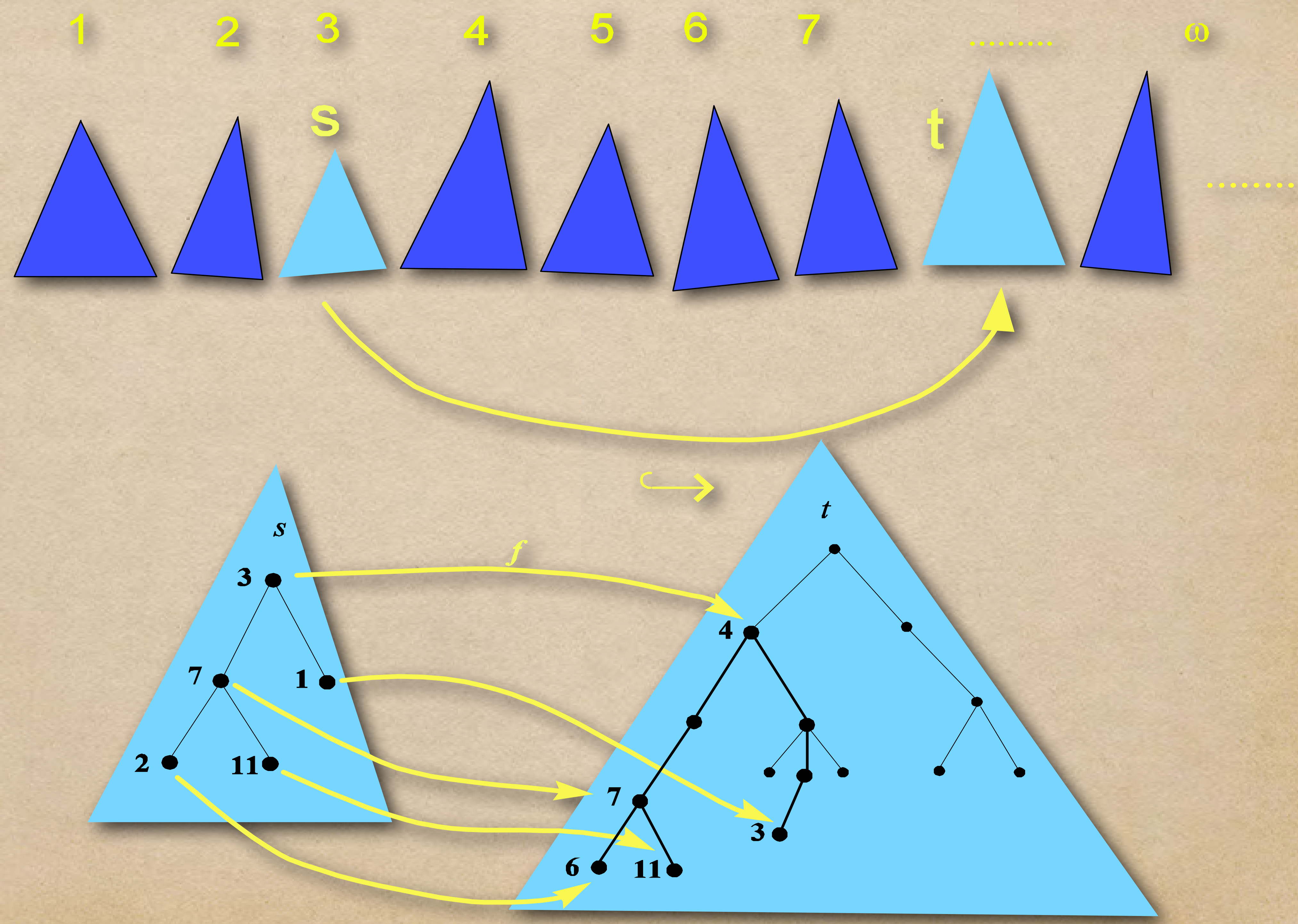
T:=
u
t
S':=
T^S':=
u
v

$T' :=$ 


T′:=
S′:=


T':=
S':=

# Labels



# Gremlins

# Gremlins

# Multiset Path Order

- $s = f(s_1,\ldots,s_m) \quad t = g(t_1,\ldots,t_n)$
- $s > t$ if $s_i \geq t$ for some $i$
- $s > t$ if
    - $(f,\{s_1,\ldots,s_m\}) >_{lex} (g,\{t_1,\ldots,t_n\})$
    - and $s > t_j$ for all $j$

# Symbolic Computation

- $Dt = 1$
- $Dc = 0$
- $D(x+y) = Dx + Dy$
- $D(xy) = xDy + yDx$
- ...

# Distributivity

- $x(y+z) = xy + xz$

# DNF

- $\neg\neg x = x$
- $\neg(x \vee y) = (\neg x) \wedge (\neg y)$
- $\neg(x \wedge y) = (\neg x) \vee (\neg y)$
- $x \wedge (y \vee z) = (x \wedge y) \vee (x \wedge z)$
- $(y \vee z) \wedge x = (y \wedge x) \vee (z \wedge x)$

# Simplification Order

- $f(\ldots,s_i,\ldots) > s_i$
- $s_i > t_i \Rightarrow f(\ldots,s_i,\ldots) > f(\ldots,t_i,\ldots)$
- Finite alphabet

# Simplification Order

- $f(\ldots,s_i,\ldots) > s_i$
- $s_i > t_i \Rightarrow f(\ldots,s_i,\ldots) > f(\ldots,t_i,\ldots)$
- $f > g \Rightarrow f(\ldots,s_i,\ldots) > g(\ldots,s_i,\ldots)$

# Lexicographic Path Order

- $s = f(s_1,\ldots,s_m) \quad t = g(t_1,\ldots,t_n)$
- $s > t$ if $s_i \gtrsim t$ for some $i$
- $s > t$ if
  - $(f,s_1,\ldots,s_m) >_{lex} (g,t_1,\ldots,t_n)$
  - and $s > t_j$ for all $j$

# Recursive Path Order

- $s = f(s_1,\ldots,s_m) \quad t = g(t_1,\ldots,t_n)$
- $s > t$ if $s_i \gtrsim t$ for some $i$
- $s > t$ if
  - $(f,s_1,\ldots,\{s_i,\ldots,s_m\}) >_{lex} (g,t_1,\ldots,\{t_i,\ldots,t_n\})$
  - and $s > t_j$ for all $j$

# Weak Simplification Order

- $f(\ldots,s_i,\ldots) \gtrsim s_i$
- $s_i \gtrsim t_i \Rightarrow f(\ldots,s_i,\ldots) \gtrsim f(\ldots,t_i,\ldots)$

# Simplification Ordering

- (Weakly) Monotonic
- (Weakly) Subterm

- They are well-quasi-orders