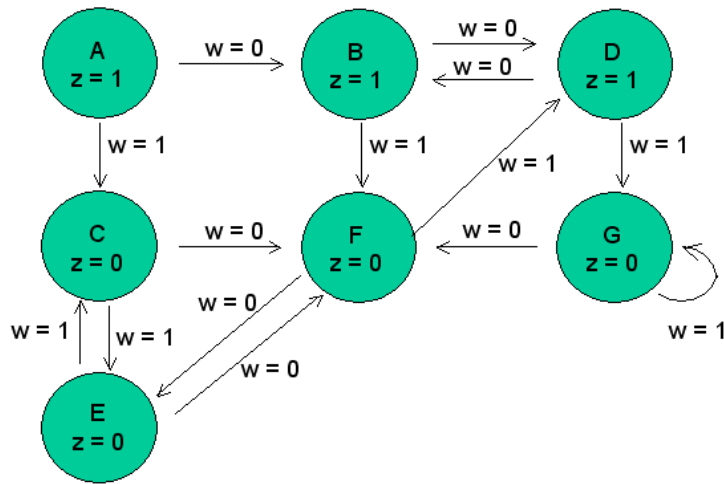


Tillståndsminimering – dagisversionen!

*Det går alltid att hitta det minimala
tillståndsdiagrammet. Vi presenterar här
en **enkel metod** som går att tillämpa på
små tillståndsdiagram.*

Tillståndstabell

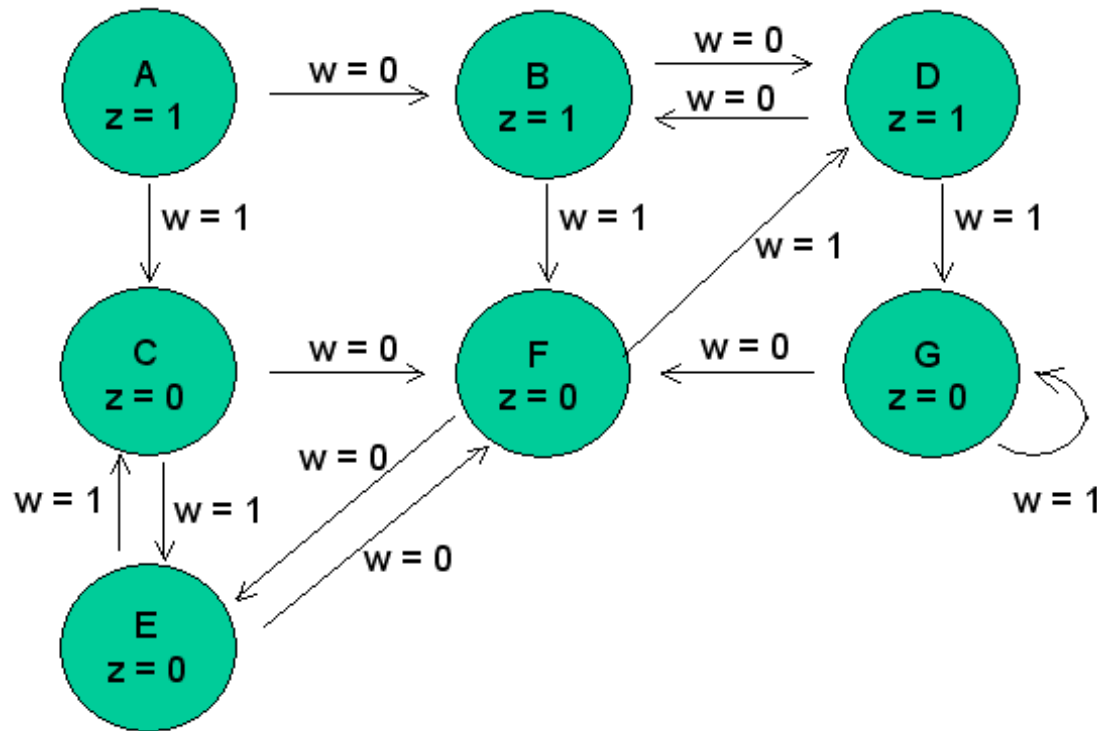


Ursprungligt tillståndsdigram

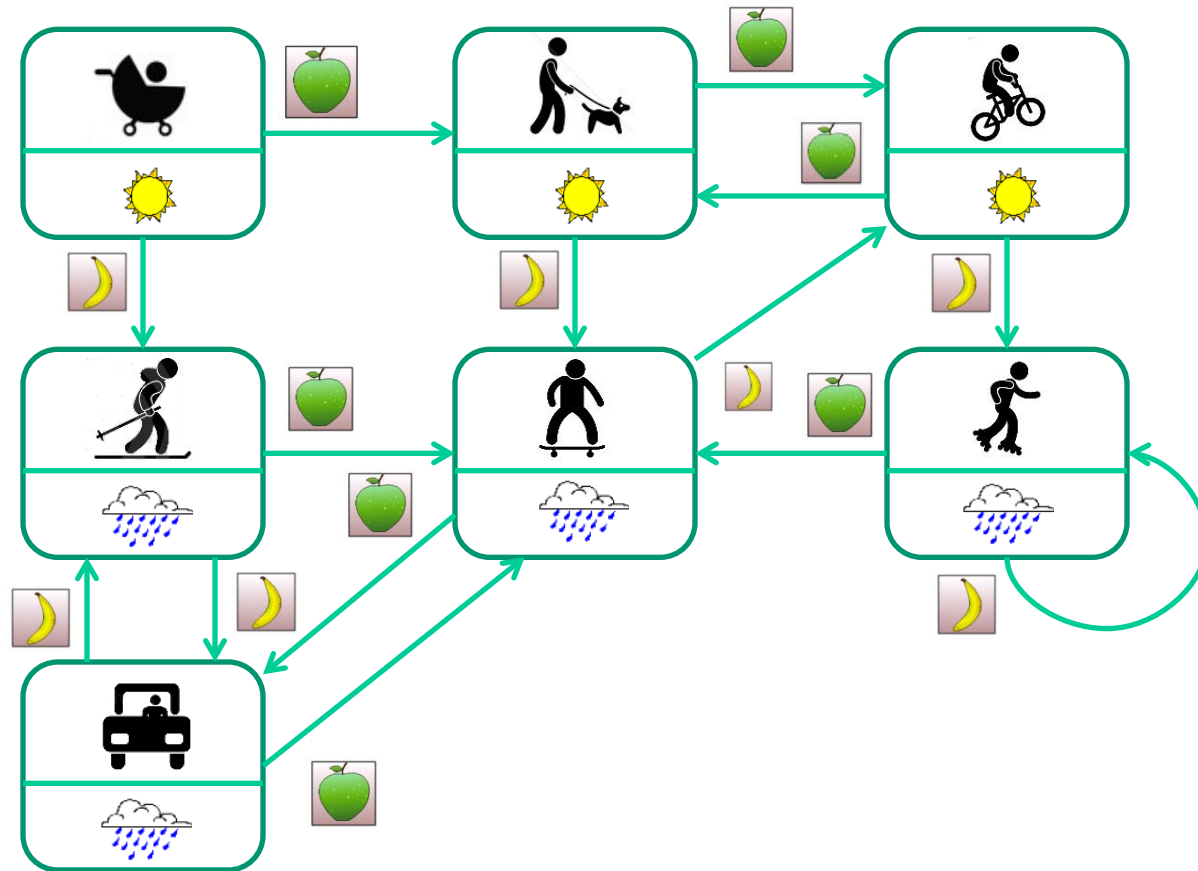
Present state	Next state		Output z
	$w = 0$	$w = 1$	
A	B	C	1
B	D	F	1
C	F	E	0
D	B	G	1
E	F	C	0
F	E	D	0
G	F	G	0

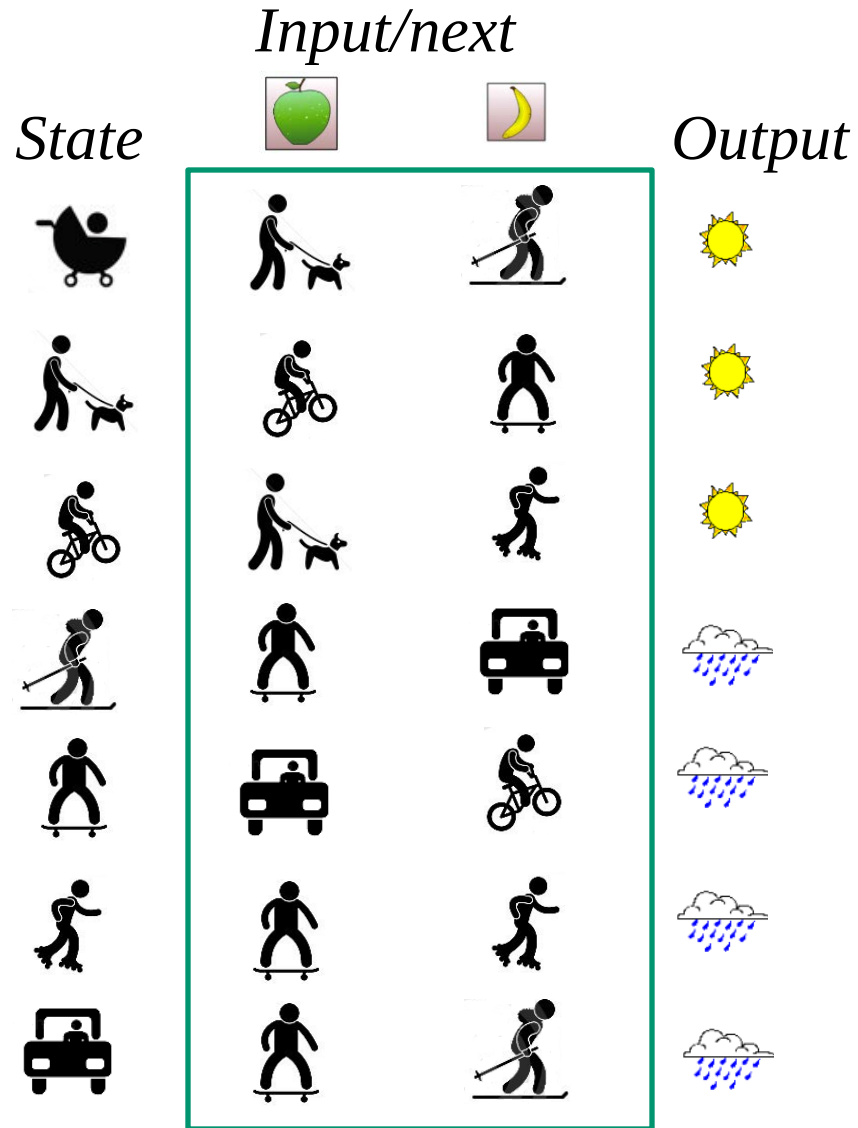
Ursprunglig tillståndstabell

Är tillståndsminimering svårt?

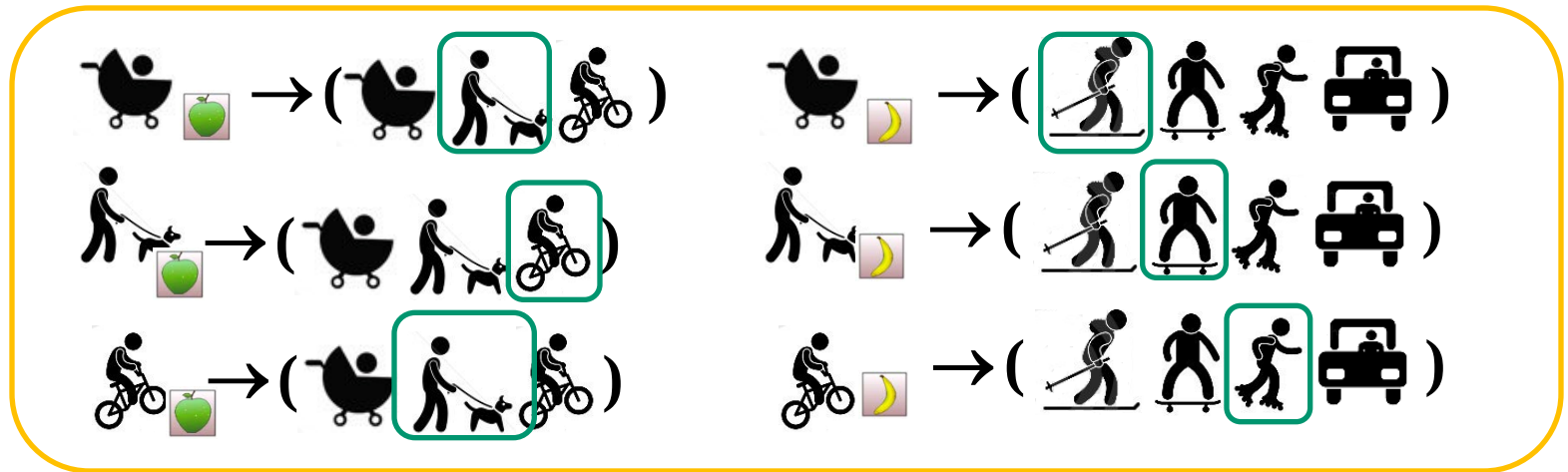
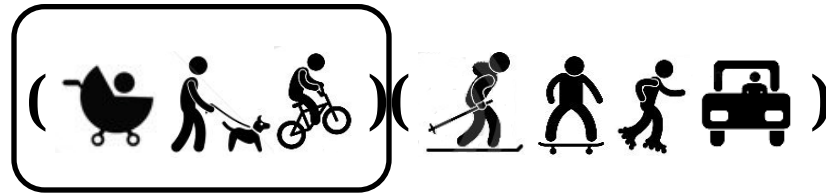


Dagisversionen ...

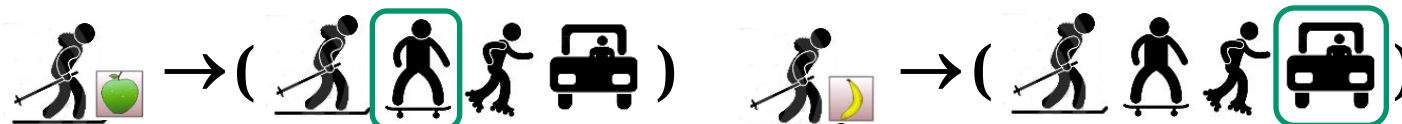
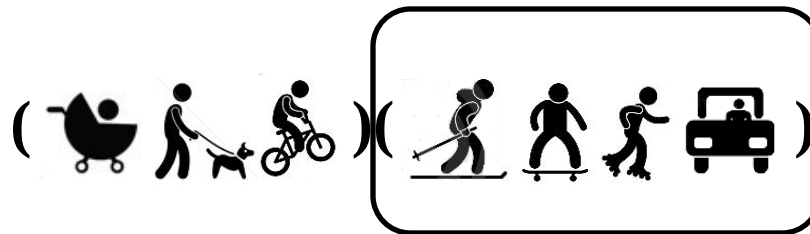




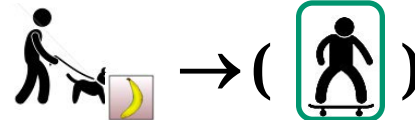
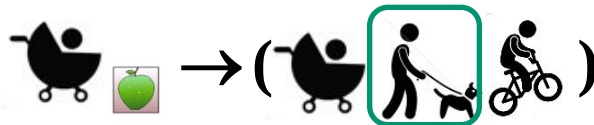
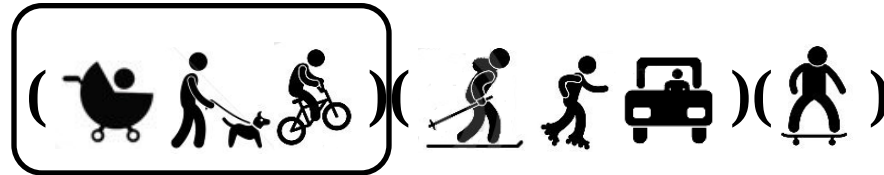
State	Input/next		Output
			
			
			
			
			
			
			
			



State	Input/next		Output



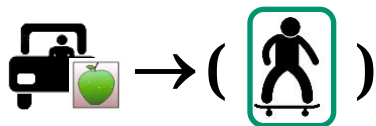
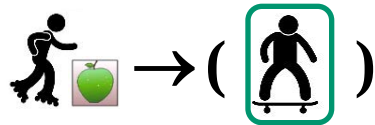
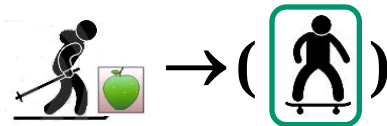
State	Input/next		Output

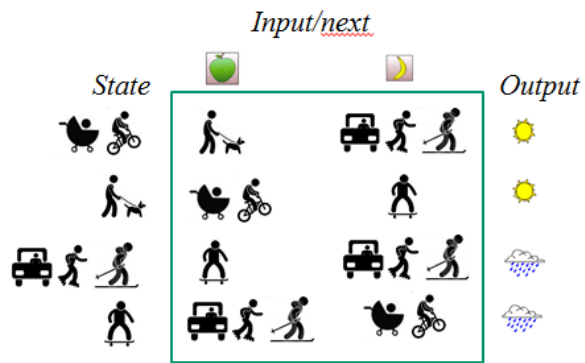


State	Input/next		Output

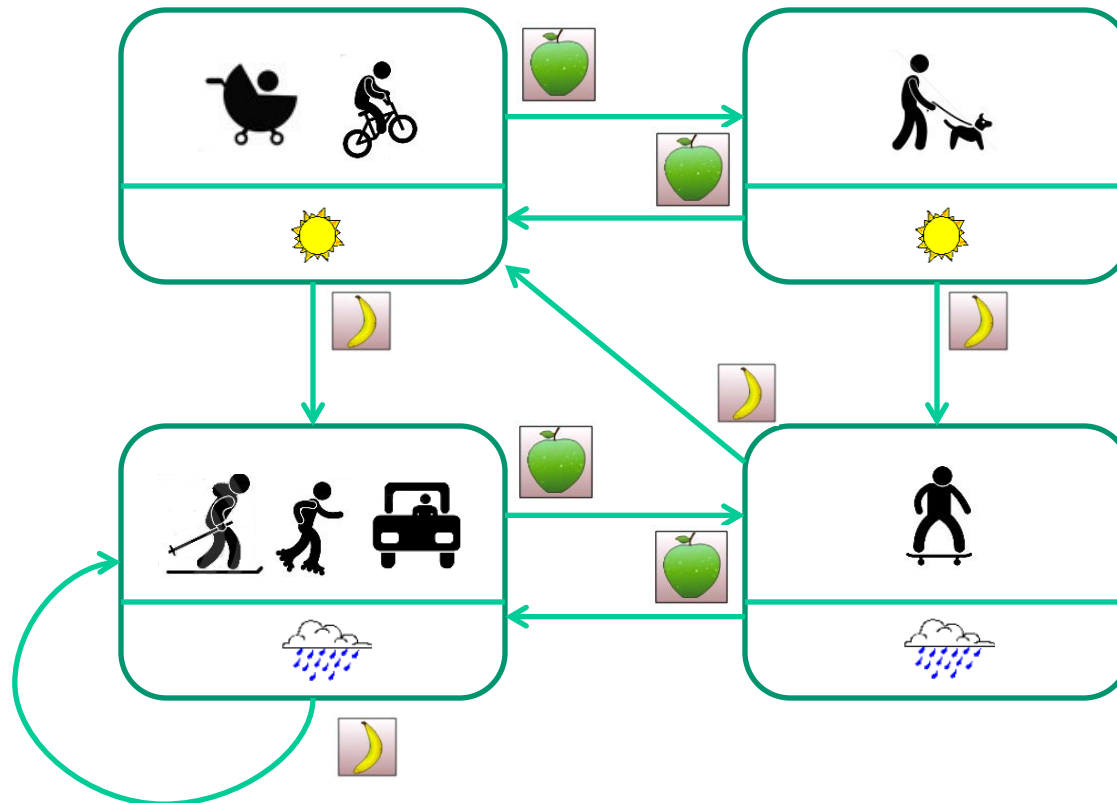


State	Input/next		Output



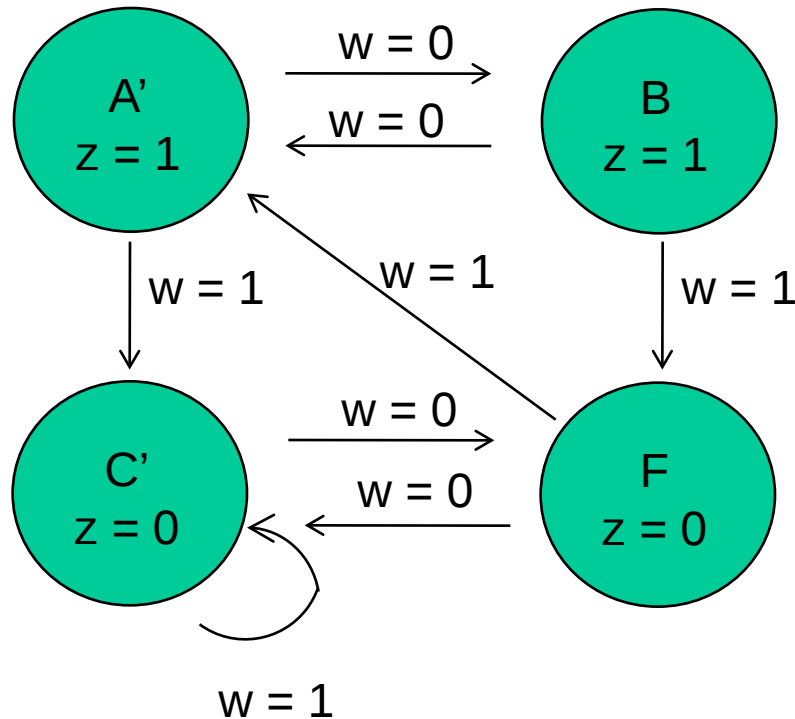


Minimalt antal tillstånd.



Minimerat tillståndsdigram

Present state	Nextstate		Output z
	w = 0	w = 1	
A'	B	C'	1
B	A'	F	1
C'	F	C'	0
F	C'	A'	0



4 tillstånd kräver 2 vippor ($2^2 = 4$).

Mer om tillståndsminimering kommer i ”Datastrukturer och Datoralgoritmer”

- Denna metod, (Moore 1956). Tillståndsminimering av n tillstånd – **körtid** på dator blir $\propto n^2$
- Hopcroft’s algoritm (1971) – **körtid** på dator $\propto n \cdot \log(n)$

Sifferexempel: 100 tillstånd.

$$100^2 = \mathbf{10000}$$

$$100 \cdot \log(100) = \mathbf{460}$$

*Datorprogram på internet
måste vara **skalbara**, de kan
”över en natt” bli mycket
populära – och riskerar då
att bli överbelasta servrarna!*

