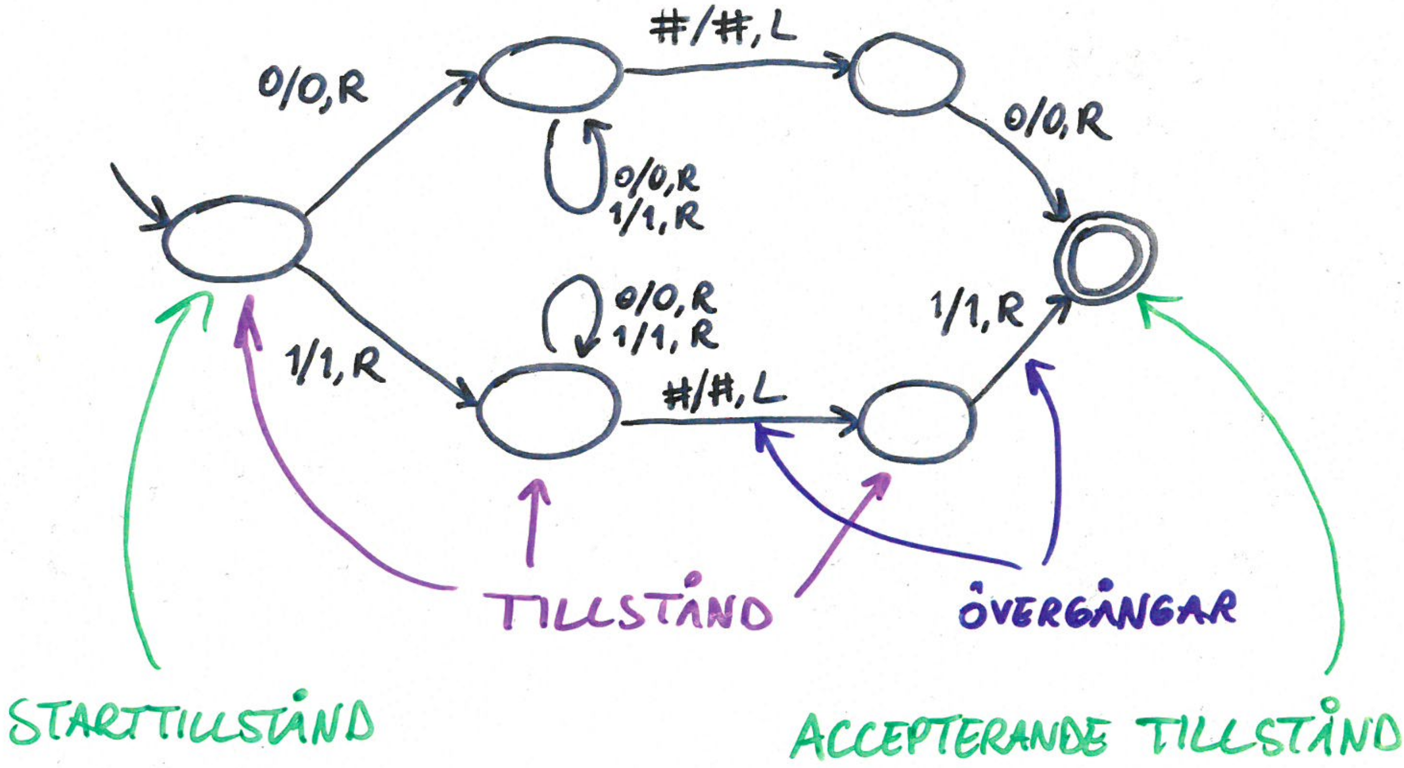
TURINGMASKIN

EXEMPEL: KOLLA OM DEN BINÄRA STRÄNGEN PÅ BANDET (INMATNINGEN) BÖRJAR OCH SLUTAR MED SAMMA SIFFRA



$\xrightarrow{a/b,L}$ BETYDER: OM LÄSHUVUDET LÄSER a, FÖLJ ÖVERGÅNGEN, SKRIV b OCH FLYTTA HUVUDET ETT STEG TILL VÄNSTER

$\xrightarrow{a/b,R}$ SAMMA SAK, MEN FLYTTA HUVUDET ETT STEG TILL HÖGER ISTÄLLET

REGLER FÖR TURINGMASKIN

- AUTOMATEN BÖRJAR ALLTID I STARTTILLSTÅNDET
- DÅ STÅR LÄS/SKRIVHUVUDET PÅ FÖRSTA SYMBOLEN I INDATA. INDATA OMGES AV BLANKA (TECKNAS #)
- OM TURINGMASKINEN ÄR DETERMINISTISK FÅR DET INTE FINNAS FLERA ÖVERGÅNGAR MED SAMMA LÄSSYMBOL FRÅN SAMMA TILLSTÅND.
- OM TURINGMASKINEN HAMNAR I ETT ACCEPTERANDE TILLSTÅND AVBRYTS BERÄKNINGEN OCH JA RETURNERAS.
- OM TURINGMASKINEN HAMNAR I ETT LÄGE DÄR INGEN MATCHANDE ÖVERGÅNG FINNS SÅ AVBRYTS BERÄKNINGEN OCH NEJ RETURNERAS.

ÖVERGÅNGAR:

$\xrightarrow{a/b,L}$ BETYDER: OM LÄSHUVUDET LÄSER a, SKRIV b OCH FLYTTA HUVUDET ETT STEG ÅT VÄNSTER.

L – ETT STEG ÅT VÄNSTER

R – ETT STEG ÅT HÖGER

S – FLYTTA INTE HUVUDET

CHURCHS TES

VARJE ALGORITMISKT PROBLEM SOM KAN LÖSAS MED NÅGOT PROGRAM SKRIVET I NÅGOT SPRÅK KÖRT PÅ NÅGON DATOR KAN OCKSÅ LÖSAS MED EN TURINGMASKIN.

FÖLJDSATS:

BERÄKNINGSBARHET ÄR ROBUST

FÖR BEVIS AV ÖVRE GRÄNSER:

ANVÄND KRAFTFULLT PROGRAMSPRÅK

FÖR BEVIS AV UNDRE GRÄNSER:

ANVÄND TURINGMASKINEN

SPRÅK OCH BESLUTSPROBLEM

ETT **FORMELLT SPRÅK** ÄR EN MÄNGD STRÄNGAR.

EXEMPEL:

$\{xy, yxx, xyzzy, zxy\}$

$\{2, 3, 5, 7, 11, 13, 17, 19, 23, 29, \ldots\}$

{ SYNTAKTISKT KORREKTA C-PROGRAM }

{ SATISFIERBARA BOOLESKA FORMLER }

OLIKA SÄTT ATT BESKRIVA SPRÅK:

- RÄKNA UPP STRÄNGARNA I SPRÅKET
- EN GRAMMATIK – REGLER SOM DEFINIERAR SPRÅKET
- EN ALGORITM SOM KÄNNER IGEN STRÄNGARNA I SPRÅKET, DVS $A(x)=1$ OMM X TILLHÖR SPRÅKET

VARJE BESLUTSPROBLEM MOTSVARAR ETT SPRÅK!

NÄMLIGEN SPRÅKET SOM BESTÅR AV ALLA JA-INSTANSER.

DEFINITIONER AV P OCH NP

$P = \{Q : \exists$ EN TM SOM KÄNNER IGEN Q I POL. TID$\}$

EN TURINGMASKIN A VERIFIERAR INSTANSEN X TILL PROBLEMET Q OM DET FINNS EN "LÖSNING" y SÅ ATT

$$A(x,y)=1 \Leftrightarrow x \in Q$$

SPRÅKET Q SOM VERIFIERAS AV TURINGMASKINEN A ÄR

$$Q = \{x \in \{0,1\}^* : \exists y \in \{0,1\}^* : A(x,y)=1\}$$

$NP = \{Q : \exists$ EN TM SOM VERIFIERAR Q I POLYNOMISK TID$\}$

ALTERNATIV DEFINITION AV NP MED NDTM

EN NDTM (ICKEDETERMINISTISK TURINGMASKIN) KÄNNER IGEN SPRÅKET Q I POLYNOMISK TID OM

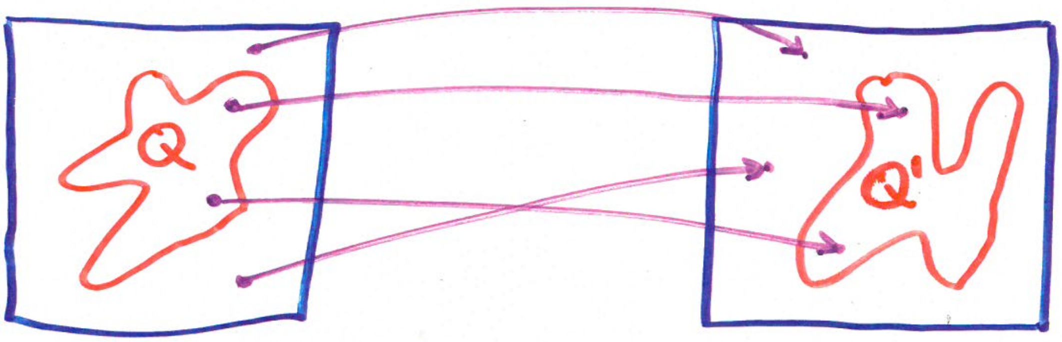
$$\begin{cases} x \in Q \Rightarrow \exists \text{ KEDJA AV ÖVERGÅNGAR AV POL. LÄNGD SOM ACCEPTERAR } x \\ x \notin Q \Rightarrow \nexists \text{ KEDJA AV ÖVERGÅNGAR SOM ACCEPTERAR } x \end{cases}$$

$NP = \{Q : \exists$ EN NDTM SOM KÄNNER IGEN Q I POL. TID$\}$

DEFINITIONERNA AV NP ÄR EKVIVALENTA!

POLYNOMISK REDUKTION

Q KAN REDUCERAS TILL Q' OM VARJE INSTANS x AV Q KAN OMFORMAS TILL EN INSTANS x' AV Q' SÅ ATT $x \in Q \Longleftrightarrow x' \in Q'$



OM REDUKTIONEN KAN GÖRAS I POLYNOMISK TID SKRIVER VI $Q \leq_p Q'$ EFTERSOM Q INTE KAN VARA SVÅRARE ATT LÖSA ÄN Q'.

OM $Q \leq_p Q'$ SÅ GÄLLER

$$Q' \in P \Rightarrow Q \in P$$

(OM Q' ÄR LÄTT SÅ ÄR Q OCKSÅ LÄTT)

$$Q \notin P \Rightarrow Q' \notin P$$

(OM Q ÄR SVÅRT SÅ ÄR Q' OCKSÅ SVÅRT)

DENNA TYP AV REDUKTION KALLAS KARP-REDUKTION.