

INF2080 – Logikk og beregninger

Forelesning 12: Grunnleggende maskiner

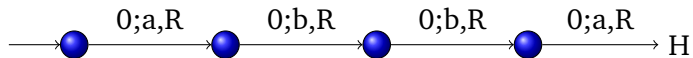


UiO : **Institutt for informatikk**

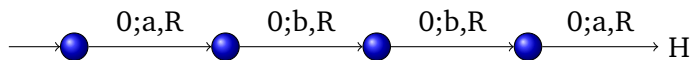
Sist oppdatert: 2012-02-21 10:00

12.1 Grunnleggende turingmaskiner

Basic 1: Skrive ord

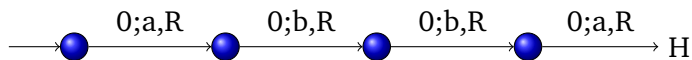


Basic 1: Skrive ord



Alfabet: $a, b, 0$ — 0 er blank

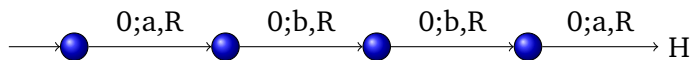
Basic 1: Skrive ord



Alfabet: $a, b, 0$ — 0 er blank

Spesifikasjon: Skriver abba og stopper

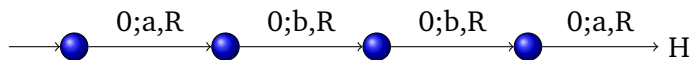
Basic 1: Skrive ord



Alfabet: $a, b, 0$ — 0 er blank

Spesifikasjon: Skrives abba og stopper

Basic 1: Skrive ord

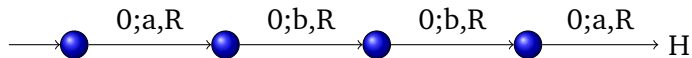


Alfabet: a, b, 0 — 0 er blank

Spesifikasjon: Skriver abba og stopper

- Til ord av lengde n brukes n tilstander + stopp

Basic 1: Skrive ord

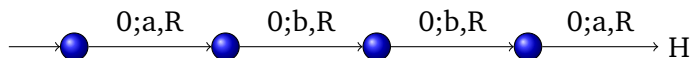


Alfabet: a, b, 0 — 0 er blank

Spesifikasjon: Skriver abba og stopper

- Til ord av lengde n brukes n tilstander + stopp
- Et ord er random om vi ikke klarer oss med færre tilstander

Basic 1: Skrive ord



Alfabet: $a, b, 0$ — 0 er blank

Spesifikasjon: Skriver $abba$ og stopper

- Til ord av lengde n brukes n tilstander + stopp
- Et ord er random om vi ikke klarer oss med færre tilstander
- Kolmogorov kompleksitet — kompleksitet av et ord = minste antall tilstander som trengs for å produsere ordet

Basic 1: Skrive ord



Alfabet: $a, b, 0$ — 0 er blank

Spesifikasjon: Skriver $abba$ og stopper

- Til ord av lengde n brukes n tilstander + stopp
- Et ord er random om vi ikke klarer oss med færre tilstander
- Kolmogorov kompleksitet — kompleksitet av et ord = minste antall tilstander som trengs for å produsere ordet
- Beveger seg bare i en retning — DFA

Basic 1: Skrive ord



Alfabet: $a, b, 0$ — 0 er blank

Spesifikasjon: Skriver $abba$ og stopper

- Til ord av lengde n brukes n tilstander + stopp
- Et ord er random om vi ikke klarer oss med færre tilstander
- Kolmogorov kompleksitet — kompleksitet av et ord = minste antall tilstander som trengs for å produsere ordet
- Beveger seg bare i en retning — DFA
- Fargeleggingsmaskin

Basic 1: Skrive ord

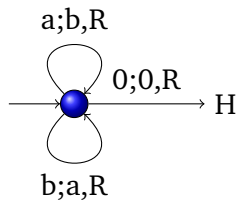


Alfabet: $a, b, 0$ — 0 er blank

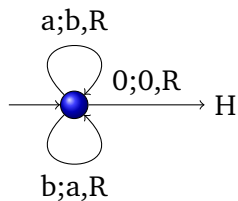
Spesifikasjon: Skriver $abba$ og stopper

- Til ord av lengde n brukes n tilstander + stopp
- Et ord er random om vi ikke klarer oss med færre tilstander
- Kolmogorov kompleksitet — kompleksitet av et ord = minste antall tilstander som trengs for å produsere ordet
- Beveger seg bare i en retning — DFA
- Fargeleggingsmaskin
- Trenger ikke være blanke vi skriver over

Basic 2: Bytte bokstaver

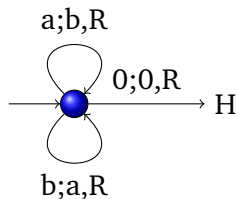


Basic 2: Bytte bokstaver



Alfabet: $a, b, 0$ — 0 er blank

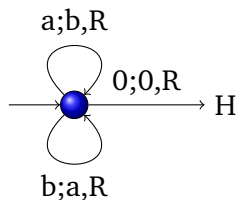
Basic 2: Bytte bokstaver



Alfabet: $a, b, 0$ — 0 er blank

Spesifikasjon: Erstatte a med b og b med a inntil den treffer 0

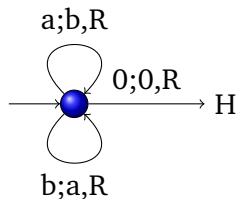
Basic 2: Bytte bokstaver



Alfabet: $a, b, 0$ — 0 er blank

Spesifikasjon: Erstatte a med b og b med a inntil den treffer 0

Basic 2: Bytte bokstaver

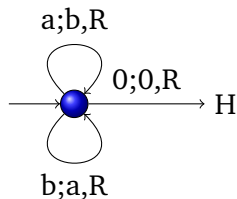


Alfabet: $a, b, 0$ — 0 er blank

Spesifikasjon: Erstatte a med b og b med a inntil den treffer 0

- Renske opp etter beregning

Basic 2: Bytte bokstaver

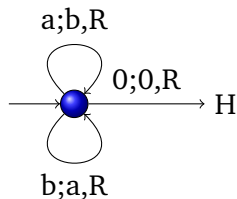


Alfabet: $a, b, 0$ — 0 er blank

Spesifikasjon: Erstatter a med b og b med a inntil den treffer 0

- Renske opp etter beregning
- Beveger seg bare i en retning — DFA

Basic 2: Bytte bokstaver

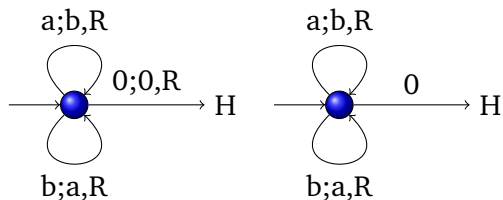


Alfabet: $a, b, 0$ — 0 er blank

Spesifikasjon: Erstatte a med b og b med a inntil den treffer 0

- Renske opp etter beregning
- Beveger seg bare i en retning — DFA
- Fargeleggingsmaskin

Basic 2: Bytte bokstaver

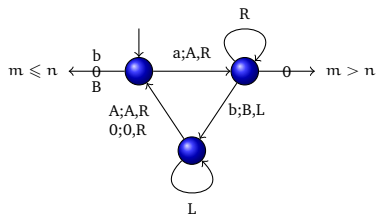


Alfabet: $a, b, 0$ — 0 er blank

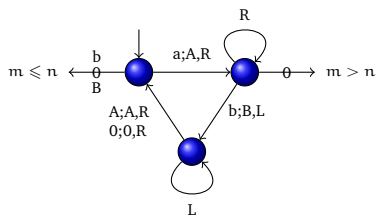
Spesifikasjon: Erstatter a med b og b med a inntil den treffer 0

- Renske opp etter beregning
- Beveger seg bare i en retning — DFA
- Fargeleggingsmaskin
- Forkortelser

Basic 3: Sammenlikning

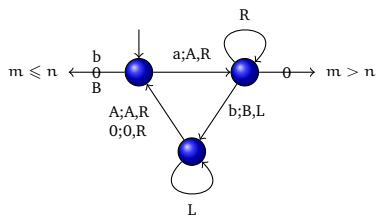


Basic 3: Sammenlikning



Alfabet: $a, b, 0$ — 0 er blank

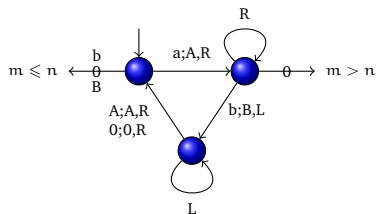
Basic 3: Sammenlikning



Alfabet: $a, b, 0$ — 0 er blank

Starttape: $0^* a^m b^n 0^*$ — start på venstre a

Basic 3: Sammenlikning

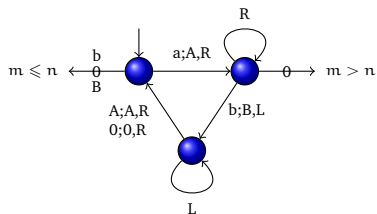


Alfabet: $a, b, 0$ — 0 er blank

Starttape: $0^* a^m b^n 0^*$ — start på venstre a

Spesifikasjon: Ser om det er fler a 'er enn b 'er

Basic 3: Sammenlikning



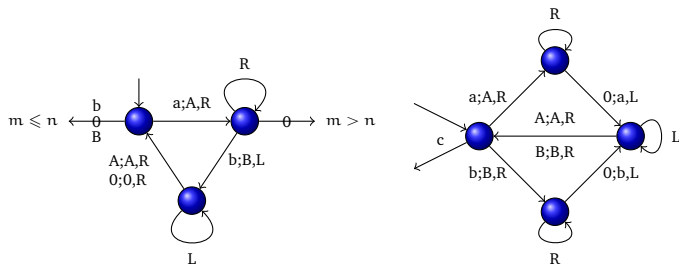
Alfabet: a, b, \emptyset — \emptyset er blank

Starttape: $0^* a^m b^n 0^*$ — start på venstre a

Spesifikasjon: Ser om det er fler a 'er enn b 'er

- Kan også sammenlikne ord i flere symboler

Basic 3: Sammenlikning



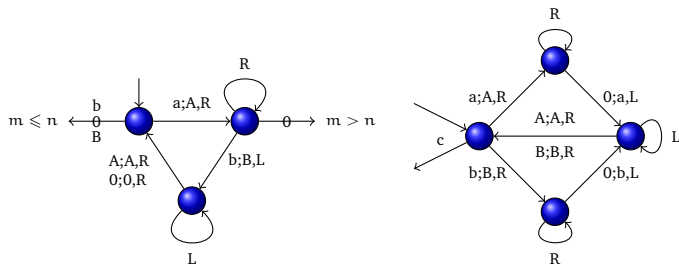
Alfabet: $a, b, 0$ — 0 er blank

Starttape: $0^* a^m b^n 0^*$ — start på venstre a

Spesifikasjon: Ser om det er fler a 'er enn b 'er

- Kan også sammenlikne ord i flere symboler
- Kan kopiere ord — $0^*(a \vee b)^*c0^*$

Basic 3: Sammenlikning

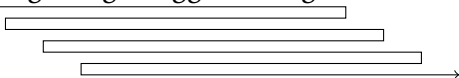


Alfabet: $a, b, 0$ — 0 er blank

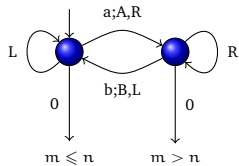
Starttape: $0^* a^m b^n 0^*$ — start på venstre a

Spesifikasjon: Ser om det er fler a 'er enn b 'er

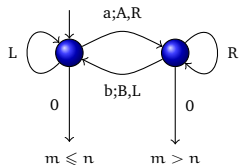
- Kan også sammenlikne ord i flere symboler
- Kan kopiere ord — $0^*(a \vee b)^*c0^*$
- Beveger seg i begge retninger



Basic 4: Omvendt sammenlikning

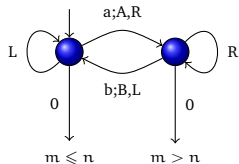


Basic 4: Omvendt sammenlikning



Alfabet: $a, b, 0$ — 0 er blank

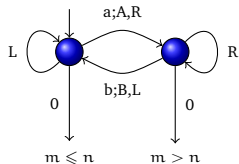
Basic 4: Omvendt sammenlikning



Alfabet: a, b, 0 — 0 er blank

Starttape: $0^* a^m b^n 0^*$ — start på venstre b

Basic 4: Omvendt sammenlikning

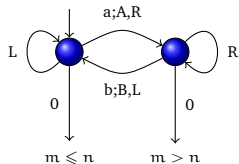


Alfabet: a, b, 0 — 0 er blank

Starttape: $0^* a^m b^n 0^*$ — start på venstre b

Spesifikasjon: Ser om det er fler a'er enn b'er

Basic 4: Omvendt sammenlikning



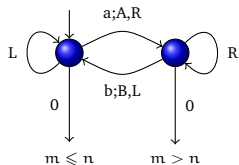
Alfabet: $a, b, 0$ — 0 er blank

Starttape: $0^* a^m b^n 0^*$ — start på venstre b

Spesifikasjon: Ser om det er fler a'er enn b'er

- Kan også sammenlikne ord i flere symboler — ekstra problem

Basic 4: Omvendt sammenlikning



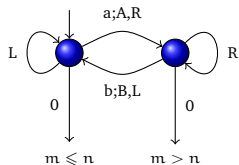
Alfabet: $a, b, 0$ — 0 er blank

Starttape: $0^* a^m b^n 0^*$ — start på venstre b

Spesifikasjon: Ser om det er fler a 'er enn b 'er

- Kan også sammenlikne ord i flere symboler — ekstra problem
- Kan kopiere ord

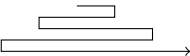
Basic 4: Omvendt sammenlikning



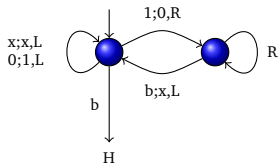
Alfabet: a, b, 0 — 0 er blank

Starttape: $0^* a^m b^n 0^*$ — start på venstre b

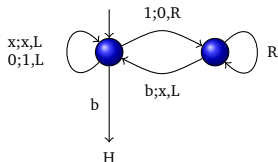
Spesifikasjon: Ser om det er fler a'er enn b'er

- Kan også sammenlikne ord i flere symboler — ekstra problem
- Kan kopiere ord
- Raskere sammenlikning
- Beveger seg i begge retninger 

Basic 5: Konverter unær/binær

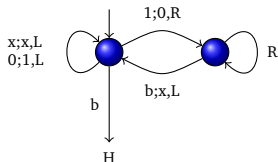


Basic 5: Konverter unær/binær



Alfabet: 0, 1, x, b — b er blank

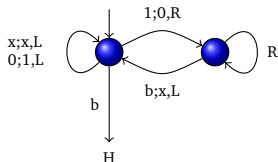
Basic 5: Konverter unær/binær



Alfabet: $0, 1, x, b$ — b er blank

Starttape: $b^*(0 \vee 1)^*b^*$ — start til høyre i tallet

Basic 5: Konverter unær/binær

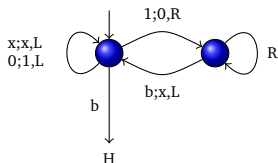


Alfabet: $0, 1, x, b$ — b er blank

Starttape: $b^*(0 \vee 1)^*b^*$ — start til høyre i tallet

Spesifikasjon: Konverterer det binære tallet $(0 \vee 1)^*$ til det unære tallet x^*

Basic 5: Konverter unær/binær



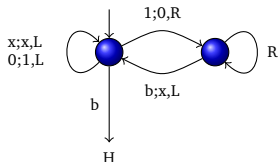
Alfabet: 0, 1, x, b — b er blank

Starttape: $b^*(0 \vee 1)^*b^*$ — start til høyre i tallet

Spesifikasjon: Konverterer det binære tallet $(0 \vee 1)^*$ til det unære tallet x^*

- Vi går i en løkke

Basic 5: Konverterer unær/binær



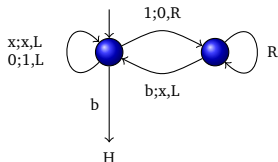
Alfabet: 0, 1, x, b — b er blank

Starttape: $b^*(0 \vee 1)^*b^*$ — start til høyre i tallet

Spesifikasjon: Konverterer det binære tallet $(0 \vee 1)^*$ til det unære tallet x^*

- Vi går i en løkke
- Trekker fra 1 binært og legger til x unært inntil vi treffer blank b

Basic 5: Konverterer unær/binær



Alfabet: $0, 1, x, b$ — b er blank

Starttape: $b^*(0 \vee 1)^*b^*$ — start til høyre i tallet

Spesifikasjon: Konverterer det binære tallet $(0 \vee 1)^*$ til det unære tallet x^*

- Vi går i en løkke
- Trekker fra 1 binært og legger til x unært inntil vi treffer blank b
- Går like bra i 10-tall systemet

Simulere turing mannen

Simulere turing mannen

- Turing mannen utfører enkle operasjoner — sammenlikne, slå opp i tabell, erstatte ord
- Med varianter av Basic 1-5 kan vi gjøre alt som turing mannen gjør

Simulere turing mannen

- Turing mannen utfører enkle operasjoner — sammenlikne, slå opp i tabell, erstatte ord
- Med varianter av Basic 1-5 kan vi gjøre alt som turing mannen gjør
- Dette er grunnlaget for Turings universelle maskin

Simulere turing mannen

- Turing mannen utfører enkle operasjoner — sammenlikne, slå opp i tabell, erstatte ord
- Med varianter av Basic 1-5 kan vi gjøre alt som turing mannen gjør
- Dette er grunnlaget for Turings universelle maskin
- Med den universelle maskinen får vi umulighets teoremer

Simulere turing mannen

- Turing mannen utfører enkle operasjoner — sammenlikne, slå opp i tabell, erstatte ord
- Med varianter av Basic 1-5 kan vi gjøre alt som turing mannen gjør
- Dette er grunnlaget for Turings universelle maskin
- Med den universelle maskinen får vi umulighets teoremer
- Maskiner som kode — intensjonale egenskaper

Simulere turing mannen

- Turing mannen utfører enkle operasjoner — sammenlikne, slå opp i tabell, erstatte ord
- Med varianter av Basic 1-5 kan vi gjøre alt som turing mannen gjør
- Dette er grunnlaget for Turings universelle maskin
- Med den universelle maskinen får vi umulighets teoremer
- Maskiner som kode — intensjonale egenskaper
- Maskiner som input/output — ekstensjonale egenskaper

Simulere turing mannen

- Turing mannen utfører enkle operasjoner — sammenlikne, slå opp i tabell, erstatte ord
- Med varianter av Basic 1-5 kan vi gjøre alt som turing mannen gjør
- Dette er grunnlaget for Turings universelle maskin
- Med den universelle maskinen får vi umulighets teoremer
- Maskiner som kode — intensjonale egenskaper
- Maskiner som input/output — ekstensjonale egenskaper
- Umulig å lage maskin som avgjør interessante ekstensjonal egenskaper