IN2080 Beregninger og kompleksitet

Løsningsforslag, eksamen våren 2024

Del 1: Tilstandsmaskiner (60 poeng)

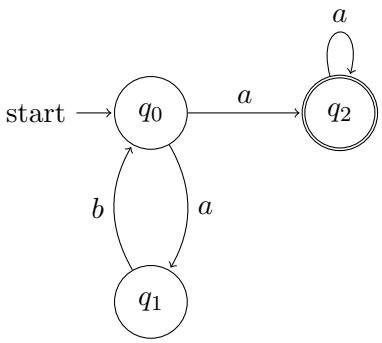
I hele del 1 antar vi at alfabetet er $\Sigma = \{a, b, c\}$ dersom ingenting annet blir sagt.

Oppgave 1 Lag en udeterministisk tilstandsmaskin (NFA) som gjenkjenner språket

$$L_1 = \{(ab)^n a^m \mid n \geq 0, m \geq 1\}.$$

Tilstandsmaskinen skal ha så få tilstander som mulig.

Løsningsforslag oppgave 1

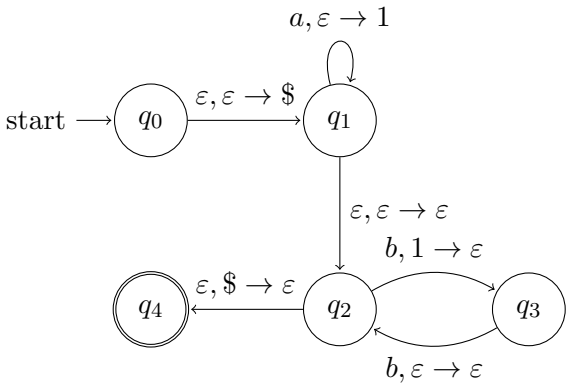


Oppgave 2 Lag en stabeltilstandsmaskin (PDA) som gjenkjenner språket

$$L_2 = \{a^n b^{2n} \mid n \geq 0\}.$$

Løsningsforslag oppgave 2

Stabel-alfabetet $\Gamma = \{\$, 1\}$ har en "bunn-av-stabel"-markør og en teller.



Oppgave 3 Denne oppgaven handler om det såkalte "Dyck"-språket D. D er et språk over alfabetet $\Sigma = \{(,)\}$ som består av balanserte parenteser. For eksempel er "(())" og "()()" i D, men ikke ")()" (fordi den første lukke-parentesen aldri åpnes) eller "(()" (fordi en av åpne-parentesene aldri lukkes).

a) List alle ord $w \in D$ med $|w| = 6$.

Vi kaller det maksimale antallet parenteser "inni" hverandre *dybden* til ordet. F.eks har "()()" dybde 1 og "((()))" dybde 3.

b) La D_3 være språket av Dyck-ord med dybde ≤ 3. Er D_3 regulært? Hvorfor / hvorfor ikke?

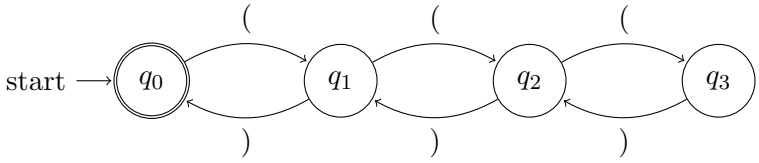
c) La D_n være språket av Dyck-ord med dybde $\leq n$ for en fastsatt $n \in \mathbb{N}$. Er D_n regulært? Hvorfor / hvorfor ikke?

d) Er D regulært? Hvorfor / hvorfor ikke?

Løsningsforslag oppgave 3

a) "()()()", "()(())", "(())()", "(()())", "((()))"

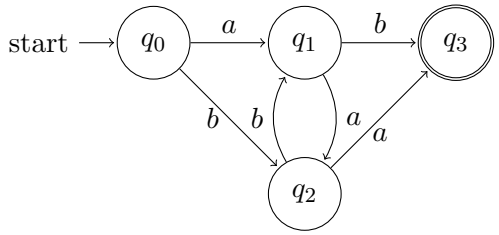
b) D_3 er regulært fordi det aksepteres av denne DFA-en:



c) Ja, D_n aksepteres av en DFA som i **b)** men med $n + 1$ tilstander.

d) Nei, $\{(^n)^m \mid n, m \in \mathbb{N}\}$ er regulært og $\{(^n)^m \mid n, m \in \mathbb{N}\} \cap D = \{(^n)^n \mid n \in \mathbb{N}\}$, men vi har sett at $\{(^n)^n \mid n \in \mathbb{N}\}$ ikke er regulært.

Oppgave 4 La L være språket som aksepteres av følgende DFA:



Gi et regulært uttrykk som beskriver L.

Løsningsforslag 4

$$ba \cup (a \cup bb)(ab)^*(b \cup aa)$$

Oppgave 5 Gi en kontekstfri grammatikk (CFG) som genererer

$$\{w \mid w \text{ har like mange a-er, b-er og c-er}\}$$

OBS: Denne oppgaven var rett og slett feil: språket er *ikke* kontekstfritt. Beklager! Kandidater som identifiserte feilen fikk bonuspoeng, men for alle andre telte vi ikke oppgaven.

Originalt (feil) Løsningsforslag En mulighet er å alltid generere én av hver bokstav. Vi må ta hensyn til alle de 6 mulige rekkefølgene.

$$S \to SaSbScS \mid SaScSbS \mid SbSaScS \mid SbScSaS \mid ScSaSbS \mid ScSbSaS \mid \varepsilon$$

Løsningsforslag 5 Denne gramatikken genererer ord med like mange a-er, b-er og c-er, men den generere ikke *alle* slike ord. For eksempel kan den ikke generere *aabbcc*, og vi kan bevise at språket ikke er kontekstfritt ved å pumpe nettopp slike ord ($a^p b^p c^p$ for pumpelengde p). En enklere løsning er å vite at snittet av et kontekstfritt og et regulært språk er kontekstrfritt, men snittet med $a^*b^*c^*$ er $\{a^n b^n c^n \mid n \geq 0\}$ som er et klassisk kontekstsensitivt språk.

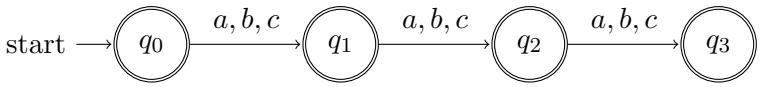
Oppgave 6 Språkene L_1, L_2 og $L_3 \subseteq \Sigma^*$ er gitt ved

- $L_1 = \{a^n b^m \mid n \neq m\}$
- $L_2 = \{w \mid |w| \leq 3\}$
- $L_3 = \{a^{n!} \mid n \geq 1\}$
  ($n!$ er definert som $n \times (n-1) \times (n-2) \times \ldots \times 2 \times 1$)

Et av språkene L_1, L_2, L_3 er regulært, et er kontekstfritt men ikke regulært, og et er ikke kontekstfritt.

a) Hvilket språk er regulært? Begrunn svaret.

Løsningsforslag a) L_2 **er regulært.** L_2 er endelig og dermed regulært. Dessuten aksepteres det av denne DFA-en:



b) Hvilket språk er kontekstfritt men ikke regulært? Begrunn svaret.

Løsningsforslag b) L_1 **er kontekstfritt men ikke regulært.** Vi viser at L_1 er kontekstfritt ved en kontekstfri grammatikk. Idéen er å dele $n \neq m$ i de to tilfellene $n < m$ og $n > m$ og generere dem hver for seg. XbB genererer $a^n b^n$ etterfulgt av minst én ekstra b, og aAX genererer minst én a etterfulgt av $a^n b^n$.

$$
\begin{aligned}
S &\to XbB \mid aAX \\
X &\to aXb \mid \varepsilon \\
B &\to bB \mid \varepsilon \\
A &\to aA \mid \varepsilon
\end{aligned}
$$

For å vise at L_1 ikke er regulært kan vi bruke pumpelemmaet, men det er ganske vanskelig (se Sipser's selected solutions 1.51). En enklere løsning er å se at $\{a^n b^m \mid n \in \mathbb{N}\} \setminus L_1 = \{a^n b^n \mid n \in \mathbb{N}\}$. Regulære språk er lukkede under differens (vi kan skrive $A \setminus B$ som $A \cap \overline{B}$) og $\{a^n b^m \mid n \in \mathbb{N}\}$ er regulært, så hvis L_1 var regulært ville $\{a^n b^n\}$ også være det — men $\{a^n b^n\}$ er ikke regulært!

c) Hvilket språk er ikke kontekstfritt? Begrunn svaret.

Løsningsforslag c) L_3 **er ikke kontekstfritt.** Vi bruker pumpelemmaet for kontekstfrie språk og velger $w = a^{p!}$. Da sier pumpelemmaet at w kan deles opp i $uvxyz$ med $|vxy| \leq p$, $|vy| > 0$ og uv^ixy^iz er i språket for alle i. Siden w kun består av a-er bryr vi oss egentlig bare om hvor mange det er, så la $k = |vy|$. Da vet vi at $0 < k \leq p$. Vi pumper med $i = 2$ og ser at $|uv^2xy^2z| = p! + k$. Ved hjelp av antagelsene vet vi at $p! < p! + k \leq p! + p < (p+1)!$, så $p! + k$ ligger strengt mellom $p!$ og $(p+1)!$ og kan ikke være fakultetet til et heltall.

(Dette argumentet holder ikke vann dersom $p = 1$, men da kan vi velge $i = 0$ eller $i = 3$ isteden.)

Del 2: Beregnbarhet- og Kompleksitetsteori (40 poeng)

Språket

$$A_{\text{TM}} = \{\langle M, w\rangle \mid M \text{ er en TM og } M \text{ aksepterer } w\}$$

er kjent fra Sipsers bok.

Oppgave 7 Språket A_{TM} er gjenkjennbart (*recognizable*). Gi en kort uformell forklaring på hvorfor A_{TM} er et gjenkjennbart språk.

Oppgave 8 Språket A_{TM} er ikke avgjørbart (*decidable*). Sipsers bok omtaler også en rekke andre språk som ikke er avgjørbare. Nevn to av dem. Svar kort, men definer språkene nøyaktig.

Oppgave 9 La L være et avgjørbart språk. La $\overline{L}$ betegne komplementspråket til L. Vis at (i) $L \leq_m A_{\text{TM}}$ og (ii) $\overline{L} \leq_m A_{\text{TM}}$.

Løsningsforslag oppgave 9 Vi viser (i). Ved definisjonen av $\leq_m$ må vi vise at det finnes beregnbar f slik at $w \in L$ hvis og bare hvis $f(w) \in A_{\text{TM}}$. Siden L er et avgjørtbart språk, så er også L et gjenkjennbart språk. Dermed finnes det en turingmaskin M som gjenkjenner L. La $f(w) = \langle M, w\rangle$ der $\langle M, w\rangle$ er kodingen av maskinen M og dens input w.

Vi viser (ii). Ved definisjonen av $\leq_m$ må vi vise at det finnes beregnbar f slik at $w \in \overline{L}$ hvis og bare hvis $f(w) \in A_{\text{TM}}$. Siden L er et avgjørbart språk,

så er $\overline{L}$ et gjenkjennbart språk (bruker lemma: L er avgjørbart hvis og bare hvis både L og $\overline{L}$ er gjenkjennbare). Dermed finnes det en turingmaskin M' som gjenkjenner $\overline{L}$. La $f(w) = \langle M', w\rangle$ der $\langle M', w\rangle$ er kodingen av maskinen M' og dens input w.

Oppgave 10 Hva betyr det at et språk er NP-komplett? Gi definisjonen.

I Sipsers bok brukes de utsagnslogiske konnektivene $\vee$ (disjunksjon), $\wedge$ (konjunksjon) og $\neg$ (negasjon)[1]. I denne oppgaven skal vi også arbeide med konnektivet $\mid$ (Sheffer-strek) som har følgende sannhetsverditabell:

X	Y	$X \mid Y$
sann	sann	usann
sann	usann	sann
usann	sann	sann
usann	usann	sann

La Ω være en mengde utsagnslogiske konnektiver. Vi sier at det utsagnslogiske uttrykket ϕ er et Ω-uttrykk dersom ϕ kun inneholder konnektiver fra mengden Ω. Det NP-komplette språket SAT (som er kjent fra Sipsers bok) kan nå formuleres ved

$$SAT = \{ \langle\phi\rangle \mid \phi \text{ er et tilfredsstillbart } \{\vee, \wedge, \neg\}\text{-uttrykk } \} .$$

La

$$SAT_1 = \{ \langle\phi\rangle \mid \phi \text{ er et tilfredsstillbart } \{\neg, \wedge\}\text{-uttrykk } \} .$$

Oppgave 11 Er SAT_1 NP-komplett? Begrunn svaret.

Løsningsforslag oppgave 11 Svaret er JA. Svaret kan begrunnes ved å vise at (i) SAT_1 er i NP og (ii) alle språk i NP kan polynom-tid reduseres til SAT_1. Det er lett å se at (i) holder. Man kan vise (ii) ved å vise at $SAT \leq_p SAT_1$, dvs., ved å vise at det finnes en polynom-tid beregnbar f slik at

$$\langle\phi\rangle \in SAT \text{ hvis og bare hvis } f(\langle\phi\rangle) \in SAT_1 . \qquad (*)$$

Vi kan definere en slik f ved å la $f(\langle\phi\rangle) = \langle g(\phi)\rangle$ der g er definert rekursivt ved

[1] I Sipsers bok skrives $\neg\phi$ ofte $\overline{\phi}$.

- $g(X) = X$ hvis X er en variable
- $g((\alpha \vee \beta)) = \neg(\neg g(\alpha) \wedge \neg g(\beta))$
- $g((\alpha \wedge \beta)) = (g(\alpha) \wedge g(\beta))$
- $g(\neg\alpha) = \neg g(\alpha)$.

Nå holder (*) siden ϕ og $g(\phi)$ er ekvivalente uttrykk (ϕ er sann under tilordningen S hvis og bare hvis $g(\phi)$ er sann under S). Vi observerer at lengden av $f(\langle\phi\rangle)$ er bundet av et polynom i lengden av $\langle\phi\rangle$ (vi har $|f(\langle\phi\rangle)| \leq k|\langle\phi\rangle|$ der k er en konstant). Dermed er det ganske opplagt at f kan beregnes i polynom tid.

La

$$SAT_2 \;=\; \{\; \langle\phi\rangle \;\mid \phi \text{ er et tilfredsstillbart } \{\,|\,\}\text{-uttrykk } \}\;.$$

Oppgave 12 Er SAT_2 NP-komplett? Begrunn svaret.

Løsningsforslag oppgave 12 Svaret er JA, men det er ikke opplagt. Det er greit nok SAT_2 er i NP, men det er ikke så opplagt at vi kan p-redusere for eksempel SAT, eller $3SAT$, til SAT_2.

Vi har $\neg\alpha = (\alpha\,|\,\alpha)$ og $\alpha \wedge \beta = (\alpha\,|\,\beta)\,|\,(\alpha\,|\,\beta)$ (i tillegg til $(\alpha \vee \beta) = \neg(\neg\alpha \wedge \neg\beta)$). Dermed kan ethvert $\{\vee, \wedge, \neg\}$-uttrykk skrives om til et ekvivalent $\{\,|\,\}$-uttrykk, men vi skal også være oppmerksom på at uttrykkets størrelse kan øke dramatisk under en slik omskriving. Hvis vi skriver om ved å bruke ekvivalense (likehetene) gitt over, så får vi

$$\begin{aligned} X \vee Y \;&=\; \neg(\neg X \wedge \neg Y) \;=\; (\neg X \wedge \neg Y)\,|\,(\neg X \wedge \neg Y) \;= \\ &((\neg X\,|\,\neg Y)\,|\,(\neg X\,|\,\neg Y))\,|\,((\neg X\,|\,\neg Y)\,|\,(\neg X\,|\,\neg Y)) \;= \\ &(((X\,|\,X)\,|\,(Y\,|\,Y))\,|\,((X\,|\,X)\,|\,(Y\,|\,Y)))\,|\,(((X\,|\,X)\,|\,\neg(Y\,|\,Y))\,|\,((X\,|\,X)\,|\,(Y\,|\,Y))) \end{aligned}$$

og det ser jo slettes ikke bra ut.

Vi viser at det finnes en polynom-tid beregnbar f slik at

$$\langle\phi\rangle \in SAT \text{ hvis og bare hvis } f(\langle\phi\rangle) \in SAT_2\;. \qquad (**)$$

Først definerer vi $g(\phi)$ der ϕ er et $\{\vee, \wedge, \neg\}$-uttrykk. Velg en variabel Z i ϕ (ethvert $\{\vee, \wedge, \neg\}$-uttrykk inneholder minst en variabel, la f.eks. Z være variablen lengst til venstre i uttrykket). La $1 := ((Z\,|\,Z)\,|\,Z)$. Vi definerer $g(\phi)$ rekursivt ved

- $g(X) = X$ hvis X er en variabel
- $g(\neg\alpha) = (g(\alpha)\,|\,1)$
- $g((\alpha \wedge \beta)) = ((g(\alpha)\,|\,g(\beta))\,|\,1)$
- $g((\alpha \vee \beta)) = ((((g(\alpha)\,|\,1)\,|\,(g(\beta)\,|\,1))\,|\,1)\,|\,1)$.

La $f(\langle\phi\rangle) = \langle g(\phi)\rangle$. Da holder (**) siden ϕ og $g(\phi)$ er ekvivalente (ϕ er sann under tilordningen S hvis og bare hvis $g(\phi)$ er sann under S). Videre så finnes det en konstant k slik at $|f(\langle\phi\rangle)| \leq k|\langle\phi\rangle|$. Dermed er det rimelig greit å se at f kan beregnes i polynom tid, og vi kan konkludere at $SAT \leq_p SAT_2$.