IN2080 Beregninger og kompleksitet

Hjemmeeksamen våren 2021

Del 1: Automater (60 poeng)

Oppgave 1 La $\Sigma = \{a, b\}$ være alfabetet.

a) Lag en automat (DFA eller NFA) som avgjør

1. $(ab)^* a$
2. $\{w^* \mid w \textit{ inneholder minst en } a \textit{ og maksimalt en } b\}$

b) Lag et regulært uttrykk som definerer

$$\{w \mid \textit{alle symboler i partallsposisjon i } w \textit{ er en } b\}$$

Oppgave 2 La L være følgende språk over alfabetet $\Sigma = \{a, b\}$:

$$L = \{(ab)^n\, z \mid |z|_{ab} = n, n \geq 0\}$$

Lag en PDA som gjenkjenner L.

Oppgave 3 Den kontekstfrie grammatikken G er gitt ved reglene

$$\begin{aligned} S &\to a\, S\, c \mid X\, Y \mid \varepsilon \\ X &\to b\, X \mid \varepsilon \\ Y &\to S\, b \end{aligned}$$

a) Hvilke av følgende ord er i $\mathcal{L}(G)$: ac, $aacca$, $acac$? Begrunn svaret.

b) Oversett G til Chomsky normalform.

c) Lag en PDA som gjenkjenner språket $\mathcal{L}(G)$.

d) Kan alle ord på formen $(ab)^n ac(bc)^n$ genereres fra G? Begrunn svaret.

Oppgave 4 La $\Sigma = \{a, b, c\}$ være et alfabet. La $x \sqsubseteq y$ bety at x er en substreng av y (dvs. at det finnes $z_1, z_2 \in \Sigma^*$ slik at $z_1 x z_2 = y$) og la $|y]_x$ være antall forekomster av substrengen x i strengen y. Språkene L_1, L_2 og $L_3 \subseteq \Sigma^*$ er gitt ved

- $L_1 = \{z_1\, c\, z_2\, c\, z_3 \mid z_1, z_2, z_3 \in \{a, b\}^* \mathit{og}\ |z_1|_a = |z_2|_a = |z_3|_a\}$
- $L_2 = \{z\, b^n \mid |z|_b = n, n \geq 0\}$
- $L_3 = \{z \mid z\ \mathit{inneholder\ ikke\ substrengen\ bb}\}$

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.

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

c) Hvilket språk er ikke kontekstfritt? Begrunn svaret.

d) Er $L_2 \cap L_3$ et regulært språk? Begrunn svaret.

Del 2: Beregnbarhetsteori (10 poeng)

Vi minner om at en eksekvering av en TM M på input w terminerer ved at eksekveringen når enten en *akspeterende konfigurasjon* eller en *avvisende konfigurasjon* (M aksepterer w hvis eksekveringen terminerer i en aksepterende konfigurasjon, M avviser w hvis eksekveringen stopper i en avvisende konfigurasjon). Eksekveringer som ikke når en aksepterende eller en avvisende konfigurasjon terminerer ikke (per definisjon).

Språket

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

er kjent fra Sipsers bok (se side 207). Språket A_{TM} er gjenkjennbart[1], men ikke avgjørbart[2]. Vi definerer de beslektede språkene B_{TM} og C_{TM} ved

$$B_{\mathrm{TM}} = \{\langle M, w\rangle \mid M \text{ er en TM og } M \text{ avviser } w\}\ .$$

og

$$C_{\mathrm{TM}} = \{\langle M, u, v\rangle \mid M \text{ er en TM, } M \text{ aksepterer } u \text{ og } M \text{ aksepterer } v\}\ .$$

[1] Sipser bruker ordet *recognizable*

[2] Sipser bruker ordet *decidable*

Oppgave 5 Er B_{TM} et avgjørbart språk? Gi en kort begrunnelse av svaret.

Oppgave 6 Er B_{TM} et gjenkjennbart språk? Gi en kort begrunnelse av svaret.

Oppgave 7 Er $\overline{B_{\text{TM}}}$ et avgjørbart språk? Gi en kort begrunnelse av svaret.

Oppgave 8 Er $\overline{B_{\text{TM}}}$ et gjenkjennbart språk? Gi en kort begrunnelse av svaret.

Oppgave 9 Vis at $A_{\text{TM}} \leq_m C_{\text{TM}}$. Du skal beskrive reduksjonen f og argumentere for at

$$w \in A_{\text{TM}} \quad \Leftrightarrow \quad f(w) \in C_{\text{TM}} \; .$$

Oppgave 10 Vis at $C_{\text{TM}} \leq_m A_{\text{TM}}$. Du skal beskrive reduksjonen f og argumentere for at

$$w \in C_{\text{TM}} \quad \Leftrightarrow \quad f(w) \in A_{\text{TM}} \; .$$

Oppgave 11 Er det slik at $\overline{B_{\text{TM}}} \leq_m B_{\text{TM}}$? Begrunn svaret.

Del 3: Kompleksitetsteori (30 poeng)

La G være en rettet graf. En *rettet sti* i G er en sekvens $a_1, \ldots, a_n$ av noder i G slik at det går en kant fra a_i til a_{i+1} (for $i = 1, \ldots, n_1$).

En rettet graf G inneholder en *bulgarsk ring* dersom det finnes en rettet sti $a_1, a_2, \ldots a_n$ i G slik at

1. enhver node i G forkommer en eller flere ganger i sekvensen $a_1, \ldots, a_n$, og
2. det går en kant fra a_n til a_1.

Merk at G inneholder en bulgarsk ring hvis og bare hvis det går en rettet sti mellom to vilkårlige noder i G.

Vi definerer språket BR ved

$$BR \;\; = \;\; \{\langle G \rangle \mid G \text{ er en rettet graf og } G \text{ inneholder en bulgarsk ring}\} \; .$$

Oppgave 12 Beskriv en deterministisk polynom-tid TM M som avgjør BR. Skisser hvordan M arbeider, og argumenter for at M arbeider i polynom tid. (Det holder at M arbeider i polynom tid: En TM som arbeider i tid $O(n^2)$ vil f.eks. ikke gi høyere uttelling enn en som arbeider i tid $O(n^4)$.)

Oppgave 13 Vis at BR er i NL.

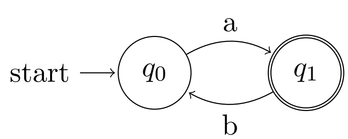
Oppgave 14 Vis at BR er i NL-komplett. Du kan benytte at $PATH$ er NL-komplett (språket $PATH$ er mye omtalt i Sipsers bok).

——————————————— LØSNINGSFORSLAG

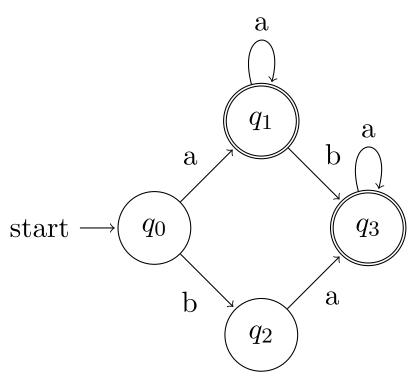
1 Del 1: Automater

Oppgave 1

a) $(ab)^*a$

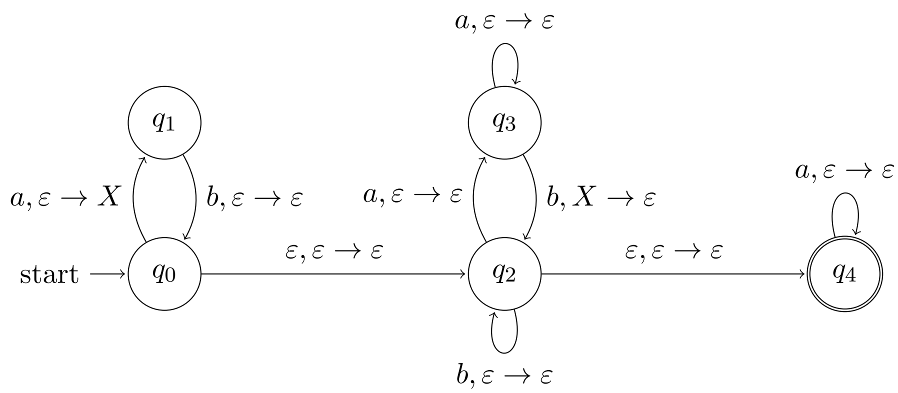


$\{w^* \mid w\ inneholder\ minst\ en\ a\ og\ maksimalt\ en\ b\}$
Vi løser oppgaven $\{w \mid w\ inneholder\ minst\ en\ a\ og\ maksimalt\ en\ b\}$ i stedet, som er morsommere.



b) $((a \cup b)b)^*((a \cup b) \cup \varepsilon)$

Oppgave 2 Definer en PDA $(Q, \Sigma, \Gamma, \delta, q_0, F)$ der $Q = \{q_0, q_1, q_2, q_3\}$, $\Sigma = \{a, b\}$, $\Gamma = \{X\}$ of $F = \{q_2\}$. Transisjonsfunksjonen δ er angitt på tegningen av automaten:



Her aksepterer automaten en streng hvis den når q_4 med tom input streng og tom stack. (Man kan også legge på enkstra tilstander først og sist i automaten som pusher \$ før q_0 og popper \$ etter q_4, I så fall er det ny siste tilstand som skal være aksepterende i stedet for q_4.)

Oppgave 3

a) Legg merke til at alle ord som ikke inneholder symbolet b må genereres av den enklere grammatikken

$$S \to a\,S\,c \mid \varepsilon$$

Det betyr at alle slike ord må være på formen $a^n c^n$.

b) Vi oversetter grammatikken til Chomsky normalform ved å følge metoden fra forelesning.

Først legger vi til ny starttilstand S_0 og fjerner tomme produksjoner. Det gir

$$\begin{aligned}
&S_0 \to S \mid \varepsilon \\
&S \to a\,S\,c \mid X\,Y \mid Y \mid a\,c \\
&X \to b\,X \mid b \\
&Y \to S\,b \mid b
\end{aligned}$$

Så fjerner vi all regler som erstatter en variabel med en annen (her $S \to Y$

og $S_0 \to S$):

$$\begin{aligned}
&S_0 \to a\,S\,c \mid X\,Y \mid S\,b \mid b \mid a\,c \mid \varepsilon\\
&S \to a\,S\,c \mid X\,Y \mid S\,b \mid b \mid a\,c\\
&X \to b\,X \mid b\\
&Y \to S\,b \mid b
\end{aligned}$$

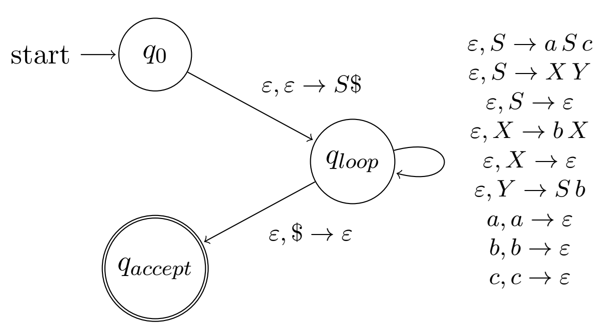
Så splitter vi reglene med tre symboler i høyresiden ved å innføre en ny variabel, og lager produskjoner for alle terminalene:

$$\begin{aligned}
&S_0 \to A\,S_1 \mid X\,Y \mid S\,B \mid b \mid A\,C \mid \varepsilon\\
&S \to A\,S_1 \mid X\,Y \mid S\,B \mid b \mid A\,C\\
&S_1 \to S\,C\\
&X \to B\,X \mid b\\
&Y \to S\,B \mid b\\
&A \to a\\
&B \to b\\
&C \to c
\end{aligned}$$

c) Vi følger oppskriften fra forelesningene og lager en PDA $(Q, \Sigma, \delta, q_0, F)$ med tre tilstander, så $Q = \{q_0, q_{loop}, q_{accept}\}$, $\Sigma = \{a, b, c\}$, $F = \{q_{accept}\}$ og transisjonsfunksjonen δ er definert ved:

$$\begin{aligned}
&q_0 \xrightarrow{\varepsilon,\varepsilon\to S\$} q_{loop}\\
&q_{loop} \xrightarrow{\varepsilon,S\to a\,S\,c} q_{loop}\\
&q_{loop} \xrightarrow{\varepsilon,S\to X\,Y} q_{loop}\\
&q_{loop} \xrightarrow{\varepsilon,S\to \varepsilon} q_{loop}\\
&q_{loop} \xrightarrow{\varepsilon,X\to b\,X} q_{loop}\\
&q_{loop} \xrightarrow{\varepsilon,X\to \varepsilon} q_{loop}\\
&q_{loop} \xrightarrow{\varepsilon,Y\to S\,b} q_{loop}\\
&q_{loop} \xrightarrow{a,a\to \varepsilon} q_{loop}\\
&q_{loop} \xrightarrow{b,b\to \varepsilon} q_{loop}\\
&q_{loop} \xrightarrow{c,c\to \varepsilon} q_{loop}\\
&q_{loop} \xrightarrow{\varepsilon,\$\to \varepsilon} q_{accept}
\end{aligned}$$

Man kan også tegne selve automaten:



Merk at vi bruker konvensjonen fra forelesningene og skriver, e.g., $q_{loop} \xrightarrow{\varepsilon,Y\to S\,b} q_{loop}$ som en forkortelse for to transjoner $q_{loop} \xrightarrow{\varepsilon,Y\to b} q_1$ og $q_1 \xrightarrow{\varepsilon,\varepsilon\to S} q_{loop}$, der q_1 er en ny hjelpetilstand i automaten.

d) Vi bruker induksjon til å argumentere for at alle ord $(a\,b)^n\,a\,c\,(b\,c)^n$ kan genereres fra G.

Basistilfelle: $n = 0$. Vi har $S \rightsquigarrow a\,S\,c \rightsquigarrow a\,\varepsilon\,c = a\,c$.

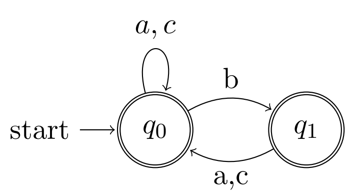
Induksjonstilfellet. Anta at $(a\,b)^n\,a\,c\,(b\,c)^n$ kan genereres fra G. Da vet vi at det fins en utledning $S \rightsquigarrow (a\,b)^n\,a\,c\,(b\,c)^n$ (induksjonshypotese). Vi viser at $(a\,b)^{n+1}\,a\,c\,(b\,c)^{n+1}$ kan genereres fra G.

Vi har at $S \rightsquigarrow a\,S\,c \rightsquigarrow a\,X\,Y\,c \rightsquigarrow a\,b\,Y\,c \rightsquigarrow a\,b\,S\,b\,c$. Ved induksjonshypotesen $S \rightsquigarrow (a\,b)^n\,a\,c\,(b\,c)^n$ får vi da

$$a\,b\,S\,b\,c \rightsquigarrow a\,b\,(a\,b)^n\,a\,c\,(b\,c)^n\,b\,c = (a\,b)^{n+1}\,a\,c\,(b\,c)^{n+1}.$$

Oppgave 4

a) L_3 er regulært. Vi viser dette ved å lage en DFA som gjenkjenner L_3.



b) L_2 er kontekstfritt men ikke regulært. Vi viser at L_2 er kontekstfritt ved å gi en grammatikk som genererer språket.

$$S \to a\,S \mid c\,S \mid bSb \mid \varepsilon$$

Vi viser så at L_2 ikke er regulært ved å anvende pumpelemmaet for regulære språk. Anta L_2 regulært. Da fins pumpelengde p slik at for alle ord w med $|w| \geq p$ så har vi $w = xyz$, $|y| > 0$, $|xy| \leq p$ og $xy^iz \in L_2$ for alle $i \geq 0$.

La $w = b^pab^p$. Vi har $x = b^k$ og $y = b^l$ slik at $k + l \leq p$ og $k < p$. Dvs. at y består av minst én b. Men da har vi $xy^0z = b^kab^p \notin L_2$. Motsigelse.

c) L_1 er ikke kontekstfritt. Vi kan vise dette ved å bruke pumpelemmaet for CFL.

Anta at L_1 er kontekstfritt. Da sier pumpelemmaet for CGL at det fins en pumpelengde p slkik at for alle ord z slik at $|w[\geq p$ så har vi $w = uvxyz$ med $|vy| > 0$, $|vxy| \leq p$ og $uv^ixy^iz \in L_1$ for alle $i \geq 0$.

La $w = a^pc\,a^pc\,a^p$. Da er det slik at uansett hvordan vi deler $w = uvxyz$ så vil $uv^0xy^0z \notin L_1$.

Tilfelle 1: Anta at v eller y inneholder c. Opplagt.
Tilfelle 2: Anta at $x = c$. Da må minst en av v og y inneholde minst en a. La

oss først se på første forekomst av c. Vi har at ordet $uv^0xy^0z = a^{p_1} c\, a^{p_2} c\, a^p$ der $p_1 < p$ eller $p_2 < p$. Dermed får vi at $uv^0xy^0z \notin L_1$. Hvis x er andre forekomst av c får vi $uv^0xy^0z = a^p c\, a^{p_1} c\, a^{p_2}$ der $p_1 < p$ eller $p_2 < p$. Dermed får vi at $uv^0xy^0z \notin L_1$.
Tilfelle 3: Anta at vxy er i en av forekomstene av a^p. Umiddelbart at $uv^0xy^0z \notin L_1$.

Dermed kan ikke w pumpes og L_1 kan ikke være kontekstfritt. Motsigelse.

d) Legg merke til at når vi tar interseksjonen av L_2 og L_2 så fjerner vi alle strenger i L_2 der forekomster av b er etter hverandre. Da står vi igjen med ord som er helt uten b, eller $\{z_1\, z_2\, b \mid |z_1|_b = 1 \ og \ |z_2|_b = 0\}$. Resultatet kan vi skrive som et regulært uttrykk:

$$((a \cup c)^* b(a \cup c)(a \cup c)^* b) \cup (a \cup c)^*$$

Vi vet dermed at $L_2 \cap L_3$ er regulært.

2 Del 2: Beregnbarhetsteori

Oppgave 5 La M' være turingmaskinen som aksepterer alt som M aviser og avviser alt som M aksepterer (så M' er M der q_{accept} og q_{reject} er byttet om. La $f(\langle M, w\rangle) = \langle M', w\rangle$. Da er f en beregnbar funksjon slik at

$$\langle M, w\rangle \in A_{\mathrm{TM}} \Leftrightarrow f(\langle M, w\rangle) \in B_{\mathrm{TM}} .$$

Dermed har vi $A_{\mathrm{TM}} \leq_m B_{\mathrm{TM}}$. Dermed, hvis B_{TM} var avgjørbart, så ville også A_{TM} være avgjørbart. Det er oppgitt i oppgaveteksten at A_{TM} ikke er avgjørbart. Dermed er ikke B_{TM} avgjørbart. (Kandidatene kan selvsagt få full uttelling på langt mindre formelle svar enn det overstående.)

Oppgave 6 Vi har $B_{\mathrm{TM}} \leq_m A_{\mathrm{TM}}$ der reduksjonen f er som i forrige oppgave (den virker begge veier). Det er oppgitt i oppgaveteksten at A_{TM} er gjenkjennbart. Dermed blir B_{TM} gjenkjennbart.

Oppgave 7 Et språk er avgjørbart hvis og bare hvis dets komplementspråk er avgjørbart. Ved oppgave 5 er ikke B_{TM} avgjørbart. Dermed er ikke $\overline{B_{\mathrm{TM}}}$ avgjørbart.

Oppgave 8 Ved oppgave 6 er B_{TM} gjenkjennbart. Hvis også $\overline{B_{\mathrm{TM}}}$ hadde vært gjennkjennbart, så ville B_{TM} ha vært avgjørbart. Ved oppgave 7 er $\overline{B_{\mathrm{TM}}}$ ikke avgjørbart. Dermed er $\overline{B_{\mathrm{TM}}}$ ikke gjenkjennbart.

Oppgave 9 La $f(\langle M, w\rangle) = \langle M, w, w\rangle$. Da er det opplagt at f er beregnbar, og det er opplagt at vi har

$$\langle M, w\rangle \in A_{\mathrm{TM}} \Leftrightarrow f(\langle M, w\rangle) \in C_{\mathrm{TM}} \ .$$

Oppgave 10 La $f(\langle M, u, v\rangle) = \langle M', u\sharp v\rangle$ hvor $\sharp$ ikke forekommer i M's inputalfabet og M' er gitt ved

$M' =$ input $u\sharp v$
kjør M med input u
AVVIS (og terminer) hvis M avviser u
kjør M med input v
AKSEPTER hvis M aksepter v; AVVIS hvis M avviser v.

Det er opplagt at f er beregnbar, og det er opplagt at

$$\langle M, u, v\rangle \in C_{\mathrm{TM}} \Leftrightarrow f(\langle M, u, v\rangle) \in A_{\mathrm{TM}} \ .$$

Oppgave 11 Det er ikke slik at $\overline{B_{\mathrm{TM}}} \leq_m B_{\mathrm{TM}}$. Språket B_{TM} er gjennkjenbart, og språket $\overline{B_{\mathrm{TM}}}$ er ikke gjenkjennbart. Hvis det hadde vært slik at $\overline{B_{\mathrm{TM}}} \leq_m B_{\mathrm{TM}}$, så ville $\overline{B_{\mathrm{TM}}}$ ha vært gjenkjennbart.

3 Del 3: Kompleksitetssteori

Oppgave 12 La $a_1, \ldots, a_k$ være nodene i G. I Sipsers bok gis det en deterministisk polynom-tid algoritme for å sjekke hvorvidt det går en rettet sti fra en node s til en node t i en rettet graf. Denne algoritmen kan brukes til å sjekke om det går en rettet sti fra a_i til a_j for alle $i, j \in \{1, \ldots k\}$ (og G inneholder en bulgarsk ring hvis og bare hvis så er tilfellet). Det er k^2 par (a_i, a_j).

Oppgave 13 (skisse) La $a_1, \ldots, a_k$ være nodene i G. For alle $i, j \in \{1, \ldots k\}$, sjekk nondeterministisk om det finnes en rettet sti fra a_i til a_j. Dette kan gjøres med et endelig antall pekere inn i input.

Oppgave 14 Vi må vise at $\mathit{PATH} \leq_L \mathit{BR}$ (dette sammen med $\mathit{BR} \in \mathrm{NL}$ gir at BR er NL-komplett). Vi beskriver en log-space beregnbar f slik at

$$\langle G, s, t\rangle \in \mathit{PATH} \Leftrightarrow f(\langle G, s, t\rangle) \in \mathit{BR} \ . \tag{1}$$

La $a_1, \dots a_n$ være nodene i G. La G' være G utvidet med kantene (t, a_i) og (a_i, s) (for $i = 1, \dots n$). La $f(\langle G, s, t\rangle) = G'$.

Det er lett å se at f er log-space beregnbar. Vi må argumentere for at (1) holder.

Anta $\langle G, s, t\rangle \in \mathit{PATH}$. La

$$sa_{i_1} \dots a_{i_k} t$$

være en rettet sti fra s til t i G Da er

$$sa_{i_1} \dots a_{i_k} t a_1 s a_{i_1} \dots a_{i_k} t a_2 s a_{i_1} \dots a_{i_k} t a_3 s \dots s a_{i_1} \dots a_{i_k} t a_n s$$

en bulgarsk ring i $f(\langle G, s, t\rangle)$.

Anta $\langle G, s, t\rangle \notin \mathit{PATH}$. Da vil det ikke finnes en rettet sti fra s til t i G. Grafen $f(\langle G, s, t\rangle)$ inneholder de samme nodene som G, og det er rimelig lett å se at kanter som går ut fra t eller inn til s ikke kan gi opphav tile en sti fra s til t. Dermed finnes det ingen sti fra s til t i $f(\langle G, s, t\rangle)$. Dermed finnes det heller ingen bulgarsk ring i $f(\langle G, s, t\rangle)$.