

INF2820 - V2017 - Oppgavesett 1 – 24.1

Noen ting å merke seg:

1. De obligatoriske oppgavene vil ikke rekke å dekke hele pensum. Det vil være eksamensoppgaver som ikke er dekket av de obligatoriske oppgavene.
2. De obligatoriske oppgavene vil bygge på ukesoppgavene. Noen ganger kan det være nødvendig å ha gjort noen ukesoppgaver for å få til de obligatoriske oppgavene.
3. Du er forventet å arbeide i gjennomsnitt 13 timer i uka med dette emnet. Hvis du går på forelesninger og gruppe, er det fremdeles 9 timer igjen. For å hjelpe deg med å bruke denne tiden fornuftig, vil vi prøve å gi flere arbeidsoppgaver enn de du rekker å gjøre i 2 timer på gruppa. Du bør arbeide med dem selv på egen hånd.
4. Det vil være en blanding av teoretiske og praktiske oppgaver. Ukesoppgavene vil til dels være mer teoretiske enn de obligatoriske oppgavene. Det vil eksamensoppgavene også. Det er viktig å arbeide med de teoretiske oppgavene med tanke på eksamen.

Endelige tilstandsautomater

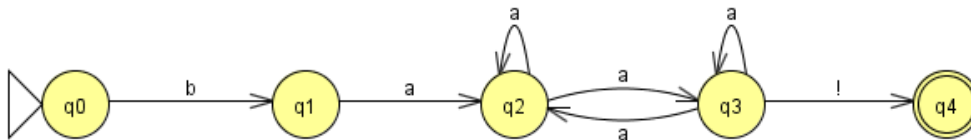
Tema for dette settet er automater. Vi vil gå noe lengre enn læreboka til Jurafsky og Martin. Dette vil være dekket av forelesningene (presentasjonene som blir lag ut), og arbeid med programmet JFLAP der du må arbeide deg gjennom deler av "tutorial". Du anbefales å laste ned programmet og begynne å arbeide med oppgavene før gruppa. Oppgavene med automater her er liknende til oppgaver du kan få til eksamen.

Python og NLTK

Jeg har laget et eget ark om hva som forventes av bakgrunn i Python og NLTK fra INF1820. Det er viktig å komme i gang med dette tidlig, og de som hverken har tatt INF1820 eller har kjennskap til Python, bør arbeide med dette så snart som mulig. Det kan også være en god repetisjon for de som har INF1820. På gruppa neste uke vil vi bruke Python og NLTK og forutsette at bakgrunnskunnskapen er på plass.

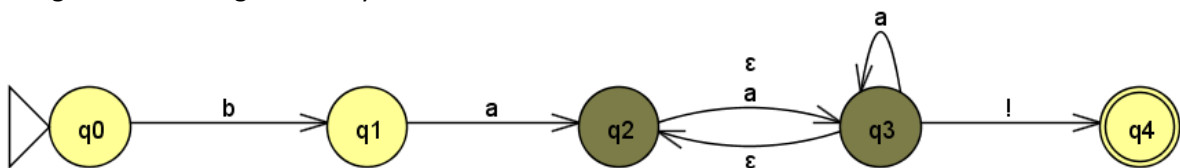
JFLAP og automater

1. Last ned JFLAP_Thin.jar fra JFLAP Version 7.0 på siden <http://www.cs.duke.edu/csed/jflap/jflaptmp/>
 - a. Du kan f.eks. bruke wget-kommandoen fra et terminalvindu slik:
wget <URL>
2. Kjør programmet fra kommandolinjen slik:
java -jar JFLAP_Thin.jar
(Programmet kan også brukes under Windows).
3. Se på Tutorial på <http://www.jflap.org/tutorial/>. Jobb deg gjennom "Finite Automata/Construct and Run" og "Manipulating Transitions", "Add Trap State to DFA" og "Convert to DFA".
4. Lag og test en DFA:
 - a. Skriv inn brekespråket på figur 2.10 i J&M seksj 2.10 og lagr den under et passende navn.
 - b. Test at maskinen gjør det den skal ved å kjøre den i step-mode på noen sekvenser som:
baaaaa! baaaaa baabaa! abaaa! b!
 - c. Legg til et "svart hull" ("trap/fail/sink state") som på figur 2.14.
 - d. Modifiser FSA-en fra (b) til figur 2.17, og lagr den under et passende navn.
 - e. Kjør den på noen sekvenser og se hvordan ikkedeterminismen virker.
5. Vi skal se på enda en variant av en FSA for det samme språket. Modifiser FSA-en fra (b) til følgende og lagr den under et passende navn.



Kjør maskinen på følgende input og se hvordan den oppfører seg
ba! baa! baaa! baaaa! baaaaa!

6. **E-transisjoner og ikkedeterminisme** (bakgrunn for oblig. 1)
Lag NFA-er for følgende uttrykk i JFLAP:



Prøv den med forskjellige inputter som: baaaa! og baaaa

Prøv inputtene både med "Step by State" og med "Step with Closure". Skjønner du forskjellen?

Oppgave 1 - Oppvarming

La alfabetet A være $\{a,b,c\}$

1. Lag en automat for språket som består av alle uttrykk som inneholder minst en a (og vilkårlig mange b -er og c -er.)
2. Lag en automat for språket som består av alle uttrykk som inneholder nøyaktig en a .
3. Lag en automat for språket som ikke inneholder a .
4. Lag en automat for språket som består av det tomme ordet (og ingen andre ord).
5. Lag en automat for det tomme språket.

Oppgave 2: DFA-er

Gitt alfabetet $A = \{a,b,c\}$. Lag DFA-er (deterministiske endelige automater) for følgende språk.

- a) Alle ord som inneholder maksimalt en b .
- b) Alle ord som inneholder to eller flere a -er.
- c) Alle ord som inneholder maksimalt en b og minst to a -er.

Oppgave 3

Gi tre forskjellige presentasjoner av automaten i pkt (2b) ("minst 2 a -er"):

- a. Nettverksdiagram
- b. Transisjonstabell
- c. Formelt med en funksjon.

Oppgave 4

La alfabetet $A = \{a, b, c\}$. Lag først en ikke-deterministisk automat for språkene og dernest en deterministisk automat for de samme språkene.

Hint til oppgave (b): Dette språket er komplementet til språket i pkt. (a).

- a) Ord som inneholder minst tre b -er på rad.
- b) Ord som ikke inneholder mer enn to b -er på rad.
- c) Ord hvor antall b -er er delelig med 3 eller antall a -er er delelig med 2 (eller begge deler).

Oppgave 5

Lag en DFA for språket som består av alle ord hvor antall b -er er delelig med 3 og antall a -er er delelig med 2.

Oppgave 6

Se på Tutorial på <http://www.jflap.org/tutorial/>. Jobb deg gjennom avsnittet "Regular expressions".

Oppgave 7

Vis at snittet av to regulære språk er regulært. (Hint: Bruk DFA-er!)

- SLUTT