

## DD1366: Ordinarie mästarprov i syntax 2025

Mästarprovet ska lösas individuellt och redovisas både skriftligt och muntligt. Inget samarbete är tillåtet, se vidare hederskodexen. Du ska alltså inte diskutera lösningar med någon annan fram till dess att alla muntliga redovisningar är avklarade. Inlämningarna plagiatgranskas. Det är inte tillåtet att använda verktyg baserade på AI/maskininlärning i arbetet med lösningarna eller under förberedelser till redovisningar. Exempel på förbjudna verktyg är ChatGPT, GitHub Copilot och Google Gemini.

Skriftliga lösningar på mästarprovet ska lämnas in som **en enda pdf** i Canvas. Skriv ditt namn och KTH-epostadress överst på framsidan i din inlämnade skriftliga lösning. Inuti din lösning, märk tydligt ut vilken av de tre mästarprovsdelarna du svarar på, eftersom varje del bedöms individuellt.

Alla som lämnar in lösningar på mästarprov i syntax har utrymme för ett mindre fel per mästarprovsdel. Nya studenter på kursen som redovisat labb S1 med bonus har utrymme för ytterligare ett mindre fel på delen om Automater och reguljära uttryck. Om antalet mindre fel är inom det tillåtna utrymmet på en mästarprovsdel så måste de rättas till under den muntliga redovisningen för att den delen ska kunna bli godkänd. Om en mästarprovsdel har fler mindre fel än de tillåtna eller något allvarligt fel innebär det att den mästarprovsdelen är underkänd. Mästarprovsdelar som godkänts efter muntlig redovisning kan tillgodoses under mästarprov i syntax under detta och nästkommande läsår. Till exempel kan du på mästarprovet i syntax under kursomgången våren 2026 tillgodoses dig delen om automater och reguljära uttryck om motsvarande del godkändes på detta mästarprov.

Det bästa är att använda LaTeX för att skapa texten för den skriftliga lösningen. De flesta som använder LaTeX gör det via ett webbgöransnitt som till exempel Overleaf. Om du bygger lokalt, tänk på att använda pdf<sub>l</sub>atex så att resultatet blir en pdf-fil. Assistenterna ska inte behöva bygga LaTeX-filer själva och framför allt inte behöva installera externa bibliotek. Det näst bästa är att använda Markdown som är enklare än LaTeX och ger nästan lika bra resultat. Precis som med LaTeX behöver du konvertera din Markdown-lösning till en pdf innan du skickar in, t ex via verktyget pandoc. Resterande: Om ni använder Microsoft Word eller liknande - tänk på att spara som pdf så att alla kan läsa texten som det är tänkt utan att ha Word installerat.

Läs uppgifterna mycket noga så att du inte råkar basera dina lösningar på en missuppfattning. Fråga en lärare på kursen om någon formulering är oklar.

Generella regler för mästarprov finns i kurs-PM. Tänk på att du behöver bli godkänd på alla delar av mästarprovet i syntax för att kunna bli klar med mästarprovsdelen i kursen, så överarbete inte en uppgift förrän du bedömer att du har rimliga lösningar på alla uppgifter.

Om du har lämnat in den skriftliga lösningen i tid så behöver du för att få godkänt även boka in dig på en muntlig redovisning som ges veckorna efter mästarprovets deadline. Redovisningstider kommer att anslås på Canvas i nära anslutning till deadline. Läs på dina lösningar inför den individuella muntliga redovisningen som kommer att ske efter deadline för någon i lärolaget. Redovisningen tar 10 minuter för dig, men de som bedömer dig har 5 minuters paus mellan varje par av redovisningar.

# 1. Grammatiker

Betrakta språket  $L$  som definieras av följande grammatik  $G$  skriven med matematisk notation:

$$\begin{array}{ll} S \rightarrow q & P \rightarrow S \\ S \rightarrow \mathbf{ay} P & P \rightarrow P * P \\ S \rightarrow \mathbf{ez} P & P \rightarrow ? P \\ S \rightarrow S * S & P \rightarrow \mathbf{xt} P \\ S \rightarrow ? S & P \rightarrow P \mathbf{uv} P \end{array}$$

där

- Startsymbolen för grammatiken är  $S$ .
  - $q$  består av en eller flera stora bokstäver från ASCII-tabellen.
  - Övriga slutsymboler är motsvarande sekvens av tecken i ASCII-tabellen (t ex “**ez**” för **ez**, “?” för ? och “\*” för \*).
- a) Visa att grammatiken  $G$  är tvetydig genom att göra två härledningar för en sträng  $u$  i  $L$  som resulterar i olika syntaxträd. Du behöver inte visa syntaxträden.
- b) Du ska nu göra en grammatik för ett språk som är en utökning av  $L$  med slutsymboler för vänsterparentes, “(”, och högerparentes, “)”.

Definiera en grammatik för det utökade språket i BNF som uppfyller följande krav:

- Startsymbolen för grammatiken är tydligt angiven.
- Grammatiken går att använda som grund för en parser konstruerad med rekursiv medåkning.
- Grammatiken är inte tvetydig.
- Alla grammatikens strängar går att skriva som en sekvens av tecken i ASCII-tabellen.
- Om alla vänsterparenteser och högerparenteser tas bort från en sträng som härletts i grammatiken så är resultatet en sträng i  $L$ .

Notera att du får (och uppmuntras att) namnge slutsymboler/tokens i din grammatik och definiera dem med reguljära uttryck.

- c) Motivera kort varför en hypotetisk implementation av en parser med rekursiv medåkning utgående från din grammatik i b) endast behöver inspektera en slutsymbol (token) framåt i en sträng för att avgöra vilken regel som ska användas för en given ickeslutsymbol.

- d) Ge ett exempel på en sträng  $v$  i ditt språk från b), och en sträng  $w$  från  $L$ , sådana att om alla vänsterparenteser och högerparenteser tas bort från  $v$  så erhålls  $w$ .
- e) Ge en härledning i din grammatik från b) som visar att strängen  $v$  från d) verkligen tillhör språket som din grammatik definierar. För att undvika långa upprepningar i härledningen kan du använda tydligt definierade förkortningar.
- f) Ge ett syntaxträd för strängen  $v$  i d) och motivera kort varför det bara finns ett enda syntaxträd för  $v$  i din grammatik. Det räcker att du använder namnen på dina tokens i trädet, dvs du behöver inte ange deras representation som sekvenser i ASCII.

## 2. Automater och reguljära uttryck

Vi betraktar språk över alfabetet  $\Sigma = \{a, b, c\}$  och definierar följande operationer och konstanter som returnerar sådana språk:

$$R(x, L_1) = \{w \mid xw \in L_1\} \text{ för } x \in \Sigma$$

$$S(L_1) = L_1^*$$

$$K(L_1, L_2) = \{uv \mid u \in L_1 \text{ och } v \in L_2\}$$

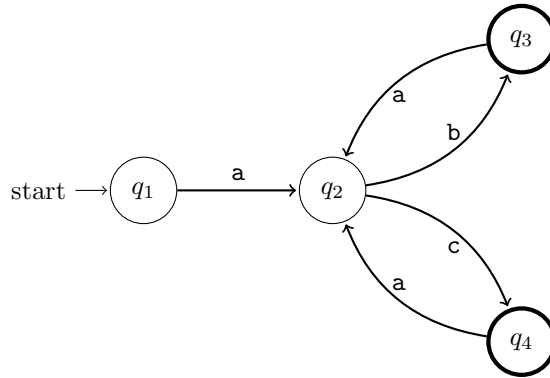
$$P(L_1, L_2) = \{u \mid u \in L_1 \text{ eller } u \in L_2\}$$

$$Z(L_1, L_2) = \{uv \mid cbu \in L_1 \text{ och } v \in L_2\}$$

$$B = \{ab\}$$

$$C = \{ac\}$$

- a) Låt  $M$  vara språket som känns igen av DFA:n över  $\Sigma$  med följande grafiska representation:



Uttryck språket  $Z(L, M)$  endast i termer av operationerna  $R$ ,  $S$ ,  $K$ ,  $P$ , och språken  $B$ ,  $C$  och  $L$ . Med andra ord, ge en definition av  $Z(L, M)$  med hjälp av  $B$ ,  $C$  och  $L$  som inte använder mängdnotation eller andra språkoperationer än  $R$ ,  $S$ ,  $K$  och  $P$ .

- b) Antag att språket  $L$  känns igen av DFA:n  $(Q, \Sigma, \delta, q, F)$ . Ge en matematisk definition av en DFA som känner igen språket  $\{u \mid cbu \in L\}$ . Med matematisk definition av en DFA menas en definition av en tupel  $(Q', \Sigma, \delta', q', F')$  och inte en grafisk representation.
- c) Antag att din DFA från b) är sådan att alla tillstånd i  $F'$  har en övergång för  $a$  till ett loopande tillstånd  $q_0 \in Q' \setminus F'$  (dvs  $q_0$  är ett icke-accepterande tillstånd som har övergångar till sig själv för alla  $x \in \Sigma$ ). Förklara kortfattat hur du med hjälp av denna DFA kan bygga en DFA som känner igen  $Z(L, M)$ .

### 3. Språkklasser

Låt  $\Sigma_1$  och  $\Sigma_2$  vara alfabet, och låt  $f$  vara en funktion från  $\Sigma_1^*$  till  $\Sigma_2^*$  sådan att  $f(uv) = f(u)f(v)$  för alla  $u \in \Sigma_1^*$  och  $v \in \Sigma_1^*$ . Antag att  $L$  är ett reguljärt språk över  $\Sigma_2$ . Antag vidare att  $L'$  är ett språk över  $\Sigma_1$  som definieras av en (kontextfri) grammatik. Vilken är den minsta språkklass som tagits upp i kursen som språket  $\{w \in \Sigma_1^* \mid f(w) \in L \text{ eller } w \in L'\}$  då med säkerhet tillhör? Motivera ditt svar noggrant.

**Ledning:** En översikt av de språkklasser som tas upp i kursen finns i ett Venn-diagram i slides för föreläsning 4 i syntax.