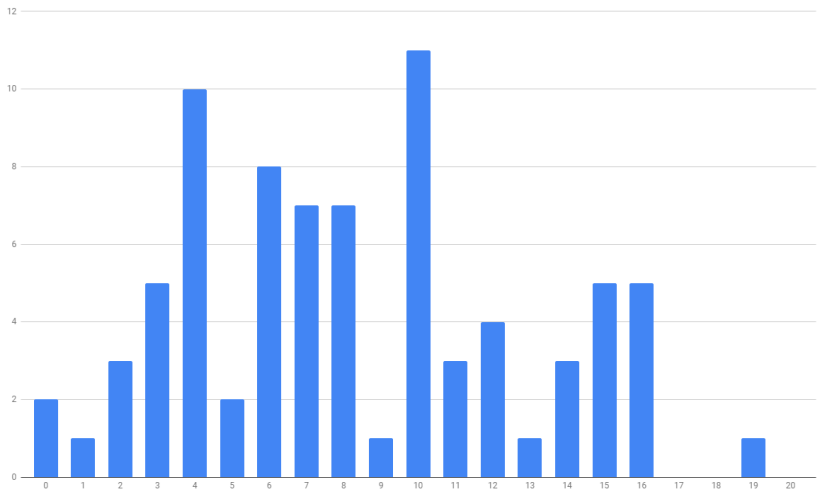
Examen Automaten en Berekenbaarheid – Bespreking

Prof. Dr. Bart Bogaerts
Mathieu Reymond

28 juni 2021

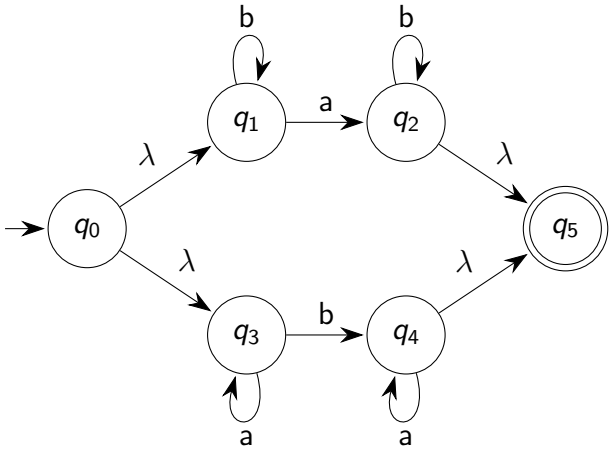
Net als vorig jaar was, omwille van Covid maatregelen, het examen weer volledig schriftelijk. De stijl van de vragen sloot erg aan bij die van de vorige jaren, en algemeen gezien had ik de impressie dat de studenten dit jaar iets beter voorbereid waren. Naar mijn inschatting was het examen ongeveer even moeilijk als dat van vorig jaar, maar het slaagpercentage lag wel beduidend hoger: 42% (toeval?) van de studenten die deelgenomen hebben (inclusief blanco indieningen) zijn geslaagd. De gemiddelde score is 8.3 De meerderheid van de punten ligt tussen 0 en 16, maar een enkeling scoort 19/20 (weliswaar gebruikmakend van de bonusvraag). Proficiat daarvoor! De onderstaande figuur bevat een overzicht van hoe de totale scores op het examen verdeeld zijn (de y-as bevat het aantal voorkomens van een resultaat).



De meest voorspelbare vragen (vraag 1 en vraag 3) werden het gemakkelijkst bevonden. Vraag 1 was een simpele NFA-DFA omzetting en vraag 3 was de te verwachten meerkeuze over sluitingen onder bepaalde operaties. Wie hierop voorbereid was kon gemakkelijk hoog scoren. Met deze twee vragen vielen reeds 8 punten te verdienen.
Vraag 4 en 5 waren de zwaardere theorievragen en hier wordt ook het laagst op gescoord. In vraag 4 was de opgave om iets meer detail uit een gezien bewijs te halen. In vraag 5 was de opgave – gelijkaardig aan een opgave van vorig jaar – om twee stellingen en hun bijhorend kort bewijsje te evalueren.
Vraag 2 lag tussenin qua resultaten. De stijl van de vraag was te verwachten, maar de eigenlijke opgaves vereisen best wat creativiteit. Heel wat studenten sprokkelen hier puntjes door goed in te schatten in welke klasses bepaalde talen zitten en een aantal "binnekoppers" op te lossen.

Vraag 1 (4 punten)

Gegeven de volgende NFA M_1, vind een NFA M_2 waarvoor $L(M_2) = \overline{L(M_1)}$.



Bespreking (vraag 1)

Dit was een vrij voordehandliggende vraag. De enige "valstrik" die er in zat was dat je bij een NFA niet gewoon finale states en niet-finale states mag wisselen. De gemakkelijkste manier om dit te doen was de NFA eerst naar een DFA omzetten met de gezien procedures.
Enkele veelgemaakte fouten:

- Sommige studenten wisselen de finale en de niet-finale states van een NFA. Dat werkt niet; ook niet na toevoegen van een trap state.
- Een aantal studenten hebben de *reverse* van de automaat berekend door pijlen van richting om te wisselen. Dat was niet de opgave.
- Een aantal studenten hebben geprobeerd om "op het zicht" een automaat te construeren voor het complement van de taal. Daar slopen echter steeds fouten in.
- Dan hebben heel wat studenten ergens een klein foutje gemaakt (bijvoorbeeld een kopieerfoutje, een state vergeten, ...) Als dat geen *systematische* fout was, kostte je dit slechts 1 punt.

Resultaten (vraag 1)

Een kleine 40% van de studenten deed dit correct en haalde de maximale score. Op deze vraag liggen de scores vrij extreem: heel wat nullen en (zoals gezegd) heel wat maximale scores. Enkele studenten verliezen een puntje met een klein foutje hier of daar, maar algemeen gezien wordt er naar verwachting gescoord op deze vraag.

Vraag 2 (6 punten)

Zijn de volgende talen (L_1, L_2, en L_3) over het vocabularium $\{a, b, c, d\}$ regulier? Zijn ze context-vrij? Zijn ze context-sensitief? Zijn ze recursief? Zijn ze recursief enumereerbaar? Toon al jouw antwoorden aan (voor positieve antwoorden bijvoorbeeld door een grammatica of automaat van de juiste soort op te stellen of door geziene eigenschappen te gebruiken; voor negatieve antwoorden bijvoorbeeld door een pompstelling, een reductie van het halting probleem, of andere geziene eigenschappen te gebruiken). Wanneer je Turing machines construeert mag je een hoogniveau beschrijving van het gedrag geven zonder de transitiefunctie in detail uit te schrijven. Je mag dan gebruik maken van bouwblokken die simpele operaties doen (bijvoorbeeld een TM die optelling of vermenigvuldiging doet).

1. De taal
$$L_1 = \{a^x b^y c^y d^y \mid x, y \in \mathbb{N} \wedge x \geqslant 1\} \cup L(b^* c^* d^*)$$

2. Zij e een encoderingsfunctie van Turing Machines als strings over $\{a, b, c, d\}$ (bijvoorbeeld een variant van diegene die we in de les gezien hebben met a, b, c, en d). De taal
$$L_2 = \left\{ e(M) \mid \begin{array}{c} M \text{ is een Turing machine met input alfabet } \{0,1\} \\ \text{die oneindig lang loopt op de input } 1^{42}0^{42} \end{array} \right\}$$

3. De taal $L_3 = L(G_3)$, waarbij G_3 de volgende grammatica met startsymbool S is:
$$S \to aA \mid \lambda$$
$$A \to Sb \mid \lambda$$

Bespreking (vraag 2)

1. De eerste vraag was hier de moeilijkste van de drie. Het juiste antwoord is: deze taal is *niet context-vrij* (en dus ook niet regulier) maar *wel context-sensitief* (en dus ook...).

   - Argumenteren dat ze *context-sensitief* is, valt wel mee. Je kan een TM maken die *als* de input string met een aantal a's begint systematisch 1 $b$tje, 1 $c$tje, en 1 $d$tje verwijdert en op het einde de check doet of het aantal gelijk was. Of, misschien zelfs nog eenvoudiger... je kan zeggen dat de dit gelijk is aan
   $$(L(aa^*) \cdot \{b^y c^y d^y \mid y \in \mathbb{N} \wedge y \geqslant 1\}) \cup L(b^* c^* d^*)$$
   Waarbij $L(aa^*)$ en $L(b^* c^* d^*)$ regulier zijn en we voor $\{b^y c^y d^y \mid y \in \mathbb{N} \wedge y \geqslant 1\}$ quasi letterlijk een context-sensitieve grammatica gezien hebben. Context-sensitieve talen zijn gesloten onder unie en concatenatie, dus het resultaat is opnieuw context-sensitief.
   - Argumenteren dat ze *niet context-vrij* is, is moeilijker. De taal L_1 voldoet immers *wel* aan de pompstelling voor context-vrije talen (je kan ongeveer altijd de $a$tjes pompen). De truuk om deze vraag op te lossen is reguliere intersectie gebruiken. Immers:
   $$L_1 \cap L(ab^* c^* d^*) = \{ab^y c^y d^y \mid y \in \mathbb{N}\}.$$
   Op deze taal kan je de pompstelling *wel* loslaten, op de gebruikelijke manier (bijvoorbeeld op de string $ab^m c^m d^m$. Het argument is dan: de a kan niet opgepompt

worden, want anders kan je $i = 0$ nemen en val je buiten de taal (de a verdwijnt dan). Voor oppompen van bs, cs, en ds, is het argument identiek aan wat we in de les zagen.

Opmerking: Herinner je je nog mijn announcement op canvas met als titel "Interessante Clinic (pompstelling)". Daarin verwijs ik naar een niet-reguliere taal die *wel* voldoet aan de pompstelling voor reguliere talen. Dit voorbeeld is heel gelijkaardig.

Opmerking: Het is niet correct om te zeggen: als L_a *niet* context-vrij is, en L_b is regulier, dan is $L_a \cup L_b$ ook *niet* context-vrij. Closure under union werkt maar in 1 richting!

Opmerking: Heel wat studenten (minstens negentig procent) vergeten bij het toepassen van pompstellingen met een string $w = ab^m c^m d^m$ dat het wel eens zou kunnen dat $v = a$ en $y = \lambda$. Wat gebeurt er met zo'n string... Wel, dan zal voor elke i de string weer in L_1 zitten, maar duidelijk niet in $L_1 \cap L(ab^*c^*d^*)$. Dit is waarom de reguliere intersectie zo belangrijk is!

2. Bij het tweede deel was het juiste antwoord: deze taal zit in *geen enkel van de geziene klasses*. Dat is een straf resultaat. Om de oplossing te verkrijgen moest je twee zaken aantonen.

   - L_2 *is niet recursief*. Dit bewijs was vrij eenvoudig: de reductie van het halting probleem is quasi gelijk aan die van Halting naar Blank-tape halting (behalve dat ipv de tape te wissen, je nu de tape wist en de string $1^{42}0^{42}$ op je tape schrijft *EN* dat er nadien nog een negatie van de output gebeurt. Immers, een string $e(M)$ zit in L_2 als de machine *niet* stopt op die bepaalde string. Als je deze reductie goed had, kreeg je al het merendeel van de punten op deze vraag.
   - L_2 *is ook niet recursief enumereerbaar*. Dit is wat moeilijker. En in ons argument zullen we het vorig resultaat (L_2 is niet recursief) gebruiken. Dat verklaart waarom we dat eerst doen. Het argument is kort, maar minder gemakkelijk te vinden. Namelijk: je kan aantonen dat $\overline{L_2}$ *wel* recursief enumereerbaar is. Hoe doe je dat? Wel... je maakt een machine die checkt of zijn input een geldige encoding is (i.e., van de vorm $e(M)$ voor een zekere M). Als dat niet het geval is, zit de input *zeker* in de taal en antwoord je ja. Als dat wel het geval is, simuleer M op de gegeven string. Als de machine stopt, zit $e(M)$ in $\overline{L_2}$ en is het antwoord dus ja. Als de machine niet stopt, dan loop je oneindig. Hoe helpt ons dit nu verder? Wel... we hebben in de les gezien dat als een taal en zijn complement RE zijn, dat de taal recursief is. Maar... dat is niet zo. Dus kan L_2 niet recursief enumereerbaar zijn!

   Opmerking De reductie hier was een vrij gemakkelijke reductie. In feite is ze quasi identiek aan degene van Halting naar Blank-tape-halting. Maar, reducties worden door veel studenten moeilijk bevonden en heel wat mensen hebben deze vraag gewoon leeg gelaten.

   Opmerking: Als je een reductie maakt, bijvoorbeeld van het halting probleem naar een ander probleem, dan moet je ervoor zorgen dat *voor elke input*, je het juiste antwoord op het halting probleem kan produceren. Verschillende mensen ontwikkelen een (foute!) oplossing van de vorm "We doen een reductie van het halting probleem naar het probleem dat overeenkomt met L_2. Als de input M, w is en w is niet gelijk aan de string $1^{42}0^{42}$, dan verwerpen (of accepteren) we sowieso, anders gebruiken we de machine voor

L_2. Dit is niet correct. Op deze manier is je antwoord *geen* oplossing voor het halting probleem.

Opmerking: Je kan niet bewijzen dat een taal Recursief Enumereerbaar is aan de hand van het halting probleem. Sommige mensen proberen dit. Je kan het halting probleem gebruiken (mbv een reductie) om aan te tonen dat de taal niet recursief is, maar daar volgt uiteraard niet uit dat ze *wel* recursief enumereerbaar is. Immers... we hebben in de les talen gezien die niet recursief, noch recursief enumereerbaar zijn. Je mag dus zeker ook niet zeggen "de taal is recursief enumeraarbaar omdat ze niet beslisbaar is". Immers: elke beslisbare taal is recursief enumereerbaar; als ook elke niet-beslisbare (niet-recursieve) taal RE zou zijn, dan waren *alle* talen RE!

3. Deze vraag was een echte weggever.

   - *De taal is context-vrij, want: er is een context-vrije grammatica voor gegeven.* Deze eenvoudige zin leverde je reeds bijna de helft van de punten van deze deelvraag op!
   - De taal die hier gedefinieerd wordt is $L_3 = \{a^x b^y \mid x = y \text{ of } y = x-1\}$. Deze taal is duidelijk niet regulier. De toepassing van de pompstelling is dezelfde als altijd, neem bijvoorbeeld $w = a^m b^m$. Als je minstens 1 a oppompt val je buiten de taal.

   Opmerking: Wie bij deze vraag een PDA tekende of eender welke andere manier verzon om aan te tonen dat ze context-vrij is, kreeg minder punten dan wie gewoon opmerkte: "dit is een context-vrije grammatica". Het feit dat je de moeite doet om er een PDA voor te maken getuigt van een gebrek aan inzicht in de materie. Bovendien is het argument "hier is een PDA voor deze taal" heel erg imprecies. Zo goed als niemand gaf er een argument bij *waarom* die PDA juist is.

   Opmerking: Er zijn ook een aantal mensen die de gegeven grammatica hebben omgevormd naar Chomsky normaalvorm om dan te concluderen dat de taal context-vrij is, maar daar is helemaal geen reden toe. Dat is naast de kwestie.

   Opmerking: De enige valkuil in deze taal was de vorm van de grammatica. Sommige mensen antwoorden "dit is een reguliere grammatica, dus de taal is regulier". Dat klopt echter niet: sommige regels zijn linkslineair, andere rechtslineair; in een reguliere grammaticas hebben alle regels dezelfde vorm. Andere mensen antwoorden "dit is een contextvrije grammatica" *dus* de taal is niet regulier. Dat is ook niet correct. Immers, een reguliere taal kan ook gedefinieerd worden door een niet-reguliere grammatica. Het enige dat je weet is dat er een reguliere grammatica voor elke reguliere taal *bestaat*. Niet dat elke grammatica die de taal definieert regulier zal zijn.

Opmerking: Heel wat studenten verspillen bij dit soort vraag enorm veel tijd door obvious zaken aan te tonen. Bijvoorbeeld bij vraag 1 door aan te tonen dat $L(b^*c^*d^*)$ regulier is door er een DFA voor tegeven. Dit moet je niet aantonen!! Je hebt een reguliere expressie gegeven. Ook bij L_3 valt er niets aan te tonen aan het context-vrij zijn. Je kan dit gewoon opmerken. De gegeven grammatica is context-vrij, dus de taal ook.

Opmerking: Als je al hebt aangetoond dat een taal niet context-vrij is, hoef je heus geen argument meer op te schrijven dat het ook niet regulier is. Je mag gewoon zeggen "daaruit volgt meteen dat de taal ook niet regulier is". Als je daar *toch* ook nog een apart bewijs voor

geeft, dan krijg ik de indruk dat je *de cursus niet goed kent*. Dat kan dus alleen maar in je nadeel spelen!

Opmerking: Het is nuttig om wat je *weet* of sterk vermoedt op te schrijven. Vraag 2 blanco laten is een zeer slecht idee. Zelfs in geval van tijdsnood, schrijf je best op wat je wel weet: bijvoorbeeld: "L_1 is niet context-vrij" (zonder uitleg). Dit levert je al een beetje punten op. Sterker nog: als je gewoon voor alle talen (correct uiteraard) uitviste tot welke klasses ze al dan niet behoren, kreeg je reeds een derde van de punten op deze vraag! Dat zijn twee snel verdiende punten.

Opmerking: Bij het toepassen van de pompstelling hoef je niet steeds exchaustief alle mogelijke combinaties van waar de "y" kan liggen af te gaan. Bijvoorbeeld, als je m gegeven hebt, en je kiest $w = a^{m+1}b^m$, dan weet je dat er een opsplitsing bestaat van w als xyz met $|xy| \leqslant m$. Heel wat studenten schrijven dit letterlijk zo op: dat is prima. Maar vervolgens volgt er dan

- Als y alleen uit as bestaat, dan ...
- Als y uit as en bs bestaat, dan ...
- Als y alleen uit bs bestaat, dan ...

Die laatste twee gevallen kunnen al niet aangezien $|xy| \leqslant m$.

Resultaten (vraag 2)

Wie mijn examens van vorige jaren heeft bestudeerd, wist dat dit type vraag er zat aan te komen. Ze oplossen vereist dat je een aantal zaken goed onder de vingers hebt (toepassen pompstelling bijvoorbeeld), maar ook dat je goed en snel kan oordelen welke talen al dan niet tot welke klasse behoren.
Er waren drie deelvragen. Elke deelvraag woog even zwaar door. Binnen elke deelvraag waren er telkens twee te bewijzen resultaten (een positief en een negatief). Deze hadden niet steeds hetzelfde gewicht. Op de eerste en het derde deelvraag was telkens ongeveer 50% van de studenten geslaagd. Bij deelvraag 1 waren de punten redelijk gespreid, bij deelvraag 3 waren de punten extremer (veel mensen halen hier de maximale score, maar veel mensen halen hier ook een nul op). Deelvraag 2 was duidelijk de moeilijkste. Daar was slechts 20% van de studenten op geslaagd (deels aangezien heel wat studenten deze gewoon blanco hebben gelaten! – uit schrik voor een reductie?)
Je totale score op deze vraag ligt iets hoger dan het gemiddelde op elk van de deelvragen: ik ben er bij het verbeteren vanuit gegaan dat ik wel 1 foutje (uit 6 te bewijzen resultaten) door de vingers wil zien. 40% van de studenten is uiteindelijk geslaagd op deze vraag. De punten liggen sterk verspreid doorheen het hele spectrum 0–20.

Vraag 3 (4 punten)

Deze vraag bestaat uit 12 meerkeuzevragen. Er wordt giscorrectie toegepast. Elk juist antwoord levert je een derde van een punt op. Voor elk fout antwoord verlies je een derde van een punt.

Beschouw de volgende klasses talen:

- `Reg`: de klasse van reguliere talen
- `CF`: de klasse van contextvrije talen
- `dCF`: de klasse van deterministische contextvrije talen
- `Rec`: de klasse van recursieve talen
- `RE`: de klasse van recursief enumereerbare talen
- `CS`: de klasse van contextsensitieve talen
- `IrReg`: de klasse van niet-reguliere talen (talen die niet regulier zijn)

En beschouw over elke klasse $\mathcal{C}$ de volgende beweringen:

- **RegInt**: de klasse talen is gesloten onder het nemen van doorsnede met een reguliere taal. Dit wil zeggen, als $L_1 \in \mathcal{C}$ en $L_2 \in \texttt{Reg}$, dan ook $L_1 \cap L_2 \in \mathcal{C}$.
- **Star**: de klasse talen is gesloten onder het nemen van Kleene-star. Dit wil zeggen: als $L \in \mathcal{C}$, dan is ook $L^* \in \mathcal{C}$.
- **Reverse**: de klasse talen is gesloten onder de reverse operatie, dit wil zeggen: als $L \in \mathcal{C}$, dan is ook $L^R \in \mathcal{C}$.
- **SymDiff**: de klasse talen is gesloten onder symmetrisch verschil, dit wil zeggen: als $L_1 \in \mathcal{C}$ en $L_2 \in \mathcal{C}$, dan is ook $L_1 \triangle L_2 \in \mathcal{C}$, waarbij $L_1 \triangle L_2 = (L_1 \backslash L_2) \cup (L_2 \backslash L_1)$. Met andere woorden: $L_1 \triangle L_2$ bevat alle strings die in exact 1 van beide talen zitten.

Vul voor alle **lege** vakjes in de volgende tabel in of de bewering waar is voor de overeenkomstige klasse talen of niet.
De vakjes gemarkeerd "XXXXXXX" moet je **niet** invullen. Sommige daarvan zijn waar, andere niet.

	`Reg`	`CF`	`dCF`	`Rec`	`RE`	`CS`	`IrReg`
RegInt	WAAR	XXXXXXX	XXXXXXX	WAAR	XXXXXXX	XXXXXXX	VALS
Star	XXXXXXX	XXXXXXX	XXXXXXX	WAAR	WAAR	XXXXXXX	XXXXXXX
Reverse	XXXXXXX	XXXXXXX	XXXXXXX	XXXXXXX	WAAR	WAAR	WAAR
SymDiff	WAAR	VALS	XXXXXXX	WAAR	VALS	XXXXXXX	XXXXXXX

Bespreking (vraag 3)

Deze was een redelijk te verwachten vraag (gezien de examens van vorige jaren). In vergelijking met de meerkeuzevragen van vorige jaren was deze – volgens mij – iets gemakkelijker. Je kon de antwoorden als volgt vinden:

- De vreemde eend in de bijt waren de irreguliere talen (talen die **niet** tot een bepaalde klasse behoren. Die kon je als volgt oplossen:
  - Ze zijn duidelijk niet gesloten onder reguliere doorsnede (want $\varnothing$ is bijvoorbeeld een reguliere taal en de doorsnede van eender welke taal met $\varnothing$ is.
  - Aangezien de reverse van een reguliere taal opnieuw regulier is, is ook de reverse van een irreguliere taal irregulier (contrapositie)

  Deze antwoorden zijn blauw in de tabel.
- Voor reguliere intersectie, star closure, en reverse, kon je rechtstreeks geziene constructies uit de les veralgemenen. Voor reguliere intersectie: een DFA simuleren tegelijkertijd met een relevante automaat van het juiste type. Voor star closure, door te redeneren op de grammatica, voor reverse: ook door te redeneren op de grammatica. De antwoorden die je zo kon vinden zijn rood in de tabel.
- Voor recursieve talen en star closure ($L = L_1^*$) werkt de bovenstaande constructie nog niet. Daar moeten we een andere oplossing voor verzinnen. Wat je hier kan doen is je input string willekeurig (mbv een niet-deterministische TM) in stukken kappen, en op elk stuk een machine simuleren die checkt of een string in L_1 zit. Dit antwoord is groen.
- Wat nog overblijft is symmetrisch verschil. Hier moest je beseffen
  - Dat
    $$L_1 \triangle L_2 = (L_1 \backslash L_2) \cup (L_2 \backslash L_1) = (L_1 \cap \overline{L_2}) \cup (L_2 \cap \overline{L_1})$$
    en dus dat als een klasse gesloten is onder unie, doorsnede, en complement, ze ook gesloten is onder symmetrisch verschil (`Reg` en `Rec`)
  - Dat $\Sigma^* \triangle L = \overline{L}$ en dus dat als een klasse niet gesloten is onder complement, ze ook niet gesloten is onder symmetrisch verschil (`CF` en `RE`).

  Deze zijn oranje gemarkeerd in de tabel.

Resultaten (vraag 3)

De score op deze vraag lag hoger dan bij de meerkeuzevragen van vorig jaar (waarschijnlijk klopt de hypothese dat de opgave iets gemakkelijker was). 58% van de studenten is geslaagd op deze vraag. De grootste groep studenten scoort tussen de 10 en 15 op 20 op deze vraag. De meeste fouten werden gemaakt bij:

- Symmetrisch verschil (in het bijzonder van recursieve talen)
- Reguliere intersectie (irreguliere talen), waar nu de vraag was: is het zo dat de doorsnede van een reguliere en een irreguliere taal *gegarandeerd* irregulier is?

Vraag 4 (3 punten)

In de les hebben we gezien dat het complement van een context-vrije taal niet per se context-vrij is. Het bewijs ging ruwweg als volgt. We weten dat de unie van twee context-vrije talen zelf ook context-vrij is, en we beschouwen de talen

$$L_1 = \{a^x b^x c^y \mid x, y \in \mathbb{N}\}$$
$$L_2 = \{a^x b^y c^y \mid x, y \in \mathbb{N}\}$$

Dan is $L_1 \cap L_2 = \{a^x b^x c^x \mid x \in \mathbb{N}\}$. We weten dat deze taal niet context-vrij is. Maar... doorsnede, unie en complement zijn aan mekaar gerelateerd met de wetten van de Morgan. Dus... kan het complement van een context-vrije taal niet context-vrij zijn.
In dit bewijs worden (impliciet) heel wat complementen genomen (drie, om precies te zijn). Jammer genoeg is dit bewijs niet echt constructief. Het zou veel overtuigender als je me één concrete taal zou kunnen geven die zelf context-vrij is, maar waarvan het complement niet context-vrij is? Kan je **uit dit bewijs** zo'n taal halen (dus: een van de drie voorkomens van complement)? Beargumenteer waarom je antwoord juist is.

Bespreking

Voor wie de lesopnames actief[1] had bekeken, was deze vraag een weggevertje. Ik had namelijk toen we deze stelling in de les zagen expliciet gesuugereerd om je deze vraag eens te stellen (Na 1u41min in de opname). Wie dit vak goed heeft bijgehouden, had deze examenvraag dus al eens op voorhand gemaakt.
Zelfs zonder die "hint", is het antwoord niet ver te zoeken:

1. De relevante wet van de Morgan is $L_1 \cap L_2 = \overline{\overline{L_1} \cup \overline{L_2}}$.

2. Hierin worden (zoals in de opgave vermeld) drie complementen genomen. Er zijn dus drie kandidaat talen die *mogelijks* context-vrij zijn, maar hun complement *mogelijks* niet: L_1, L_2, en $\overline{L_1} \cup \overline{L_2}$. Eén van deze drie moet het antwoord zijn. De eerste vraag die je dus moet oplossen is: is $\overline{L_1}$ context-vrij of niet? Als die vraag opgelost is, ben je eigenlijk al klaar! Namelijk: het geval $\overline{L_2}$ is symmetrisch. Als deze niet CF zijn, is L_1 een goed antwoord op de vraag. Als deze *wel* CF zijn, dan is $\overline{L_1} \cup \overline{L_2}$ het antwoord op de vraag (zie lager).

3. We weten dat

$$\begin{aligned}\overline{L_1} &= \{a^x b^z c^y \mid x, y, z \in \mathbb{N}, x \neq z\} \cup \overline{L(a^* b^* c^*)} \\ &= \{a^x b^y \mid x \neq y\} \cdot L(c^*) \cup \overline{L(a^* b^* c^*)}\end{aligned}$$

   We weten dat $\{a^x b^y \mid x \neq y\}$ context-vrij is (gezien in de les, maar ook: een context-vrije grammatica opstellen is niet moelijk eens je beseft dat er twee opties zijn: ofwel is x groter dan y of omgekeerd). Aangezien context-vrije talen gesloten zijn onder concatenatie en unie is $\overline{L_1}$ dus ook context-vrij.

4. Analoog is $\overline{L_2}$ context-vrij

[1] Met "actief" bedoel ik: wanneer die clown vooraan in de aula zegt "denk daar zelf eens over na; ik ga het je niet vertellen", dit effectief ook doen.

5. Aangezien context-vrije talen gesloten zijn onder unies, is dan ook $\overline{L_1} \cup \overline{L_2}$ context-vrij.
6. In de opgave stond reeds dat $L_1 \cap L_2$ niet context-vrij is.
7. Dus.... (omwille van de wet van de Morgan van hoger) is $\overline{L_1} \cup \overline{L_2}$ een context-vrije taal waarvan het complement niet context-vrij is.

Opmerking: Heel wat mensen hebben geschreven dat $\overline{L_1}$ gelijk is aan

$$\{a^x b^z c^y \mid x, y, z \in \mathbb{N}, x \neq z\}$$

Dit is technisch gezien *niet correct*. Bijvoorbeeld de string *abcabc* zit in $\overline{L_1}$ maar niet in bovenstaande taal. Hier heb ik echter geen punten voor afgetrokken (dit is een detail).

Opmerking: Sommige mensen hebben de volledige toepassing van de pompstelling opgeschreven. Dat is niet nodig. Vuistregel: als je bij mijn examens meer dan 1 pagina voor een opgave schrijft, ben je te veel aan het schrijven!

Opmerking: Een alternatief argument in stap 3 waarom $\{a^x b^y c^z \mid x \neq y\}$ context-vrij is, is "je kan met een PDA het aantal *a*tjes bijhouden en matchen tov het aantal *b*tjes, terwijl je ze ziet. Nadien accepteer je een willekeurig aantal *c*tjes". Dit argument is op een vrij hoog niveau, maar is voor mij voldoende. Je hoeft hier geen concrete PDA uit te tekenen. Het volstaat dat je me kan overtuigen dat je weet waarom het complement van L_1 context-vrij is.

Opmerking: een nog korter (maar minder voordehandliggend) argument voor stap 3 zou zijn: L_1 is een deterministiche context-vrije taal, dus zijn complement is ook context-vrij.

Resultaten (vraag 4)

Ondanks het feit dat deze vraag in zekere zin "aangekondigd" was, zijn de resultaten toch erg tegenvallend. Van alle vijf de vragen, heeft deze vraag het grootste aantal nullen als resultaat (bijna 40% van de studenten - dit is inclusief de blanco ingediende kopijen uiteraard). Op deze vraag is 29% van de studenten geslaagd; wie geslaagd is op deze vraag haalt wel bijna altijd erg hoge resultaten (16 of hoger).

Vraag 5 (3 punten)

In de les hebben we gezien dat het quotient van een reguliere taal met een andere reguliere taal ook steeds regulier is. Dus, als L_1 en L_2 regulier zijn, dan is

$$L_1/L_2 = \{w \mid \exists w_2 \in L_2 : ww_2 \in L_1\}$$

opnieuw regulier.
Beschouw nu eens de volgende beweringen. Geef voor elk van de bewering aan of ze juist of fout is. Bespreek de bewijzen. Worden er fouten gemaakt? Zoja, waar? Kan je ze verbeteren? Zijn er stappen die misschien wat te kort door de bocht gaan? Kan je aanvullen?

STELLING 1: *Als L_1 een reguliere taal is en L_2 een recursief enumereerbare taal, dan is L_1/L_2 een context-sensitieve taal.*
BEWIJS VAN STELLING 1: Aangezien L_1 regulier is, bestaat er een NFA voor. Deze NFA is altijd deterministisch. We noemen deze automaat $M_1 = \langle Q, \Sigma, \delta, q_0, F\rangle$. Aangezien L_2 recursief enumereerbaar is, bestaat er een Turing Machine M_2 die L_2 accepteert. Beschouw nu de volgende verzameling states

$$F' = \{q \in Q \mid \exists y : M_2 \text{ accepteert } y \text{ en } \delta^*(q, y) \in F\}$$

Deze verzameling is goed gedefinieerd. Dankzij een vleugje magie zien we nu dat $\delta^*(q_0, x) \in F'$ als en slechts als er een y is zodat $xy \in L_1$ en $y \in L_2$. Dus de automaat $\langle Q, \Sigma, \delta, q_0, F'\rangle$ accepteert het quotient van L_1 met L_2. Dit quotient is bijgevolg zeker context-sensitief.

STELLING 2: *Als L_1 regulier is en L_2 context-vrij is, dan is L_1/L_2 niet per se opnieuw context-vrij.*
BEWIJS VAN STELLING 2: Dit is een bewijs uit het ongerijmde. Veronderstel dat het wel zo zijn dat voor elke reguliere taal L_1 en context-vrij taal L_2, L_1/L_2 context-vrij is. In de cursus hebben we gezien dat er talen bestaan die context-vrij zijn, maar waarvan het complement niet context-vrij is (zie ook vraag 4). Bovendien weten we dat voor elk vocabularium Σ, Σ^* regulier is. Door $L_1 = \Sigma^*$ te nemen voor een welgekozen vocabularium Σ, zouden we krijgen dat het complement van elke context-vrije taal opnieuw context-vrij is, en dat geeft een contradictie.

BONUSVRAAG (2 bonuspunten): *Met deze vraag kan je twee bonuspunten verdienen. Als je ze perfect hebt, kan je dus 22 punten scoren op dit examen! (wie meer dan 20 scoort, krijgt wel slechts 20 als uiteindelijk resultaat)* Wat is de sterkste ware stelling die je kan verzinnen, geinspireerd op de twee voorgaande beweringen (en hun bewijs). Kan je ze ook bewijzen?

Bespreking (vraag 5)

In deze vraag zaten een aantal subtiliteiten. Ik bespreek eerst **STELLING 1**.

- Zoals heel wat studenten opmerkten klopt de zin "deze NFA is altijd determinstisch" niet. Niet elke NFA is deterministisch. Het is wel altijd zo dat er ook een DFA bestaat. Deze zin was dus fout, maar de rest van het bewijs wordt hier niet door aangetast.

- De verzameling F' is inderdaad goed gedefiniëerd. De conditie "$\exists y : ...$" is ofwel waar of niet. F' is een deelverameling van Q. Hoe nuttig deze verzameling is, is op dit moment nog niet duidelijk, maar het is wel degelijk een deelverzameling van Q. Merk hier ook even op, dat F' nog wat eenvoudiger te schrijven viel:

  $$F' = \{q \in Q \mid \exists y : y \in L_2 \text{ en } \delta^*(q, y) \in F\}$$

  (over dit punt "goed gedefinieerd" moest je niet per se iets zeggen)

- De volgende zin bevat "een vleugje magie". Hier kon je nog wat meer details geven; wanneer uitgewerkt zie je dat de bewering inderdaad klopt. Dit is een herformulering van de definitie van F'.

- We krijgen dus een automaat L_1/L_2 accepteert. Deze automaat is een DFA! De taal L_1/L_2 is dus regulier, en bijgevolg zeker ook context-sensitef. Er is op dit punt **geen nood** om een LBA op te stellen die de taal ook accepteert. We weten immers dat alle reguliere talen ook context-sensitief zijn.

Opmerking: In dit bewijs hebben we nauwelijks gebruik gemaakt van eigenschappen van L_2. Als we de "eenvoudigere" definitie van F' nemen, hebben we zelfs helemaal geen gebruik gemaakt van eigenschappen van L_2! Bovendien hebben we reeds bewezen dat L_1/L_2 niet alleen context-sensitief, maar ook *regulier* is. In feite hebben we dus bewezen dat:

STELLING 3: *Als L_1 een reguliere taal is en L_2 een taal is, dan is L_1/L_2 een reguliere taal.*

Dat is een straf resultaat! Wie dit als antwoord gaf bij de **BONUSVRAAG**, kreeg twee bonuspunten.

Opmerking: Sommige studenten geraken in de war met berekenbaarheid: De definitie van F' verwijst naar "$y \in L_2$" en dit kunnen we niet beslissen! Dat is best lastig. Mogen we F' dan zo wel definiëren. Het antwoord is als volgt: wiskundig is de set F' perfect in orde. Dit is een deelverzameling van Q en voor elke state is het goed gedefinëerd of hij er in zit of niet. Het probleem "zit een state in F'" gaan we echter niet kunnen beslissen. Heel concreet betekent dit dus dat we *zeker zijn* dat er een DFA *bestaat* voor L_1/L_2, maar dat we hem niet kunnen *berekenen*. Maar, dit is geen probleem, om aan te tonen dat een taal regulier is, volstaat het aan te tonen dat er een DFA voor bestaat, we hoeven daarvoor niet te weten welke automaat het precies is.

De bovenstaande STELLING 3 (waarvan we ondertussen weten dat ze klopt) is meteen ook in tegenspraak met **STELLING 2**. Dus: stelling 2 was verkeerd. Waar ging het mis in het bewijs? Wel, er wordt in het bewijs schaamteloos verondersteld dat L_1/L_2 en $L_1 \backslash L_2$ hetzelfde zijn. Dit is uiteraard niet zo! De symbolen lijken misschien op elkaar, maar die twee zijn hoe

dan ook niet gelijk. Sterker nog: Σ^*/L_2 zal *altijd* gelijk zijn aan Σ^* of aan $\varnothing$ (het laatste enkel als $L_2 = \varnothing$). Dat is helemaal niet gelijk aan het complement van L_2; dit bewijs is dus compleet fout.

Resultaten (vraag 5)

De resultaten op deze vraag liggen erg laag. Het grootste deel van de studenten scoort 0–4 op 20 op deze vraag. Er zijn een aantal mensen die een foutje in een bewijs spotten en daardoor concluderen dat de stelling niet waar is. Dat klopt uiterard niet. Er zijn wel een paar enkelingen die hoog scoren en een vijftal mensen die de bonusvraag goed (gelijkaardig aan Stelling 3 of iets zwakker) beantwoorden en hier dus nog wat "gratis" puntjes sprokkelen. 28% van de studenten is geslaagd op deze vraag.