

EXTRA TENTAMEN COMPUTABILITY

Dinsdag 19 december 2025, 09.00 - 12.00 uur

Dit tentamen bestaat uit vijf opgaven, waarbij steeds tussen [en] staat hoeveel punten er ongeveer mee te verdienen zijn. In totaal zijn er 100 punten te verdienen. Wanneer er bij een vraag om uitleg, motivatie of toelichting gevraagd wordt, is het belangrijk om die ook te geven.

1. [28 pt] Laat

$$L = \{a^i b^j a^{i+j} \mid 0 \leq i < j\}$$

- (a) Geef de eerste vier elementen in de canonieke volgorde van L .
- (b) Teken een gewone (deterministische, 1-tape) Turing machine T , zó dat $L(T) = L$.
Leg ook duidelijk uit hoe T werkt. Als je voor T gebruik wilt maken voor componenten, dan zul je die ook moeten tekenen.

2. [16 pt]

- (a) Laat T_1 en T_2 Turingmachines zijn, die respectievelijk talen L_1 en L_2 over hetzelfde alfabet Σ accepteren.

Beschrijf de werking van een **niet-deterministische** Turingmachine T om $L_1 L_2$ te accepteren.

Hint: Vraag je af wanneer een invoer x in $L_1 L_2$ zit.

Omdat we van L_1 en L_2 niets anders weten dan dat $L_1 = L(T_1)$ en dat $L_2 = L(T_2)$, zal T gebruik moeten maken van componenten T_1 en T_2 , of varianten daarvan. Beschrijf de werking van T en eventueel benodigde aanpassingen van T_1 en T_2 duidelijk, op het niveau van de functionaliteit (je hoeft geen beschrijving op het niveau van transities te geven).

N.B.: T moet daadwerkelijk de kracht van niet-determinisme gebruiken bij het accepteren van $L_1 L_2$.

- (b) Beredeneer dat je niet-deterministische Turingmachine T uit onderdeel (a) inderdaad (precies) $L_1 L_2$ accepteert.

3. [18 pt] Laat (opnieuw)

$$L = \{a^i b^j a^{i+j} \mid 0 \leq i < j\}$$

Geef een *unrestricted grammar* G , zó dat $L(G) = L$.

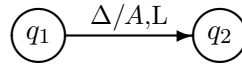
Leg uit wat de functie is van de diverse variabelen en producties in G .

4. [15 pt] In het bewijs van Stelling 8.14 in het boek wordt beschreven, hoe je bij een willekeurige Turingmachine $T = (Q, \Sigma, \Gamma, q_0, \delta)$ een unrestricted grammar $G = (V, \Sigma, S, P)$ kunt construeren, zó dat $L(G) = L(T)$.

De grammatica G bevat drie soorten producties. De tweede soort producties is voor het simuleren van berekeningen van de Turingmachine.

- (a) Wat is het doel van de eerste soort en de derde soort producties van G ?

- (b) Geef alle producties van de tweede soort in G die volgens de constructie van Stelling 8.14 corresponderen met de volgende transitie van Turing-machine T :



Je mag de producties algemeen beschrijven in termen van elementen σ_i van Σ en/of Γ en/of $\{\Delta\}$. Maak dan wel duidelijk wat de σ_i 's kunnen zijn. Als alternatief mag je ook aannemen dat $\Sigma = \{a, b\}$ en dat $\Gamma = \{a, b, A\}$.

5. [23 pt] Je mag er bij deze opgave vanuit gaan dat alle Turingmachines invoeralfabet $\{a, b\}$ hebben.

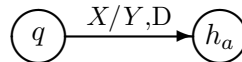
Beschouw de volgende beslissingsproblemen:

Accepts- Λ : gegeven een Turing machine T_1 , is $\Lambda \in L(T_1)$?

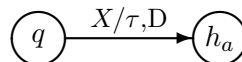
WritesSymbol: gegeven een Turing machine T_2 en een symbool σ uit het tape alfabet van T_2 , schrijft T_2 ooit σ op de tape bij invoer Λ ?

- (a) Stel dat we willen aantonen dat $\text{Accepts-}\Lambda \leq \text{WritesSymbol}$. Toon aan, met behulp van een tegenvoorbeeld, waarom we geen geldige reductie krijgen wanneer we T_2 en σ als volgt definiëren:

Laat T_1 een willekeurige instantie van *Accepts- Λ* zijn, en laat τ een element van het tape-alfabet van T_1 zijn.
 T_2 is in principe gelijk aan T_1 , maar als T_1 een transitie heeft van de vorm



(ofwel: een transitie naar h_a) dan vervangen we die in T_2 door



Ten slotte kiezen we $\sigma = \tau$.

Teken voor je tegenvoorbeeld een concrete Turingmachine, en beredeneer waarom we met bovenstaande constructie geen geldige reductie krijgen.

- (b) Beschouw ook het volgende beslissingsprobleem:

Accepts-ab:

Gegeven een Turingmachine T_3 , is $ab \in L(T_3)$?

Toon aan dat $\text{WritesSymbol} \leq \text{Accepts-ab}$. Doe dit door een geldige reductie tussen de twee beslissingsproblemen te beschrijven. Laat uiteraard ook zien dat aan alle eisen van een reductie is voldaan.

Als het je niet lukt om $\text{WritesSymbol} \leq \text{Accepts-ab}$ aan te tonen, mag je, voor drie punten minder, ook aantonen dat $\text{WritesSymbol} \leq \text{Accepts-}\Lambda$.