AUTOMATA & COMPUTABILITY

JENSEN BERNARD

CONTENTS

ABSTRACT

Dit document is tot stand gekomen om een hulp te zijn bij het studeren van het vak **Automaten en Berekenbaarheid**. Het bevat enkel voorbeelden van examenvragen, samen met een uitgewerkte oplossing. In geen enkel geval kan dit document de cursus vervangen. Ik raad aan om de cursus, die verkrijgbaar is via de cursusdienst van **Scientica**, eerst door te nemen om op zijn minst te weten waarover het gaat. Indien u fouten vind of informatie wil toevoegen, kan dit altijd via Github. Good luck*.

* *You'll need it.*

1 VOORWOORD

Lorem ipsum dolor sit amet, consectetuer adipiscing elit. Ut purus elit, vestibulum ut, placerat ac, adipiscing vitae, felis. Curabitur dictum gravida mauris. Nam arcu libero, nonummy eget, consectetuer id, vulputate a, magna. Donec vehicula augue eu neque. Pellentesque habitant morbi tristique senectus et netus et malesuada fames ac turpis egestas. Mauris ut leo. Cras viverra metus rhoncus sem. Nulla et lectus vestibulum urna fringilla ultrices. Phasellus eu tellus sit amet tortor gravida placerat. Integer sapien est, iaculis in, pretium quis, viverra ac, nunc. Praesent eget sem vel leo ultrices bibendum. Aenean faucibus. Morbi dolor nulla, malesuada eu, pulvinar at, mollis ac, nulla. Curabitur auctor semper nulla. Donec varius orci eget risus. Duis nibh mi, congue eu, accumsan eleifend, sagittis quis, diam. Duis eget orci sit amet orci dignissim rutrum.

Nam dui ligula, fringilla a, euismod sodales, sollicitudin vel, wisi. Morbi auctor lorem non justo. Nam lacus libero, pretium at, lobortis vitae, ultricies et, tellus. Donec aliquet, tortor sed accumsan bibendum, erat ligula aliquet magna, vitae ornare odio metus a mi. Morbi ac orci et nisl hendrerit mollis. Suspendisse ut massa. Cras nec ante. Pellentesque a nulla. Cum sociis natoque penatibus et magnis dis parturient montes, nascetur ridiculus mus. Aliquam tincidunt urna. Nulla ullamcorper vestibulum turpis. Pellentesque cursus luctus mauris.

Nulla malesuada porttitor diam. Donec felis erat, congue non, volutpat at, tincidunt tristique, libero. Vivamus viverra fermentum felis. Donec nonummy pellentesque ante. Phasellus adipiscing semper elit. Proin fermentum massa ac quam. Sed diam turpis, molestie vitae, placerat a, molestie nec, leo. Maecenas lacinia. Nam ipsum ligula, eleifend at, accumsan nec, suscipit a, ipsum. Morbi blandit ligula feugiat magna. Nunc eleifend consequat lorem. Sed lacinia nulla vitae enim. Pellentesque tincidunt purus vel magna. Integer non enim. Praesent euismod nunc eu purus. Donec bibendum quam in tellus. Nullam cursus pulvinar lectus. Donec et mi. Nam vulputate metus eu enim. Vestibulum pellentesque felis eu massa.

2 TALEN EN AUTOMATEN

Fusce mauris. Vestibulum luctus nibh at lectus. Sed bibendum, nulla a faucibus semper, leo velit ultricies tellus, ac venenatis arcu wisi vel nisl. Vestibulum diam. Aliquam pellentesque, augue quis sagittis posuere, turpis lacus congue quam, in hendrerit risus eros eget felis. Maecenas eget erat in sapien mattis porttitor. Vestibulum porttitor. Nulla facilisi. Sed a turpis eu lacus commodo facilisis. Morbi fringilla, wisi in dignissim interdum, justo lectus sagittis dui, et vehicula libero dui cursus dui. Mauris tempor ligula sed lacus. Duis cursus enim ut augue. Cras ac magna. Cras nulla. Nulla egestas. Curabitur a leo. Quisque egestas wisi eget nunc. Nam feugiat lacus vel est. Curabitur consectetuer.

3 TALEN EN BEREKENBAARHEID

3.1 A_{TM} is niet-beslisbaar

Bewijs in detail dat A_{TM} niet beslisbaar is en steun daarbij niet op de stelling van Rice. Zou het helpen als het toegelaten was op de stelling van Rice te steunen? Is A_{TM} herkenbaar? Co-herkenbaar?

A_{TM} is niet beslisbaar

Proof. Beter bekend als het acceptatieprobleem voor Turingmachines. De geassocieerde taal is

$$A_{TM} = \{< M, s > \mid M \textit{ is een Turingmachine en } s \in L_M\}$$

Stel er bestaat een beslisser B voor A_{TM}. Dat betekent dat bij input $< M, s >$ B accepteert als M bij input s stopt in zijn q_a en verwerpt als M bij input s stopt in zijn q_r of loopt. We schrijven

$$B(< M, s >) \textit{ is accept als } M \ s \textit{ accepteert en anders reject}$$

Construeer nu de contradictie machine C met eigenschap:

$$C(< M >) = \mathtt{opposite}(B(< M, M >)) \textit{ voor elke Turingmachine } M$$

Daarbij is $\mathtt{opposite}(\mathtt{accept}) = \mathtt{reject}$ en $\mathtt{opposite}(\mathtt{reject}) = \mathtt{accept}$.
Neem nu voor M hierboven C zelf, dan krijgen we:

$$C(< C >) = \mathtt{opposite}(B(< C, C >))$$

Als $C(< C >) = \mathtt{accept}$, dan is $B(< C, C >) = \mathtt{accept}$, dan is $\mathtt{opposite}(B(< C, C >)) = \mathtt{reject}$, dan is $C(< C >) = \mathtt{reject}$, dan is $B(< C, C >) = \mathtt{reject}$, dan is $\mathtt{opposite}(B(< C, C >)) = \mathtt{accept}$, dan is $C(< C >) = \mathtt{accept} \ldots$
Dus C kan niet bestaan, dus B bestaat niet. Dus A_{TM} is niet beslisbaar. □

A_{TM} is herkenbaar

Proof. De herkenner A voor A_{TM} laat, met input $< M, x >$, M lopen op s. Indien deze M s accepteerd, dan accepteerd A zijn input. Indien deze de input $\mathtt{reject}$, of gewoon niet stopt, dan zal A deze ook niet accepteren. A_{TM} is dus herkenbaar. □

A_{TM} is niet co-herkenbaar

Proof. A_{TM} kan echter niet co-herkenbaar zijn. We bewijzen dit met contradictie. Indien A_{TM} co-herkenbaar is, is deze dus herkenbaar en co-herkenbaar (zie vorig bewijs). Wanneer een taal deze beide eigenschappen bezit, is deze beslisbaar. Dit is een contradictie met het eerste bewijs. □

De stelling van Rice

Deze stelling bewijst dat elke taal (die aan bepaalde voorwaarden voldoet) niet-beslisbaar is. We moeten eerst de volgende twee definities toelichten alvorens we verder kunnen gaan met het bewijs van A_{TM}, gebruik makend van deze stelling (wat korter zal zijn als voorgaande bewijzen).

Theorem 1 (Niet-triviale eigenschap)**.** *Een eigenschap P van Turingmachines heet niet-triviaal indien* $\mathrm{Pos}_P \neq \emptyset$ *en ook* $\mathrm{Neg}_P \neq \emptyset$.

Theorem 2 (Taal-invariante eigenschap)**.** *Een eigenschap* P *heet taal-invariant indien alle machines die dezelfde taal bepalen, hebben ofwel allemaal* P*, ofwel heeft geen enkele ervan* P*.*

$$L_{M_1} = L_{M_2} \Rightarrow P(M_1) = P(M_2)$$

Theorem 3 (Formelere stelling van Rice)**.** *Voor elke niet-trivial, taal-invariante eigenschap* P *van Turingmachines geldt dat* Pos_P *(en ook* Neg_P*) niet beslisbaar is.*

Proof. Veronderstel dat $M_\emptyset$ (de machine die de lege taal beslist) de eigenschap P niet heeft - indien dat niet zo is, verander dan P in zijn negatie. Vermits P niet-triviaal is, bestaat er een taal L_X zodat X een Turingmachine is met de eigenschap P. Stel dat Pos_P (en dus ook Neg_P) beslisbaar id: we zullen een beslisser B voor Pos_P gebruiken om een beslisser A te maken voor A_{TM}. A krijgt als input $< M, s >$ en doet het volgende:

1. construeer een hulpmachine $H_{M,s}$ die het volgende doet bij input x:
   a) laat M lopen op s
   b) indien M s accepteert, laat dan X lopen op x en accepteer als X x accepteert
2. laat nu B los op $H_{M,s}$
3. als B $H_{M,s}$ accepteert, dan $\mathtt{accept}$, anders $\mathtt{reject}$

$H_{M,s}$ accepteert ofwel de lege taal, ofwel L_X (TODO: waarom). A accepteert $< M, s >$ als en slechts als B $H_{M,s}$ accepteert, als en slechts als $H_{M,s}$ de eigenschap P heeft, als en slechts als $H_{M,s}$ accepteert L_X, als en slechts als M accepteert s.
Dus, A is een beslisser voor A_{TM}, hetgeen niet kan, dus kan B niet bestaan en Pos_P is niet beslisbaar. □

4 HERSCHRIJFSYSTEMEN

Quisque ullamcorper placerat ipsum. Cras nibh. Morbi vel justo vitae lacus tincidunt ultrices. Lorem ipsum dolor sit amet, consectetuer adipiscing elit. In hac habitasse platea dictumst. Integer tempus convallis augue. Etiam facilisis. Nunc elementum fermentum wisi. Aenean placerat. Ut imperdiet, enim sed gravida sollicitudin, felis odio placerat quam, ac pulvinar elit purus eget enim. Nunc vitae tortor. Proin tempus nibh sit amet nisl. Vivamus quis tortor vitae risus porta vehicula.

Fusce mauris. Vestibulum luctus nibh at lectus. Sed bibendum, nulla a faucibus semper, leo velit ultricies tellus, ac venenatis arcu wisi vel nisl. Vestibulum diam. Aliquam pellentesque, augue quis sagittis posuere, turpis lacus congue quam, in hendrerit risus eros eget felis. Maecenas eget erat in sapien mattis porttitor. Vestibulum porttitor. Nulla facilisi. Sed a turpis eu lacus commodo facilisis. Morbi fringilla, wisi in dignissim interdum, justo lectus sagittis dui, et vehicula libero dui cursus dui. Mauris tempor ligula sed lacus. Duis cursus enim ut augue. Cras ac magna. Cras nulla. Nulla egestas. Curabitur a leo. Quisque egestas wisi eget nunc. Nam feugiat lacus vel est. Curabitur consectetuer.

5 ANDERE REKENPARADIGMA'S

Suspendisse vitae elit. Aliquam arcu neque, ornare in, ullamcorper quis, commodo eu, libero. Fusce sagittis erat at erat tristique mollis. Maecenas sapien libero, molestie et, lobortis in, sodales eget, dui. Morbi ultrices rutrum lorem. Nam elementum ullamcorper leo. Morbi dui. Aliquam sagittis. Nunc placerat. Pellentesque tristique sodales est. Maecenas imperdiet lacinia velit. Cras non urna. Morbi eros pede, suscipit ac, varius vel, egestas non, eros. Praesent malesuada, diam id pretium elementum, eros sem dictum tortor, vel consectetuer odio sem sed wisi.

6 TALEN EN COMPLEXITEIT

Sed feugiat. Cum sociis natoque penatibus et magnis dis parturient montes, nascetur ridiculus mus. Ut pellentesque augue sed urna. Vestibulum diam eros, fringilla et, consectetuer eu, nonummy id, sapien. Nullam at lectus. In sagittis ultrices mauris. Curabitur malesuada erat sit amet massa. Fusce blandit. Aliquam erat volutpat. Aliquam euismod. Aenean vel lectus. Nunc imperdiet justo nec dolor.