Turing Maskiner

Jacob Nielsen

I 1936 publicerede Alan Turing artiklen[1] "On Computable Numbers with an Application to the Entscheidungsproblem". I artiklen lancerer Turing det, der senere kommer til at hedde Turing maskiner. Turing maskiner er programmerbare beregnings maskiner. Alle moderne computere er opbygget efter samme pricipper som en Turing maskine. Turings arbejde er behandlet i B. Jack Copeland´s bog[2]: "The Essential Turing".

Alan Turing interesserede sig for, hvilke opgaver der i princippet kan løses af sådan en maskine. Alan Turing og Alonso Church løste problemet. Deres løsning har siden givet navn til den såkaldte Church-Turing sætning:

Enhver beregning, der kan beskrives som en algoritme, kan udføres af en Turing maskine.

En algoritme er en opskrift på en beregning. Det vi i dag kalder et computerprogram er en algoritme. Turing beskrev også en universel turingmaskine, der kan udføre enhever beregning, der kan udføres af nogen turingmaskine. Den universelle turingmaskine svarer til en moderne computer, der kan udføre forskellige opgaver afhængigt af hvilket program der indlæses i computeren.

David Hilberts Entscheidungsproblem.drejer sig om, hvilke matematiske spørgsmål, vi kan afgøre sandhedsværdien af; eller med andre ord: Hvilke sætninger vi kan bevise eller modbevise. På Hilberts tid omkring år 1900 drømte matematikerne om at finde en bevis algoritme, hvor man kunne starte med en sætning og så regne sig frem til dens sandhedsværdi.
Turing så mere specifikt på, hvilke tal, der kan beregnes med en turingmaskine. Dette førte frem til det såkaldte "halt problem"; altså spørgsmålet om maskinen nogensinde stopper, når den er sat igang med en given beregning. Den berømte "Fermat´s store sætning" kan omformuleres som et halt problem.

Fermats sætning handler om løsninger til ligningen:

$$x^{n+2} + y^{n+2} = z^{n+2} \quad x, y, z, n \in \mathbb{N}$$

Sætningen siger, at ligningen ikke har nogen løsninger.

Vi kan skrive et computerprogram, der starter fra en ende af og prøver alle sæt af fire hele tal efter. Maskinen skal stoppe, hvis den finder et talsæt, der opfylder ligningen. Hvis maskinen stopper, så er problemet løst. Hvis vi kan bevise, at maskinen aldrig stopper, så er problemet også løst. Vi står altså med et halt problem !

En turing maskine består af nogle programlinjer,, en læse/skriveenhed og et lager. I den oprindelige udgave af turingmaskinen er lagerenheden en lang strimmel med nogle felte, hvor der kan stå et "input" inden maskinen starter beregningen. Maskinen bevæger sig nu frem og tilbage langs strimmelen og skriver enten nuller eller ettaller i felterne. Maskinens output har ingen værdi, før maskinen stopper, idet værdien i alle felter i princippet kan ændres, indtil beregningen er færdig.

Vi skal i det følgende se grundigt på en konkret turing maskine; nemlig maskinen der udfører Euclids algoritme. Euclids algoritme er en metode til beregning af den største fælles divisor i to tal.

---

[1] Proceedings of the London Mathematical Society, Series 2, 42 (1936-7).

[2] B.J.Copeland, "The Essential Turing", Claredon Press Oxford, 2004.

Roger Penrose har behandlet Turing maskiner i sin bog om kunstig intelligens[3]: “The Emperors New Mind”. Nedenstående gennemgang af en Turing maskine bygger på kapitel to i Penroses bog.

Programmet ser ud som følger:

I :	0 **0**→0 **0** R	0 **1**→1 **1** L
II :	1 **0**→10 1 R	1 **1**→1 **1** L
III:	10 **0**→1010 **0** R	10 **1**→11 **0** R
IV:	11 **0**→100 **0** R	11 **1**→11 **1** R
V:	100 **0**→100 **0** R	100 **1**→101 **0** R
VI:	101 **0**→111 **0** L	101 **1**→110 **1** L
VII:	110 **0**→110 **0** L	110 **1**→1 **1** L
VIII:	111 **0**→111 **0** L	111 **1**→1000 **1** L
IX:	1000 **0**→1001 **0** L	1000 **1**→1000 **1** L
X:	1001 **0**→10 **0** R	1001 **1**→1 **1** L
XI:	1010 **0**→0 **0** STOP	1010 **1**→1010 **1** R

Ser man for eksempel på linje III, så betyder koden: 10 ***0***→ *1010* ***0*** *R, at hvis maskinen er i tilstand 10 og møder et nul, så går maskinen til tilstand 1010, skriver nul i feltet (det stod der godt nok i forvejen) og går til højre.*

Vi følger nu algoritmen, og ser om maskinen finder den største fælles divisor i fire og seks. På strimlen står de to tal fire og seks adskilt af et nul. Her skrives tal med et antal ettaller svarende til tallets størrelse. Maskinens læser er fra start anbragt til venstre for det første ettal, der overhovedet optræder på strimlen. Instruksen o0→o0R får læsehovedet til at bevæge sig til højre, indtil det møder det første ettal. Det røde felt markerer læsehovedets position. Tallet i feltet er det tal, som maskinen møder. De blå tal yderst til venstre markerer maskinens tilstand, når den ankommer til det markerede felt..

0	00000000000000000000000000000000111101111100000000000000000000000000000000000
1	00000000000000000000000000000000111101111100000000000000000000000000000000000
10	00000000000000000000000000000001111101111100000000000000000000000000000000000
11	00000000000000000000000000000001011101111100000000000000000000000000000000000
11	00000000000000000000000000000001011101111100000000000000000000000000000000000
11	00000000000000000000000000000001011101111100000000000000000000000000000000000
11	00000000000000000000000000000001011101111100000000000000000000000000000000000
100	00000000000000000000000000000001011101111100000000000000000000000000000000000
101	00000000000000000000000000000001011100111100000000000000000000000000000000000
110	00000000000000000000000000000001011100111100000000000000000000000000000000000
110	00000000000000000000000000000001011100111100000000000000000000000000000000000
110	00000000000000000000000000000001011100111100000000000000000000000000000000000
1	00000000000000000000000000000001011100111100000000000000000000000000000000000
1	00000000000000000000000000000001011100111100000000000000000000000000000000000
1	00000000000000000000000000000001011100111100000000000000000000000000000000000
10	00000000000000000000000000000001111100111100000000000000000000000000000000000

I det følgende springes nogle trin over, når der ikke sker andet end, at læsehovedet flytter sig.

11	00000000000000000000000000000001101100111100000000000000000000000000000000000
11	00000000000000000000000000000001101100111100000000000000000000000000000000000

[3]Roger Penrose, “The Emperors New Mind”, Oxford University Press 1989.

Vi fortsætter, hvor vi slap. Programmet gentages, så vi har det ved hånden.

I :	0 **0**→0 **0** R	0 **1**→1 **1** L
II :	1 **0**→10 1 R	1 **1**→1 **1** L
III:	10 **0**→1010 **0** R	10 **1**→11 **0** R
IV:	11 **0**→100 **0** R	11 **1**→11 **1** R
V:	100 **0**→100 **0** R	100 **1**→101 **0** R
VI:	101 **0**→111 **0** L	101 **1**→110 **1** L
VII:	110 **0**→110 **0** L	110 **1**→1 **1** L
VIII:	111 **0**→111 **0** L	111 **1**→1000 **1** L
IX:	1000 **0**→1001 **0** L	1000 **1**→1000 **1** L
X:	1001 **0**→10 **0** R	1001 **1**→1 **1** L
XI:	1010 **0**→0 **0** STOP	1010 **1**→1010 **1** R

11	00000000000000000000000000000001**10110011111**0000000000000000000000000000000000000
100	00000000000000000000000000000001**10110011111**0000000000000000000000000000000000000
100	00000000000000000000000000000001**10110011111**0000000000000000000000000000000000000
101	00000000000000000000000000000001**10110001111**0000000000000000000000000000000000000
110	00000000000000000000000000000001**10110001111**0000000000000000000000000000000000000
110	00000000000000000000000000000001**10110001111**0000000000000000000000000000000000000
1	00000000000000000000000000000001**10110001111**0000000000000000000000000000000000000
1	00000000000000000000000000000001**10110001111**0000000000000000000000000000000000000
10	00000000000000000000000000000001**11110001111**0000000000000000000000000000000000000
11	00000000000000000000000000000001**11010001111**0000000000000000000000000000000000000
11	00000000000000000000000000000001**11010001111**0000000000000000000000000000000000000
100	00000000000000000000000000000001**11010001111**0000000000000000000000000000000000000
101	00000000000000000000000000000001**11010000111**0000000000000000000000000000000000000
110	00000000000000000000000000000001**11010000111**0000000000000000000000000000000000000
110	00000000000000000000000000000001**11010000111**0000000000000000000000000000000000000
1	00000000000000000000000000000001**11010000111**0000000000000000000000000000000000000
10	00000000000000000000000000000001**11110000111**0000000000000000000000000000000000000
11	00000000000000000000000000000001**11100000111**0000000000000000000000000000000000000
100	00000000000000000000000000000001**11100000111**0000000000000000000000000000000000000
110	00000000000000000000000000000001**11100000011**0000000000000000000000000000000000000
1	00000000000000000000000000000001**11100000011**0000000000000000000000000000000000000
1	00000000000000000000000000000001**11100000011**0000000000000000000000000000000000000
10	00000000000000000000000000000011**11100000011**0000000000000000000000000000000000000
11	00000000000000000000000000000010**11100000011**0000000000000000000000000000000000000
11	00000000000000000000000000000010**11100000011**0000000000000000000000000000000000000
100	00000000000000000000000000000010**11100000011**0000000000000000000000000000000000000
1	00000000000000000000000000000010**11100000001**0000000000000000000000000000000000000

I :	0 0→0 0 R	0 1→1 1 L
II :	1 0→10 1 R	1 1→1 1 L
III:	10 0→1010 0 R	10 1→11 0 R
IV:	11 0→100 0 R	11 1→11 1 R
V:	100 0→100 0 R	100 1→101 0 R
VI:	101 0→111 0 L	101 1→110 1 L
VII:	110 0→110 0 L	110 1→1 1 L
VIII:	111 0→111 0 L	111 1→1000 1 L
IX:	1000 0→1001 0 L	1000 1→1000 1 L
X:	1001 0→10 0 R	1001 1→1 1 L
XI:	1010 0→0 0 STOP	1010 1→1010 1 R

1	00000000000000000000000000000010111000000010000000000000000000000000000000000000
10	00000000000000000000000000000011111000000010000000000000000000000000000000000000
11	00000000000000000000000000000011011000000010000000000000000000000000000000000000
11	00000000000000000000000000000011011000000010000000000000000000000000000000000000
100	00000000000000000000000000000011011000000010000000000000000000000000000000000000
111	00000000000000000000000000000011011000000000000000000000000000000000000000000000
1000	00000000000000000000000000000011011000000000000000000000000000000000000000000000
1001	00000000000000000000000000000011011000000000000000000000000000000000000000000000
1	00000000000000000000000000000011011000000000000000000000000000000000000000000000
10	00000000000000000000000000000111011000000000000000000000000000000000000000000000
11	00000000000000000000000000000101011000000000000000000000000000000000000000000000
100	00000000000000000000000000000101011000000000000000000000000000000000000000000000
101	00000000000000000000000000000101001000000000000000000000000000000000000000000000
110	00000000000000000000000000000101001000000000000000000000000000000000000000000000
1	00000000000000000000000000000101001000000000000000000000000000000000000000000000
10	00000000000000000000000000000111001000000000000000000000000000000000000000000000
11	00000000000000000000000000000110001000000000000000000000000000000000000000000000
100	00000000000000000000000000000110001000000000000000000000000000000000000000000000
101	00000000000000000000000000000110000000000000000000000000000000000000000000000000
111	00000000000000000000000000000110000000000000000000000000000000000000000000000000
1000	00000000000000000000000000000110000000000000000000000000000000000000000000000000
1001	00000000000000000000000000000110000000000000000000000000000000000000000000000000
10	00000000000000000000000000000110000000000000000000000000000000000000000000000000
1010	00000000000000000000000000000110000000000000000000000000000000000000000000000000
1010	00000000000000000000000000000110000000000000000000000000000000000000000000000000

STOP

Til venstre for læseren står nu resultatet nemlig to, der er den største fælles divisor i fire og seks.