por **Amílcar Branquinho e Alexander Kovačec**
[Departamento de Matemática, Universidade de Coimbra]

Máquinas de Turing

Caro Leitor,
Propomos-lhe a construção, em essência, de computadores ideais, apresentada por A.M. Turing, e dotados não propriamente de inteligência artificial, mas de testes e algoritmos que nos ajudam a compreender a complexidade humana.

Um dos conceitos-chave das ciências de computação é o de *máquina de Turing*. Este modelo simplista de um computador fictício foi inventado pelo matemático inglês Alan Mathison Turing (1912-1954) num famoso trabalho de 1936 intitulado *On computable numbers with an application to the Entscheidungsproblem*, citado, aliás, no artigo de Paulo Mateus, "O que é o problema P≠NP?",*Gazeta de Matemática*, 157, 2009, 49-51, no qual se indica que com a sua solução se pode ganhar um milhão de dólares. Nesta altura, expressões tais como *métodos efectivos* ou *procedimentos mecânicos*, etc., foram introduzidas para aludir a métodos sistemáticos para o cálculo de objectos como, por exemplo, o máximo divisor comum de dois inteiros positivos à maneira de Euclides, ou decisões como a da primalidade de um número p, testando sucessivamente os números $2,\ldots,\lfloor\sqrt{p}\rfloor$ como candidatos a divisores. O seu nome está ainda ligado à nossa história recente, como se pode ver no artigo de A. Machiavelo, "ENIGMA: uma história que devia ser melhor contada", *Gazeta de Matemática*, 147, 2004, 14-15, mostrando que a matemática pode alimentar o sonho de liberdade do Homem e salvar milhares de vidas.

Na famosa palestra "Mathematische Probleme" proferida em 1900, o influente matemático David Hilbert enunciou como o seu décimo problema: *dada uma equação diofantina arbitrária, dar um método que permita decidir num número finito de passos se a equação tem uma solução no conjunto dos números inteiros*.

Aparentemente Hilbert estava convicto da solubilidade deste problema, i.e. da existência de um tal método mecânico-algorítmico, para resolver tais questões para quaisquer equações diofantinas.

O artigo de Turing contém, se bem que em linguagem diferente, argumentos convincentes de que as funções $f\colon \mathrm{IN}^k \to \mathrm{IN}$ computáveis ou problemas decidíveis, no sentido vago anterior exposto, são exactamente aqueles calculáveis ou decidíveis por máquinas automáticas, hoje chamadas máquinas de Turing. Na verdade, apenas através de uma clara e convincente definição técnica dos conceitos de computabilidade e decidibilidade é possível dar respostas demonstravelmente negativas a tais questões. Nesse trabalho foram ainda demonstradas a não computabilidade de certas funções e a não decidibilidade (por métodos efectivos) de certas questões. Com efeito, o autor demonstra que o seu conceito de computabilidade equivale aos conceitos, aparentemente muito diferentes, da λ-definibilidade de Church e, um pouco mais tarde, mostra também a equivalência com o conceito de computabilidade proposto por Gödel, que usa funções recursivas – conceitos estes que na altura estavam por definir. A equivalência da calculabilidade efectiva de Herbrand com a λ – definibilidade foi mostrada por Church. Estas equivalências dão credibilidade à tese de Turing e Church de que todos os procedimentos efectivos em matemática – aqueles efectuáveis por quem obedece rigorosamente a uma sucessão de instruções - são implementáveis por máquinas de Turing.

Na última secção do artigo citado, Turing escreve: *"Proponho-me demonstrar que não existe um procedimento geral para determinar se uma dada fórmula do cálculo funcional K é demonstrável."*

Deste modo fica resolvido, de forma negativa, o importante problema, denominado "Entscheidungsproblem" (i.e. problema da decisão), do famoso livro de Hilbert e Ackermann de 1931, *"Grundzüge der Theoretischen Logik"* (i.e. Linhas condutoras da lógica teórica).

Essencialmente, uma *máquina de Turing* consiste numa cabeça apta a ler e escrever 0's e 1's numa fita infinita de papel dividida em células e numa tabela que indica como actuar face ao seu estado interno e ao símbolo lido. À partida, a fita está preenchida com os símbolos 0 e 1. Cada passo de uma computação decompõe-se em outros três passos elementares executados pela seguinte ordem:

- o símbolo lido é alterado ou reescrito;
- a cabeça move-se zero (O) ou um passos para a esquerda (L de "left") ou para a direita (R de "right");
- o estado da máquina é alterado ou reassumido.

	0	1
1		0R2
2	1L3	1R2
3	0R4	1L3

$\overset{1}{0}111011$

0R2

$00\overset{2}{1}1011$

1R2

$001\overset{2}{1}011$

1R2

$0011\overset{2}{0}11$

1L3

$001\overset{3}{1}111$

1L3

$00\overset{3}{1}1111$

1L3

$0\overset{3}{0}11111$

0R4

$00\overset{4}{1}1111$

A tabela que doravante vamos identificar com a própria máquina M tem um número finito de filas indexadas com certos inteiros positivos, chamados os *estados da máquina*, e duas colunas indexadas por 0 e 1. Para o estado $i \in \{1, 2, 3, ...\}$ de M e o símbolo lido $\delta \in \{0, 1\}$, o comando a obedecer é o que está na intersecção da fila i com a coluna δ, i.e. na entrada $M_{i,\delta}$ da máquina. A entrada pode ser vazia ou não existir; neste caso, a máquina, por definição, pára. Por exemplo, na máquina ao lado, se $i = 2$, $\delta = 1$, então $M_{i,\delta} = M_{2,1} = 1R2$.

O comando 1R2 significa: "escrever 1, ir um passo para a direita e assumir o estado 2". A coluna ao lado exemplifica uma computação.

A máquina começa com a fita mais acima; as partes da fita que não vemos são supostamente preenchidas com 0's. Está no estado 1 e lê um 1. Logo, o comando a executar é $M_{1,1}$= 0R2. A cabeça substitui então o símbolo que lê por 0, vai para a direita, e a máquina assume o estado 2. Agora a máquina lê no estado 2 um 1. Portanto faz como manda a entrada $M_{2,1}$= 1R2: escreve um 1, vai para a direita, e assume o estado 2; etc. O leitor verifique o resto da computação. A máquina pára quando chega ao estado indefinido 4.

A computação mostrada transforma uma fita com um bloco de três 1's e um de dois 1's, separados por um único 0, num único bloco de 3 + 2 = cinco 1's, e pára com a cabeça sobre o primeiro 1 do bloco produzido. De facto, esta máquina transforma qualquer fita, $0\overset{1}{1}11\ldots1011\ldots1$ consistindo em dois blocos de m e n 1's, respectivamente num único bloco de $m + n$ 1's, e parará com a cabeça sobre o primeiro 1 do primeiro bloco.

Reflectindo um pouco, vê-se porquê. A máquina implementa a ideia de "apagar o primeiro 1, ir com a cabeça até ao 0 separador dos blocos de 1's, substituir esse 0 por 1, e voltar ao início". Ideias como estas guiam a construção de uma máquina.

A máquina que acabámos de descrever demonstra a computabilidade da função $\mathrm{IN}^2 \ni (m, n) \mapsto m + n \in \mathrm{IN}$, i.e. da função soma, no sentido da definição técnica seguinte, que utiliza a notação 0^r ou 1^r para indicar blocos de r 0's ou r 1's consecutivos; assim, por exemplo, $01^4 0^2 1^3$ é o mesmo que 0111100111.

Uma função $g : \mathrm{IN}^k \to \mathrm{IN}$ diz-se *computável* se existir uma máquina que para todo o k-uplo $\underline{n} = (n_1, ..., n_k) \in \mathrm{IN}^k$ transforma a fita $0\overset{1}{1}{}^{n_1}01^{n_2}0...01^{n_k}$ numa fita $0^s \overset{*}{1}{}^{g(\underline{n})}$ (com s natural certo), e pára.

Se escrevemos, como aqui, $\overset{1}{1}{}^{n_1}$ ou $\dot{1}^m$, então supomos que a cabeça está sobre o 1 mais à esquerda do bloco (e a máquina em estado 1 ou num estado não especificado, respectivamente). Esta definição traduz que a máquina, se alimentada com a codificação natural de um k-uplo $\underline{n} = (n_1, ..., n_k)$, i.e. uma fita contendo k blocos de 1's de comprimentos respectivos $n_1, ..., n_k$, separados por 0's isolados, vai parar quando atingir a sua forma de dizer $g(\underline{n})$, i.e. quando atingir um único bloco de $g(\underline{n})$ símbolos 1.

Como mencionámos, as máquinas de Turing podem também ser usadas para problemas de decisão. Vejamos um exemplo.

Diz-se que um conjunto $X \subseteq IN$ é *decidível* se a função indicatriz 1_X: $\mathrm{IN} \rightarrow \{1,2\}$, definida por $1_X(n)=1$ se $n \in X$ e $1_X(n)=2$ se $n \notin X$, for computável.

Devolver 1 ou 2 é uma forma para uma máquina dizer "sim" ou "não" quando "interrogada" sobre se $n \in X$.

É neste contexto que surgem os seguintes problemas.

	0	1
1	1R2	1R2
2	0L6	0R3
3	0L4	0R2
4	0L4	1R5
5	1L6	
6	0L6	1O7

Problema 1. Mostre que o conjunto $X = \{1, 3, 5, \ldots\}$ dos naturais ímpares é decidível. Para tal, mostre que, se a máquina do lado arrancar com uma fita da forma $0\overset{1}{1}11\ldots1$ com n 1s, ela devolve $0\overset{7}{1}1$ se n for ímpar, e $0\overset{7}{1}$ se n for par.

Preparado com estas análises, o leitor pode consolidar a sua perícia respondendo aos problemas 2, 3, e 4 a seguir.

Problema 2. Mostrar que a função $\mathrm{IN} \ni n \overset{f}{\mapsto} n+1$ é computável (por uma máquina com não mais do que cinco filas).

Problema 3. Mostrar que a função

$$\mathrm{IN} \ni n \overset{f}{\mapsto} \begin{cases} n & se \quad n \in \{1,2,3\} \\ n-2 & se \quad n \geq 4 \end{cases} \text{ é computável.}$$

Problema 4. Mostrar que o conjunto X={3,6,9,...}dos inteiros positivos divisíveis por 3 é decidível.

Envie as soluções para:
Projecto Delfos
Departamento de Matemática da FCTUC
Apartado 3008
EC Universidade
3001-454 Coimbra

No entanto, não se esqueça o leitor de que indecidibilidade algorítmica ou indemonstrabilidade automática não é o mesmo que qualquer tipo de indecidibilidade absoluta ou indemonstrabilidade informal. Os humanos são genuinamente mais complexos do que os autómatos...

Apresentamos agora as propostas de resolução dos problemas 2 e 3 do *Canto Délfico* na *Gazeta de Matemática* 157, 2009, por **Carlos Alberto Silva Gomes**, Escola S/3 Amarante.

Problema 2: Considere-se uma pirâmide com altura H e área de base F. Mostre que o volume da pirâmide é igual a $FH/3$.

Solução: Dividindo a altura em n partes iguais e traçando por elas planos paralelos ao plano da base, obtemos n prismas V_k, com altura H/n. O seu volume vem dado por $F_k H/n$. Uma vez que

$$\frac{F_k}{F} = \left(\frac{kH/n}{H}\right)^2 = \left(\frac{k}{n}\right)^2$$

tem-se $F_k = F(k/n)^2$ e, assim, o volume de $V_k = FH(k^2/n^3)$.

O volume da pirâmide será então dado por:

$$V = \lim \frac{FH}{n^3} \sum_{k=1}^{n} k^2 = FH \lim \frac{n(n+1)(2n+1)}{6n^3}$$

$$\text{i.e.} \quad V = \frac{FH}{6} \lim \left(2 + \frac{3}{n} + \frac{1}{n^2}\right)$$

Logo, o volume da pirâmide será:

$$V = \frac{FH}{6} \lim \left(2 + \frac{3}{n} + \frac{1}{n^2}\right) = \frac{FH}{3}$$

Problema 3: Considere-se a semiesfera de raio R dividida em n planos com espessura R/n. Sendo o raio de cada um destes cilindros resultantes r_k, mostre por um processo análogo ao do problema 2 que o volume da esfera é $\frac{4}{3}\pi R^3$.

Solução: O volume de cada um dos cilindros é $V_k = F_k R/n$, em que F_k é a sua base. Mas

$$\frac{F_k}{F} = \left(\frac{r_k}{R}\right)^2 = \frac{r_k^2}{R^2}, \text{ logo } F_k = F\left(1 - \frac{k^2}{n^2}\right).$$

Temos então que $V_k = \frac{FR}{n}\left(1 - \frac{k^2}{n^2}\right)$, logo

$$\sum_{k=1}^{n} \frac{FR}{n}\left(1 - \frac{k^2}{n^2}\right) = FR - \frac{FR}{n^3}\left(\frac{n(n+1)(2n+1)}{6}\right)$$

e, portanto,

$$\lim \left[FR - \frac{FR}{6}\left(2 + \frac{3}{n} + \frac{1}{n^2}\right)\right] = \frac{2}{3}\pi R^3.$$

Assim,

$$V_{esfera} = 2 \times \left(\frac{2}{3}\pi R^3\right) = \frac{4}{3}\pi R^3.$$

Caderno_2
quinta-feira, 10 de Dezembro de 2009 18:25:02