

Máquina de Turing

É um dispositivo imaginário que formou a estrutura para fundamentar a ciência da computação moderna.

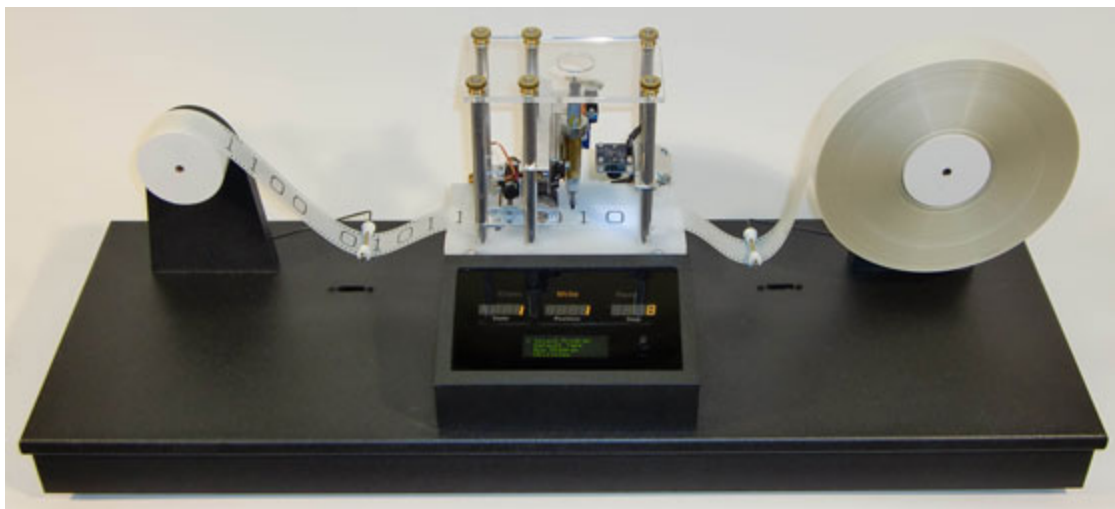
Seu inventor, o matemático Alan Mathison Turing, mostrou que a computação das operações de leitura, escrita e exclusão de símbolos binários poderiam ser satisfeitas por uma máquina que continha uma fita de comprimento ilimitado, com quadrados de tamanho definido sobre ela e um dispositivo com um número finito de estados, que realizava as operações na fita.

Em 1936 foi formalizado o termo algoritmo: um conjunto finito de instruções simples e precisas, que são descritas com um número finito de símbolos.

“Qualquer processo aceito por nós homens como um algoritmo é precisamente o que uma máquina de Turing pode fazer” (Alonzo Church, matemático).

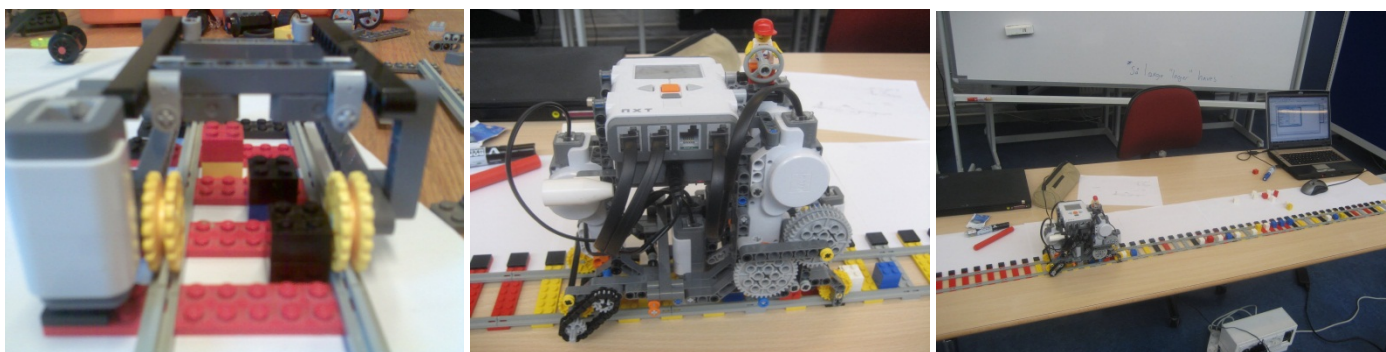
Máquinas de Turing para construção: Foram encontrados dois modelos que se destacaram em nossas pesquisas.

1. Modelo mais completo, com circuitos digitais e implementações de diversos algoritmos: <http://aturingmachine.com>, com vídeo demonstrativo: <http://www.youtube.com/watch?v=E3keLeMwfHY>



É de difícil implementação, por isso foi feito o contato para a exposição.

2. Utilizando um kit Lego: <http://legoofdoom.blogspot.com>, com vídeo demonstrativo: <http://www.youtube.com/watch?v=cYw2ewoO6c4>



Em relação a primeira (1), sua implementação é mais fácil, porém vai requerer bastante trabalho para ela realizar as operações de leitura, escrita e exclusão e executar alguns algoritmos simples como somatório e subtração. No endereço indicado há um detalhamento da construção e o contato com os autores.

A MÁQUINA DE TURING

Rogério Xavier de Azambuja

Elias Ramos

Novembro/2011.

Trata-se de um dispositivo imaginário embasado por uma teoria revolucionária do seu autor, o britânico Alan Mathison Turing, concebida aos 24 anos de idade. A máquina de Turing formou a estrutura básica para fundamentar a ciência da computação moderna e a computabilidade. Foi responsável anos depois, pelo reconhecimento da comunidade científica, declarando Turing com o título simbólico de “pai da computação”.

A teoria foi publicada pela primeira vez em 1936, num artigo intitulado "*On Computable Numbers, with an Application on the Entscheidungsproblem*", em resposta ao tratamento do problema da decisão, formulado por Hilbert. Turing estudava na universidade de Princeton, Nova Jersey-EUA.

Apesar da máquina de Turing não ter sido implementada fisicamente, na totalidade pelo seu autor, o processo computacional foi matematicamente demonstrado e provado no artigo. Turing explicitou um dispositivo lógico que ele chamou de "*automatic machine*" (ou "*a-machine*"), capaz de ler, escrever e apagar símbolos binários em uma fita de comprimento ilimitado e dividida por quadrados de igual tamanho. Uma cabeça de leitura/gravação se moveria em qualquer direção ao longo da fita, um quadrado por vez, e uma unidade de controle poderia interpretar uma lista de instruções simples, movendo-se para a direita ou esquerda. A regra executada determina o que se convencionou chamar de estado da máquina.

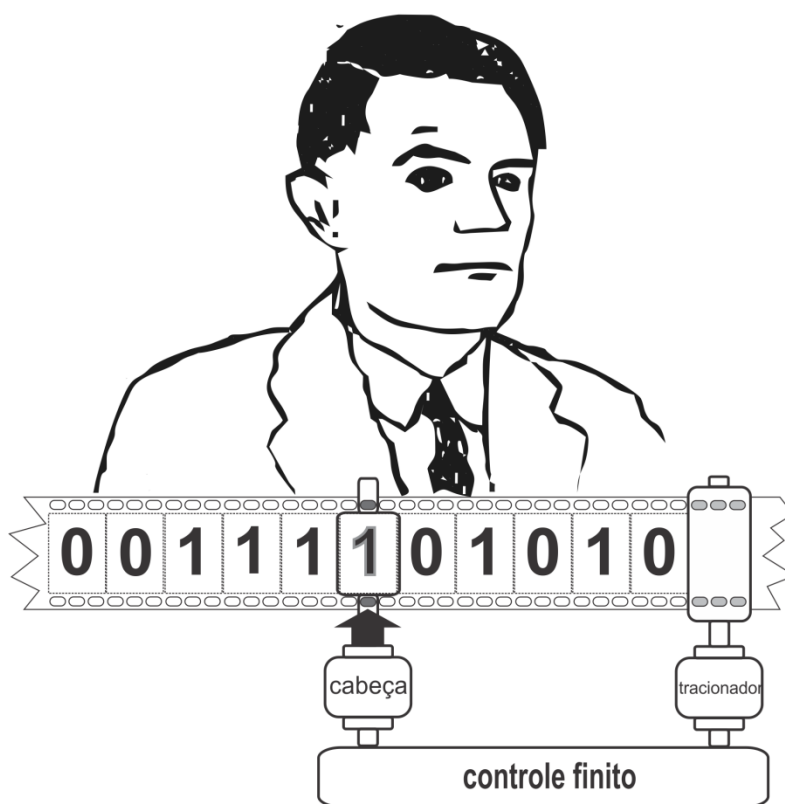


Figura 1: Modelo conceitual de da máquina de Turing.

O conceito de máquina de Turing é semelhante ao de uma fórmula ou equação. Assim, há uma infinidade de possíveis máquinas de Turing, cada uma correspondendo a um método definido ou algoritmo. Turing propôs que cada algoritmo, formalizado como um conjunto finito de instruções bem definidas, pudesse ser interpretado e executado por um processo mecânico.

Formalmente a máquina de Turing pode ser definida como uma máquina que contém:

- Um conjunto finito de estados Q com um estado inicial distinto,
- Um conjunto finito de símbolos Σ .

A interpretação e execução dos algoritmos são realizadas por estados e uma função de transição determina o novo conteúdo da fita. Desse modo, por restrição imposta ao algoritmo, pode-se alterar o conteúdo de apenas um quadrado por vez ou movimentar a cabeça, no máximo uma célula em qualquer direção. É permitida também a utilização de qualquer conjunto finito de símbolos para o alfabeto Σ , mesmo que a definição original tenha insistido em $\Sigma = \{0,1\}$. Esta mudança não tem impacto sobre a definição do conjunto de funções computáveis pela máquina.

O que torna uma máquina de Turing capaz de executar uma tarefa é a tabela de regras de transição que compõem o programa da máquina e um determinado *estado inicial*. O conjunto de instruções conhecidas e processadas pelo módulo de controle finito, ilustrado na Figura 1, é relacionado abaixo:

- IMPRIMA 0 NO QUADRADO QUE PASSA PELA CABEÇA
- IMPRIMA 1 NO QUADRADO QUE PASSA PELA CABEÇA
- VÁ UM QUADRADO PARA A ESQUERDA
- VÁ UM QUADRADO PARA A DIREITA
- VÁ PARA O PASSO i SE O QUADRADO QUE PASSA PELA CABEÇA CONTÉM 0
- VÁ PARA O PASSO j SE O QUADRADO QUE PASSA PELA CABEÇA CONTÉM 1
- PARE

Por exemplo: uma computação pode ser representada por três estados, nomeados s_0 , s_1 , s_2 e com algumas instruções formalizadas no Algoritmo 1:

1. $\langle s_0, 1, s_0, \gg \rangle$
2. $\langle s_0, 0, s_1, 1 \rangle$
3. $\langle s_1, 1, s_1, \ll \rangle$
4. $\langle s_1, 0, s_2, \gg \rangle$

Algoritmo 1: Exemplo de instruções entre três estados s_0 , s_1 e s_2 .

As duas primeiras instruções (linhas 1 e 2) descrevem o que acontecerá no estado s_0 . Há duas possibilidades: na primeira, a máquina faz a leitura de um dígito '1', movimentará a cabeça para a direita e permanecerá no estado s_0 . Na segunda, se for lido um dígito '0' a máquina deixará o estado s_0 , entrará no estado s_1 e escreverá o dígito '1' nessa transição. As instruções descritas nas linhas 3 e 4 mostram o que acontecerá no estado s_1 , ou seja, se for lido o dígito '1', a máquina movimentará a cabeça para a esquerda e permanecerá no estado s_1 . Se for lido o dígito '0', a cabeça será movimentada para a direita e a máquina passará para o estado s_2 . Como não há instruções definidas pelo algoritmo no estado s_2 , a máquina pára a sua execução (condição de parada) ao atingir este estado.

Quando estamos interessados em examinar o comportamento de uma máquina de Turing, é eficiente representarmos a máquina usando um diagrama de estados. A Figura 2 representa o conjunto de instruções neste formato visual.

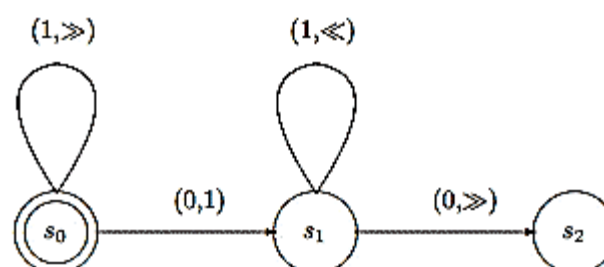


Figura 2: Um diagrama de estados representativo ao Algoritmo 1.

Nesta figura, os estados são representados por círculos, com um círculo duplo identificando o estado inicial. Uma transição é representada por uma aresta ou arco proveniente de um estado para outro ou para o mesmo estado. As arestas são rotuladas por pares (símbolo, ação), constituído primeiro pelo símbolo que deverá ser lido e depois, pela ação que deverá ser executada com a transição. Os símbolos pertencem ao alfabeto Σ e a ação será o símbolo a ser escrito, ou ainda « ou », indicando um movimento para a esquerda ou direita.

A Figura 2 ilustra uma máquina que calcula o sucessor de n , como uma execução simples para fins meramente demonstrativos. Assim, o estado inicial da fita representa o número n e após executar a sequência de passos, ele irá parar na configuração padrão que representa o número $n + 1$. Uma possível execução é demonstrada na Figura 3 (a-c).

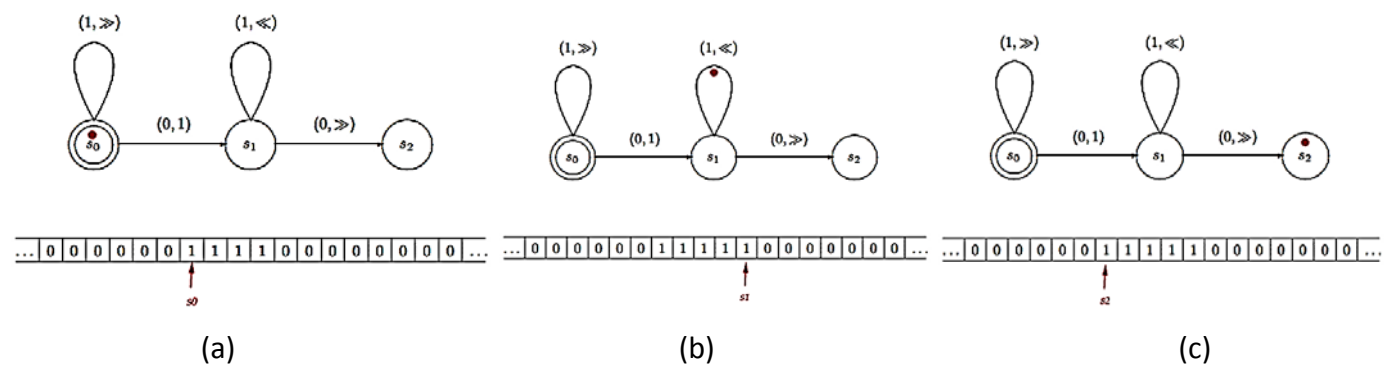


Figura 3: Sequencia de execução do algoritmo 1, a partir de um estado inicial.

Na atualidade é fácil relacionar um programa de computador com uma máquina de Turing e a tarefa mecânica de interpretação e execução obedecendo ao algoritmo. Assim, a máquina de Turing Universal incorpora o princípio essencial do computador: uma máquina simples que poderá executar qualquer tarefa bem definida, desde que especificada como um programa apropriado.

É possível encontrarmos também algumas simulações modernas para a conceitual máquina de Turing. A Figura 4 ilustra um modelo misto entre componentes digitais e mecânicos, encontrado em <http://aturingmachine.com>, com vídeo demonstrativo em <http://www.youtube.com/watch?v=E3keLeMwfHY>.



Figura 4: Simulação utilizando componentes digitais.

A Figura 5 ilustra uma simulação utilizando o kit de referência Lego, encontrado em <http://legoofdoom.blogspot.com>, com vídeo demonstrativo em <http://www.youtube.com/watch?v=cYw2ewoO6c4>.

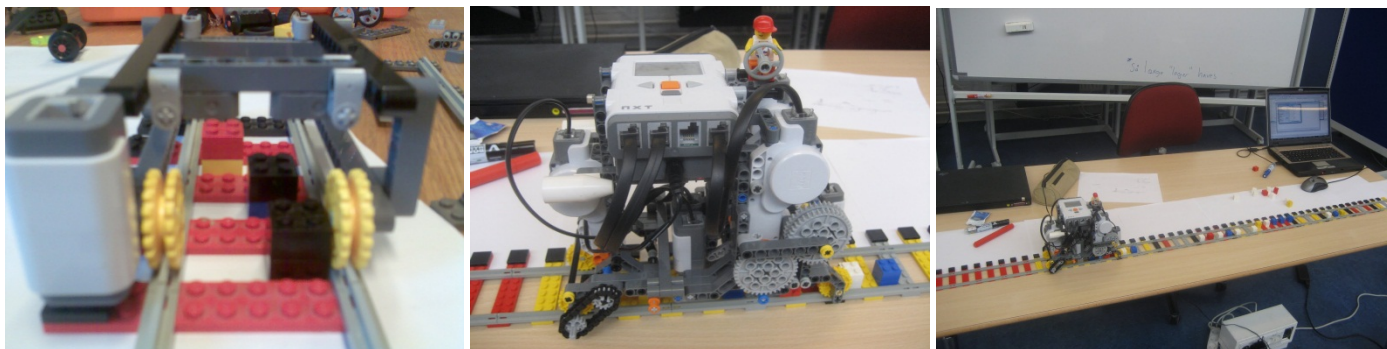


Figura 5: Simulação utilizando kit de referência Lego.

Turing provou que para qualquer sistema formal existe uma máquina de Turing que pode ser programada para imitá-lo. Era este sistema formal genérico, com a habilidade de imitar qualquer outro sistema formal, o que Turing procurava essencialmente. Tais sistemas chamam-se Máquinas de Turing Universais. O lógico matemático Alonzo Church chegou a definir: “Qualquer processo aceito por nós homens como um algoritmo é precisamente o que uma máquina de Turing pode fazer”.

Em abril de 1936, Turing mostrou seus resultados para John Von Neumann em Princeton, quando os computadores, no sentido moderno, ainda não existiam. Turing criou os conceitos e a fundamentação matemática, que nove anos depois seria a tecnologia utilizada para materializar os primeiros computadores eletrônicos, com grande participação de Neumann, ou seja, a transformação da lógica de suas ideias abstratas em engenharia real. Durante este período de tempo, Turing retornou à Inglaterra e a ideia viveu apenas em sua mente. A correspondência entre instruções lógicas, a ação da mente humana e uma máquina, que poderia ser fisicamente construída, foi a contribuição definitiva de Alan Mathison Turing.

Referências Bibliográficas:

Máquinas de Turing em Princeton, disponível em <http://introcs.cs.princeton.edu/java/74turing>, acesso em outubro de 2011.

Máquinas de Turing em Stanford, disponível em <http://plato.stanford.edu/entries/turing-machine>, acesso em outubro de 2011.