

Linguagem LOGO

GUILHERMINA LOBATO MIRANDA (*)

O LOGO

O LOGO é uma linguagem de programação concebida por Papert e col., em finais dos anos sessenta, no laboratório de inteligência artificial do Massachusetts Institut of Technology (MIT), especialmente para uso das crianças a partir dos 4-5 anos.

Precisamente por isso, assenta num simbolismo com significado para sujeitos que não possuem qualquer conhecimento informático, ao contrário do que sucede com a maior parte das outras linguagens de programação.

Ao criar esta linguagem, Papert procurou que ela fosse simples e poderosa, utilizável por crianças do pré-escolar e também por estudantes universitários, que induzisse a criação de programas interessantes quase desde o início da aprendizagem e permitisse desenvolver técnicas de resolução de problemas e capacidades de pensamento, convergente e divergente. O LOGO deveria ainda, segundo as intenções do seu criador, facultar às crianças um conhecimento delas próprias como aprendizes e pensadores, ou dito por outras palavras, possibilitar-lhes uma reflexão sobre o próprio acto de aprender e de pensar.

Por isso, o LOGO é considerado mais como uma linguagem com a qual se pode aprender do que uma linguagem que se aprende, visando

mais a aprendizagem de um processo (modos de pensar) do que a aprendizagem de conteúdos (o que pensar).

A estrutura do LOGO tem a ver com a trajectória científica de Papert, que se situou na confluência de duas correntes de investigação. Matemático e Físico, Papert interessou-se pelo desenvolvimento da inteligência humana depois de ter trabalhado com Piaget no Centro de Epistemologia Genética de Genebra. E foi precisamente ao investigar o problema da relação entre inteligência artificial e humana que haveria de desenvolver o projecto LOGO (Papert, 1963, 1968, 1976).

O LOGO tem, com efeito, como base uma epistemologia construtivista do processo de aprendizagem, assente no princípio de que o sujeito constrói as suas estruturas mentais em interacção com os objectos. O desenvolvimento cognitivo não consiste pois numa acumulação de factos, mas em progressivas reorganizações do conhecimento levadas a cabo pelo sujeito activamente envolvido no seu ambiente físico e social. Numa tal perspectiva a aprendizagem é importante para movimentar o processo de desenvolvimento, mas é, ao mesmo tempo, condicionada pelas capacidades de desenvolvimento daquele que aprende.

Retomando a concepção piagetiana de estádios de desenvolvimento, Papert sugere que certos conceitos que eram adquiridos tardiamente, podem ser aprendidos mais precocemente se o ambiente que envolve a criança for fértil em

(*) Assistente na Faculdade de Psicologia e de Ciências da Educação da Universidade de Lisboa.

materiais que lhe permitam assimilar esses conceitos de um modo natural.

Papert vai atribuir ao computador o poder de concretizar e personalizar aquilo que é formal. O computador poderia apoiar as crianças a fazerem mais rapidamente a passagem do pensamento infantil ao adulto, deslocando as fronteiras entre o concreto e o formal.

«Conhecimentos que só eram acessíveis através de processos formais podem agora ser abordados concretamente» (Papert, 1980, 1985, p. 37).

O criador do LOGO refere-se mais especificamente ao pensamento combinatório (onde se raciocina em termos de estados possíveis de um sistema) e ao pensamento auto-referencial (onde a reflexão se exerce sobre o próprio pensamento).

Sobretudo a partir da publicação de «Mindstorms: Children, Computers and Powerful Ideas» (1980), o LOGO transcendeu o seu estatuto de simples linguagem de programação tornando-se numa filosofia de educação.

A geometria da «tartaruga»

O LOGO permite operar com estruturas parciais, mas autónomas dessa linguagem. A geometria da «tartaruga» é a mais conhecida das suas componentes, constituindo a abordagem comum na iniciação ao LOGO.

Na geometria da «tartaruga», o papel central é desempenhado por um pequeno triângulo luminoso, a «tartaruga», que permite «pensar com» («To think with», Papert, 1980).

É a esta «tartaruga» assim definida que está associado um conjunto de instruções elementares, os comandos primitivos (Martí, 1980), também designados apenas por primitivas (Papert, 1980), que fazem apelo a movimentos naturais do ambiente humano, como avançar e recuar e virar à direita ou à esquerda.

No LOGO os projectos podem ser realizados passo a passo, sendo igualmente possível integrar as várias instruções em procedimentos. O sujeito pode atribuir-lhes um nome e utilizá-lo posteriormente noutros projectos mais complexos. Esta possibilidade de construir procedimentos à custa de outros procedimentos é que confere um grande poder a esta linguagem, reforçando ao mesmo tempo um tipo de

estratégia modelar na resolução de problemas, que consiste na decomposição de cada problema em sub-problemas mais simples.

Princípios da linguagem LOGO

Os princípios de base em que acenta esta linguagem de programação, segundo Larivée e Michaud (1980), podem resumir-se a seis: a noção de *estado*, de *procedimento*, de *nomear*, de *recursão*, de *bug*(1) e de *debugging*.

A *noção de estado*, refere-se a dois elementos indissociáveis do real, a saber: o tempo e o espaço. No plano espacial, o sujeito tenta controlar, desde as primeiras manipulações a direcção e posição da «tartaruga» através dos comandos primitivos do LOGO: dois associados à variável direcção, que são Direita (VD) e Esquerda (VE) e outros tantos associados à variável posição, que são Avançar (PF) e Recuar (PT).

A nível mais elaborado, trata-se de descobrir e precisar a evolução de um projecto no preciso momento da sua realização, o que permite ao sujeito situar-se nesse projecto tendo em vista a sua correcção (*debugging*) ou continuação.

No plano temporal, a decomposição de um projecto em procedimentos e a análise do projecto nos diferentes tempos da sua realização permite ao sujeito discernir o desenrolar temporal e as transformações ocorridas durante as diferentes sequências temporais.

A *noção de procedimento*: na realização de um projecto a noção de procedimento consiste na elaboração da sequência lógica de um percurso conducente ao objectivo a atingir. Constitui a dimensão estrutural que o sujeito pode utilizar para resolver um problema. Dimensão subjacente às noções de organização no espaço e no tempo, de planificação e de elaboração de estratégias.

O primeiro momento da realização de um projecto consiste na definição de um problema de aprendizagem para o sujeito. Num segundo momento, a planificação ilustrará o resultado desejado. Posteriormente, e de acordo com o nível de evolução do sujeito, há a elaboração

(1) Usamos o termo inglês, por ser difícil encontrar um termo em português que recubra o mesmo conceito.

dos meios para resolver o problema. Finalmente, o ensaio de concretização do processo que resulta na realização dos procedimentos, leva, entre outras coisas, à utilização de estratégias de correcção (*debugging*) que têm como finalidade atingir os resultados pretendidos. Trata-se de «pensar qualquer coisa que se quer realizar no computador e de o ensinar a fazê-la».

A estrutura do procedimento pode ser simples (isto é, a utilização de um procedimento que visa atingir um único objectivo) ou complexo (quer dizer, o sujeito utiliza um conjunto de procedimentos locais, que visam atingir objectivos múltiplos).

O conjunto destes objectivos parciais e final, corresponde às noções de sub-procedimento e procedimento. A diferenciação do objectivo global em sub-objectivos permite uma descrição mais clara dos meios a utilizar o que gera um conjunto de acções mais operacionais, permitindo uma maior generalização. Em termos piagetianos a estrutura de procedimento simples corresponde a um raciocínio intuitivo e a estrutura de procedimento complexo a um raciocínio de tipo operatório.

A linguagem LOGO está organizada com base em procedimentos. Para o seu criador, Papert, grande parte do pensamento humano pode ser estruturado em procedimentos. Ao permitir concretizar procedimentos e manipulá-los, o LOGO faculta aos sujeitos a possibilidade de explorar e jogar com os seus próprios procedimentos mentais (Papert, 1980).

A noção de nomear: a actividade de nomear consiste em dar um nome significativo aos procedimentos. Chamar QUADRADO a um quadrado é fácil; o mesmo já não acontece quando os procedimentos não coincidem com algo de conhecido. Encontrar um nome para dar a uma trajectória desconhecida (por exemplo, deslocar a «tartaruga» de um local do ecrã para outro com o objectivo de continuar a realização de um dado projecto) constitui uma actividade de nomeação. Esta actividade é um suporte representativo, dando a possibilidade de manipular «objectos» com a finalidade de atingir determinados objectivos. A escolha de um nome fornece indicadores sobre a compreensão do sujeito em relação a um fenómeno particular.

A noção de recursão: no plano teórico a

recursão consiste em constatar a repetição de um fenómeno num dado contexto. No plano prático, a recursão consiste na reaplicação de um conjunto de acções para resolver um problema. A recursividade encoraja o sujeito a organizar e a planificar estratégias de rendimento máximo e custo mínimo. No LOGO a recursividade consiste em utilizar procedimentos que se utilizam a si próprios como sub-procedimento.

A noção de bug: o termo *bug* quer dizer dificuldade, obstáculo, erro, designa uma falha na planificação e na elaboração de um projecto. No ecrã o *bug* corresponde a uma falha relativamente ao que tinha sido planificado, por outras palavras, uma inadequação entre o resultado desejado e o obtido. O conceito de *bug* difere do de erro na medida em que não é estático e sujeito a desaprovação. O *bug* é uma noção dinâmica e por isso pode ser utilizada para múltiplos fins. Os *bugs* são reveladores dos processos utilizados pelo sujeito na resolução de um dado problema.

Perante um *bug*, uma criança pode querer corrigi-lo para realizar o projecto inicial ou simplesmente mudar de projecto aproveitando o próprio erro. O *bug* possui o mesmo estatuto dinâmico atribuído ao erro na concepção piagetiana da aprendizagem (Inhelder, Sinclair & Bovet, 1974).

A noção de debugging: a noção de *debugging* está directamente relacionada com a noção de *bug*. O processo de *debugging* visa fundamentalmente o como e o porquê dum determinado fenómeno — reconhecendo e percebendo qual foi e elaborando as estratégias necessárias para a sua resolução. Esta atitude procura incentivar um efeito de generalização a situações posteriores e desenvolver a capacidade de analisar problemas de modo a estabelecer os meios mais eficazes para os resolver.

BIBLIOGRAFIA

- Inhelder, B., Sinclair, R. & Bovet, M. (1974). *Apprentissage et Structures de la Connaissance*. Paris: PUF.
- Larivée, S. & Michaud, N. (1980). L'ordinateur au secours de l'inadaptation, *Revue des Sciences de l'Éducation*, 6(3): pp. 451-472.

- Papert, S. (1963). Étude comparée de l'intelligence chez l'enfant et chez le robot. In *La filiation des structures* (L. Apostel, J.B. Grize, S. Papert & J. Piaget, Ed.). *Études d'Épistémologie Génétique*, Vol. XIV, pp. 131-194. Paris: PUF.
- Papert, S. & Voyat, G. (1968). A propos du perception. Qui a besoin de l'épistémologie?. In *Cybernétique et Épistémologie* (G. Cellier, S. Papert & G. Voyat, Ed.), *Études d'Épistémologie Génétique*, Vol. XXII, pp. 92-121. Paris: PUF.
- Papert, S. (1976). *An Evaluation Study of Modern Technology in Education*. Cambridge: Massachusetts Institute of Technology.
- Papert, S. (1980). *Mindstorms: children, computers and powerful ideas*. New York: Basic Books.