

Robison Cris Brito
Handrey Emanuel Galon

INTRODUÇÃO AOS AMBIENTES DE
PROGRAMAÇÃO NXT-G E leJOS
PARA O

LEGO MINDSTORMS



**INTRODUÇÃO AOS AMBIENTES DE
PROGRAMAÇÃO NXT-G E leJOS**

PARA O

LEGO

MINDSTORMS



Reitor: Luiz Alberto Pilatti. **Vice-Reitora:** Vanessa Ishikawa Rasoto. **Diretora de Gestão da Comunicação:** Mariangela de Oliveira Gomes Setti. **Coordenadora da Editora:** Camila Lopes Ferreira.

Conselho Editorial da Editora UTFPR. Titulares: Bertoldo Schneider Junior, Isaura Alberton de Lima, Juliana Vitória Messias Bittencourt, Karen Hylgemager Gongora Bariccatti, Luciana Furlaneto-Maia, Maclovía Corrêa da Silva, Mário Lopes Amorim e Sani de Carvalho Rutz da Silva. **Suplentes:** Anna Silvia da Rocha, Christian Luiz da Silva, Ligia Patrícia Torino, Maria de Lourdes Bernartt e Ornella Maria Porcu.

Editora filiada a



**Robison Cris Brito
Handrey Emanuel Galon**

**INTRODUÇÃO AOS AMBIENTES DE
PROGRAMAÇÃO NXT-G E IeJOS
PARA O **LEGO**
MINDSTORMS**

Curitiba
UTFPR Editora
2016

© 2016 Editora da Universidade Tecnológica Federal do Paraná.



Esta obra está licenciada com uma Licença Creative Commons - Atribuição-NãoComercial-SemDerivações 4.0 Internacional.

Esta licença permite o download da obra e o compartilhamento desde que sejam atribuídos créditos ao(s) autor(es), mas sem a possibilidade de alterá-la de nenhuma forma ou utilizá-la para fins comerciais.

Disponível também em: <<http://repositorio.utfpr.edu.br/jspui/>>.

Dados Internacionais de Catalogação na Publicação

B862 Brito, Robison Cris

Introdução aos ambientes de programação NXT-G e leJOS para o Lego Mindstorms. / Robison Cris Brito, Handrey Emanuel Galon. – Curitiba: Ed. UTFPR, 2016.

204 p. : il. color.

ISBN: 978-85-7014-169-9

1. Robótica. 2. LEGO (Brinquedo) – Inovações tecnológicas. 3. Dispositivos de lógica programável. 4. Programação lógica (Computação). 5. Inovações educacionais. 6. Tecnologia educacional. 7. Java (Linguagem de programação de computador). I. Galon, Handrey Emanuel. II. Título.

CDD (23. ed.) 629.892

Bibliotecária: Maria Emília Pecktor de Oliveira CRB-9/1510

Coordenação editorial

Camila Lopes Ferreira

Emanuelle Torino

Projeto gráfico, capa e editoração eletrônica

Vanessa Constance Ambrosio

Normalização

Camila Lopes Ferreira

Revisão gramatical e ortográfica

Adão Araújo

UTFPR Editora

Av. Sete de Setembro, 3165

80.230-901 - Curitiba – PR

www.utfpr.edu.br



DEDICATÓRIA

Dedicamos esta obra a um grande grupo de pessoas, em especial, nossos pais, irmãos, amigos, que sempre estiveram ao nosso lado, apoiando e incentivando o nosso nobre ofício de aprender para ensinar.

Dedicamos também aos amantes de novas tecnologias, em especial aos que desejam iniciar no mundo da robótica, mundo fascinante e que certamente os prenderá por muitos anos.





AGRADECIMENTOS

Queremos agradecer a um conjunto de pessoas que permitiram a transformação deste sonho em realidade.

Um obrigado muito especial à Universidade Tecnológica Federal do Paraná (UTFPR), na qual eu (Robison) estou envolvido diretamente desde 1996, quando entrei no curso Técnico em Eletrônica. Foram muitos trabalhos e desafios para me tornar professor desta instituição. E eu (Handrey), por me preparar para o mercado de trabalho no curso de Engenharia da Computação, e dar oportunidades, seja nos concursos em que participei como acadêmico, seja nos congressos que participei como autor de artigos.

Um agradecimento ao egresso do curso de tecnologia em Análise e Desenvolvimento de Sistemas da UTFPR, Iuri Menin, por acreditar no desenvolvimento de aplicativos para a plataforma LEGO Mindstorms, comprando com recursos próprios o primeiro robô e fazendo uma contribuição importantíssima para esta obra.

Agradecemos também a um grupo de professores e alunos da universidade. À professora Beatriz Borsoi e ao professor Fábio Favarim, por não medirem esforços na compra de novos kits de LEGO Mindstorms para a universidade, assim como à professora Beatriz por promover vários grupos de alunos para estudar e multiplicar o conhecimento referente à plataforma, envolvendo alunos do ensino básico, de ensino médio e de graduação.





SUMÁRIO

| | |
|--|-----------|
| 1 INTRODUÇÃO À ROBÓTICA | 13 |
| 2 LEGO MINDSTORMS | 21 |
| 2.1 KITS DE DESENVOLVIMENTO LEGO MINDSTORMS..... | 25 |
| 2.1.1 LEGO Robô 8547 Mindstorms NXT 2.0 | 26 |
| 2.1.2 LEGO Robô Mindstorms NXT 2.0 <i>Education Set Base</i> 9797 | 29 |
| 2.1.3 LEGO 31313 LEGO Mindstorms EV3 | 30 |
| 2.2 APRESENTANDO OS SENSORES DA PLATAFORMA | 34 |
| 2.2.1 Sensor de Luz (<i>Light Sensor</i>)..... | 35 |
| 2.2.2 Sensor de Toque (<i>Touch Sensor</i>)..... | 35 |
| 2.2.3 Sensor Ultrassônico (<i>Ultrasonic Sensor</i>) | 36 |
| 2.2.4 Sensor de Som (<i>Sound Sensor</i>) | 36 |
| 2.2.5 Sensor de Cor (<i>Color Sensor</i>) | 37 |
| 2.2.6 Servo Motores | 37 |
| 2.2.7 Outros Sensores | 38 |
| 2.3 PROCESSADOR NXT <i>BRICK</i> | 39 |
| 2.3.1 Portas de Comunicação..... | 39 |
| 2.3.2 O Visor | 42 |
| 2.4 AMBIENTES DE DESENVOLVIMENTO | 42 |
| 2.4.1 NXT-G | 43 |
| 2.4.2 leJOS..... | 44 |
| 3 AMBIENTE NXT-G..... | 47 |
| 4 PALETA DE COMPONENTES PADRÃO | 57 |
| 4.1 PRINCIPAIS COMPONENTES DA PALETA PADRÃO (<i>COMMON PALLETE</i>)..... | 60 |
| 4.1.1 Comando <i>Move</i> | 61 |
| 4.1.2 Comando <i>Sound</i> | 62 |
| 4.1.3 Comando <i>Display</i> | 63 |
| 4.1.4 Comando <i>Loop</i> | 65 |

| | |
|---|-----------|
| 4.1.5 Comando <i>Switch</i> | 67 |
| 4.2 EXEMPLO DE PROGRAMAS QUE UTILIZAM OS PRINCIPAIS COMPONENTES | 73 |
| 4.2.1 <i>Keep Forward</i> | 73 |
| 4.2.2 <i>Forward and Backward</i> | 75 |
| 4.2.3 <i>Forward and Backward Forever</i> | 76 |
| 4.2.4 <i>Avoid Stuff</i> | 77 |
| 4.2.5 <i>Keep Distance</i> | 79 |
| 4.2.6 <i>Line Follow</i> | 82 |
| 4.3. DEMAIS COMPONENTES DA PALETA PADRÃO..... | 84 |
| 4.3.1 Comando <i>Record/Play</i> | 84 |
| 4.3.2 Comando <i>Wait</i> | 85 |
| 5 PALETA COMPLETA (COMPLETE PALLETE) | 87 |
| 5.1 DATA HUB | 90 |
| 5.1.1 Ícones do <i>Data Hub</i> da Categoria <i>Common</i> | 92 |
| 5.1.1.1 <i>Data Hub</i> componente <i>Move</i> | 92 |
| 5.1.1.2 <i>Data Hub</i> componente <i>Record/Play</i> | 93 |
| 5.1.1.3 <i>Data Hub</i> componente <i>Sound</i> | 94 |
| 5.1.1.4 <i>Data Hub</i> componente <i>Display</i> | 96 |
| 5.1.2 Exemplo de um Aplicativo com <i>Data Hub</i> | 97 |
| 5.2 ACTION | 98 |
| 5.2.1 Enviar Mensagem (<i>Send Message</i>) | 98 |
| 5.2.2 Lâmpada (<i>Lamp</i>) | 100 |
| 5.3 SENSOR | 101 |
| 5.3.1 <i>Touch Sensor</i> | 103 |
| 5.3.2 <i>Sound Sensor</i> | 104 |
| 5.3.3 <i>Ligth Sensor</i> | 105 |
| 5.3.4 <i>Color Sensor</i> | 106 |
| 5.3.5 <i>Ultrasonic Sensor</i> | 108 |
| 5.3.6 <i>NXT Buttons</i> | 109 |
| 5.3.7 <i>Rotation Sensor</i> | 110 |
| 5.3.8 Componente <i>Timer</i> | 111 |
| 5.3.9 <i>Receive Message</i> | 111 |
| 5.4 FLOW..... | 113 |
| 5.4.1 <i>Stop</i> | 113 |
| 5.5 DATA | 113 |
| 5.5.1 Componente <i>Logic</i> | 114 |
| 5.5.2 Componente <i>Math</i> | 116 |
| 5.5.3 Componente <i>Compare</i> | 117 |

| | |
|---|------------|
| 5.5.4 Componente <i>Range</i> | 118 |
| 5.5.5 Componente <i>Random</i> | 120 |
| 5.5.6 Componente <i>Variable</i> | 121 |
| 5.5.7 Componente <i>Constant</i> | 122 |
| 5.6 <i>ADVANCED</i> | 125 |
| 5.6.1 <i>Number to Text</i> | 126 |
| 5.6.2 <i>Text</i> | 127 |
| 5.6.3 <i>Keep Alive</i> | 128 |
| 5.6.4 <i>File Access</i> | 128 |
| 5.6.5 <i>Calibrate</i> | 130 |
| 5.6.6 <i>Reset Sensor</i> | 131 |
| 5.6.7 <i>Bluetooth Connection</i> | 133 |
| 6 PALETA PERSONALIZADA (<i>CUSTOM PALLETE</i>) | 137 |
| 6.1 <i>MY BLOCKS</i> | 139 |
| 7 leJOS | 143 |
| 7.1 INSTALAÇÃO DO leJOS..... | 145 |
| 7.2. INSTALAÇÃO DO <i>PLUGIN</i> leJOS NXJ PARA O ECLIPSE..... | 152 |
| 7.3 CRIANDO UM PROJETO leJOS NO ECLIPSE..... | 156 |
| 7.4 INSTALANDO E EXECUTANDO UM APLICATIVO leJOS NO LEGO MINDSTORMS..... | 161 |
| 7.5 CLASSES leJOS..... | 161 |
| 7.5.1 <i>Class Motor</i> | 162 |
| 7.5.2 <i>Class SensorPort</i> | 162 |
| 7.5.3 <i>Class Ultrasonic Sensor</i> | 162 |
| 7.5.4 <i>Class Color Sensor</i> | 162 |
| 7.6 EXEMPLOS DE PROGRAMAS USANDO leJOS..... | 163 |
| 7.6.1 Andar até o Obstáculo..... | 163 |
| 7.6.2 Andar Aleatoriamente (<i>Random Move</i>)..... | 164 |
| 7.6.3 Manter Distância..... | 166 |
| 7.6.4 Gravando Arquivo de LOG..... | 167 |
| 7.6.5 Servidor e Cliente <i>Bluetooth</i> | 168 |
| 8 RESTAURANDO <i>FIRMWARE</i> LEGO MINDSTORMS..... | 173 |
| REFERÊNCIAS..... | 179 |
| APÊNDICES..... | 183 |
| APÊNDICE A - OPERAÇÕES LÓGICAS..... | 185 |
| APÊNDICE B - EXEMPLO DE PROGRAMA QUE GRAVA E REPRODUZ AÇÕES..... | 187 |





INTRODUÇÃO À ROBÓTICA



O termo robô tem origem em uma peça teatral do autor tcheco Karel Carpek, e data do início da década de 1920, de nome *Os robôs universais de Rossum* em inglês *Rossum's universal robots* (Figura 1). Robô (em tcheco, *robota*) significa **trabalhador forçado**, e esta ficção de Karel Carpek se refere aos robôs do cientista Rossum, criados para, obedientemente, servir à humanidade. O drama acontece quando estas **criaturas** passam a não mais gostar do papel de subserviência e se rebelam contra seus criadores. O autor da peça faz uso da imaginação para satirizar a forma de progresso técnico implantados na Europa pelos norte-americanos (PONTES, 2010).



Figura 1 – Representação de Rossum's universal robots

Fonte: Richardson (2013).

As primeiras referências de que se tem notícia sobre robótica datam de 350 a.C., do matemático grego Arquitas de Tarento. Amigo de Platão, Tarento teria criado um pássaro mecânico a que chamou **O Pombo** e que teria a capacidade de voar usando jato de ar comprimido. Contudo, foi Leonardo da Vinci quem elaborou o primeiro projeto de um robô humanoide, por volta do ano de 1495. Suas anotações, descobertas em 1950, continham desenhos detalha-

dos de um cavaleiro que aparentemente podia movimentar pernas e braços. O trabalho foi baseado em sua pesquisa da anatomia humana conhecida como homem vitruviano (PONTES, 2010), como se vê na Figura 2.

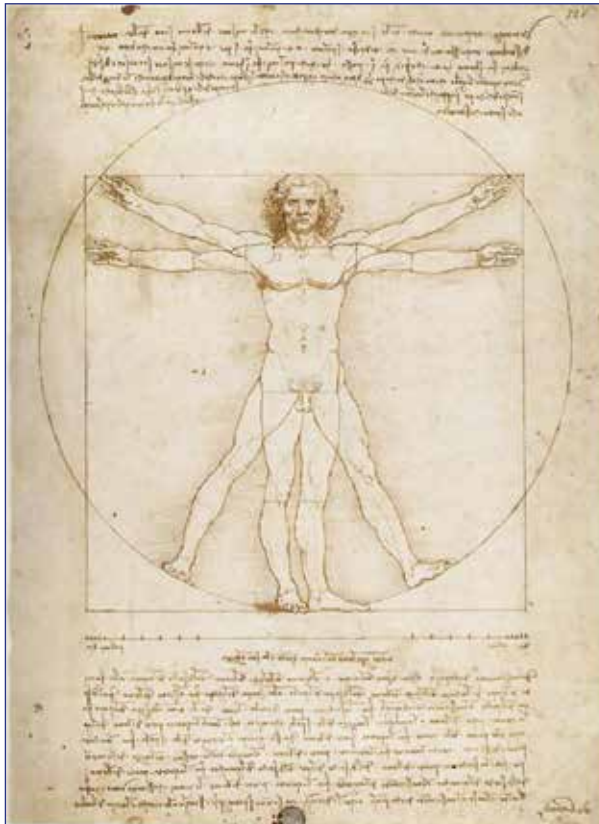


Figura 2 – Homem vitruviano, de Leonardo da Vinci

Fonte: World Mysteries (2011).

No século XVII, trabalhadores japoneses criaram um robô autônomo chamado **Karakuri** (Figura 3), que era capaz de servir chá (PONTES, 2010).

Outro exemplo de criatura mecânica é o pato de Jacques de Vaucanson (século XVIII) (Figura 4). Esse artefato ficou conhecido pela articulação realista de partes de seu corpo, por comer, digerir e defecar automaticamente. Vaucanson construiu ainda três outras criaturas humanoides: um tocador de mandolim que batia o pé, um pianista que simulava a respiração e movia a cabeça e um flautista (PONTES, 2010).



Figura 3 – Robô Karakuri, utilizado para servir chá

Fonte: Sentosa (2011).



Figura 4 – Pato de Jacques de Vaucanson

Fonte: Outer Places (2014).

Não obstante sua complexidade mecânica e a precisão com que desempenhavam suas tarefas, esses robôs, obviamente, não apresentavam capacidades cognitivas. O modelo de robôs que se conhece hoje só começou a ser desenvolvido com a chegada da computação e da inteligência artificial (PONTES, 2010).

Os primeiros passos nessa direção aconteceram em 1950, quando Alan Turing propõe, no artigo *Computing machinery and intelligence* (TURING, 1950), uma definição operacional de pensamento. Seu experimento, *imitation game*, sugere que, em vez de perguntar se uma máquina pode pensar, deve-se verificar se ela é capaz de passar em um teste de inteligência. Nesse teste, uma máquina é considerada inteligente se sua resposta for igual a resposta de um humano (PONTES, 2010).

O desafio para construir máquinas capazes de simular o comportamento cognitivo humano foi estudado por John McCarthy e Marvin Minsky, ainda na mesma década. No final dos anos 50, esses cientistas fundaram o *Artificial Intelligence Laboratory* do *Massachusetts Institute of Technology* (MIT), o primeiro laboratório dedicado à construção de robôs e ao estudo da inteligência humana (PONTES, 2010).

O objetivo principal de todos estes estudos envolvendo robôs inteligentes é permitir que estes desenvolvam tarefas feitas até então exclusivamente por humanos, já que existe uma série de vantagens ao utilizar robôs em tarefas específicas, como o aumento da precisão em tarefas manufaturadas, robustez, rapidez, uniformidade e suporte a ambientes hostis e ainda o incremento dos índices de qualidade e minimização de peças rejeitadas.

Pode-se ainda citar vantagens de fatores econômicos como a utilização eficiente de unidades de produção intensiva, o aumento de produtividade e a redução do tempo de preparação da fabricação. Também se tem benefícios no fator sociológico, que são a redução do número de acidentes, o afastamento do ser humano de locais perigosos para a saúde, redução de horários de trabalho, e ainda o aumento do poder de compra (diminui o custo dos produtos por eles construídos).

Ao contrário dos primeiros robôs, que eram mecânicos e muito limitados, os robôs atuais podem utilizar várias tecnologias presentes no dia a dia das pessoas, uma vez que estas tecnologias estão cada vez mais acessíveis do ponto de vista econômico, e mais fáceis de utilizar, do ponto de vista técnico.

Um exemplo é o uso da visão computacional nos robôs, sendo que estes podem capturar imagens do ambiente a partir de uma câmera, processar, e tomar decisões com base nestas imagens.

Outro recurso muito importante é o uso do *global positioning system* (GPS) ou sistema global de posicionamento, que permite ao robô identificar sua posição no globo terrestre e possibilitar sua navegação de forma autônoma. O GPS tradicional utiliza uma rede de satélite artificial, o que pode torná-lo impreciso, em especial, se o robô estiver em um ambiente coberto ou imerso, como por exemplo no oceano. Nesta situação, alternativas como o uso das antenas das operadoras de celulares ou ainda redes de boias submarinas podem ser adotadas.

Tecnologias como as apresentadas anteriormente permitem o desenvolvimento das chamadas *simultaneous localization and mapping* (SLAM) (que significa localização e mapeamento simultâneo). Esta técnica permite que robôs consigam mapear ambientes enquanto o utilizam.

Do ponto de vista do desenvolvimento de aplicativos para robótica, várias ferramentas para esta finalidade podem ser citadas, como a *Robot Operating System* (ROS, 2016), *Microsoft Robotics Developer Studio* (MICROSOFT, 2016), *Carnegie Mellon Robot Navigation Toolkit* (CARMEN, 2016) e *PlayerStage* (PLAYER, 2014). Todos são ambientes e/ou ferramentas que auxiliam no desenvolvimento de aplicativos para robôs ou redes de robôs, que utilizam, na maioria, a linguagem de programação C para o desenvolvimento.

Embora muito completos, esses ambientes não são indicados para iniciantes, ainda mais com pouco ou nenhum conhecimento em programação. Neste cenário, ambientes de programação *Drag and Drop* (ou seja, arrastar e soltar) se destacam, sendo utilizados componentes com funções específicas prontas, minimizando (e muito) a tarefa de programar.

Com a chegada das últimas gerações de computadores, onde recebe destaque LEGO Mindstorms (Figura 5), já se percebe a evolução no seu ambiente de desenvolvimento nativo, chamado de NXT-G, que é toda baseada no arrastar e soltar, fazendo com que algumas dificuldades antes encontradas como a complexidade para desenvolvimento de software sejam

facilitadas, sendo este um ambiente comumente utilizado para o ensino de robótica e linguagem de programação para crianças, jovens e adultos.

Assim, crianças a partir de 10 anos (como divulgado nas embalagens de LEGO Mindstorms), estudantes, acadêmicos e pesquisadores podem usufruir das vantagens desta plataforma, aprendendo conceitos complexos, como robótica, projeto de máquinas e veículos, assim como a programação.



Figura 5 – LEGO Mindstorms NXT 2.0

Fonte: Adami (2016).



Informações sobre a aquisição deste
livro podem ser obtidas pelo e-mail:
livraria@utfpr.edu.br

A UTFPR Editora tem por finalidade principal viabilizar a publicação de obras resultantes de atividades de ensino, pesquisa e extensão, além de documentos institucionais produzidos pela UTFPR. Visa ainda a publicação de obras originais ou traduzidas, de valor técnico, científico, artístico e literário de autores nacionais e internacionais.

Saiba mais em www.utfpr.edu.br/editora



Av. Sete de Setembro, 3165
Rebouças - CEP 80230-901
Curitiba - PR - Brasil

Telefone Geral
+55 (41) 3310-4545