

Aula Teste Robocode

Professor Mestre: Adilson Lopes Khouri

28 de fevereiro de 2019

Apresentação

O que é Robocode?

História do Robocode

Instalação do Robocode

Visão Geral do Jogo

Visão geral do Robô

Hands on

Exercício: Guerra de robôs!

Apresentação

- ▶ Adilson Khouri, jogador de Magic the Gathering, nerd, apaixonado por computação e machine learning!



Figura: Ministrando uma palestra no Peru e trabalhando na Argentina

Formação Acadêmica

- ▶ Bacharel em Sistemas de Informação (2011 - USP)
- ▶ Mestre em Sistemas de Informação (2016 - USP)
- ▶ Doutorando em Sistemas de Informação (cursando - USP)

Experiência Acadêmica

- ▶ Quatro estágios em docência na USP (duração de dois anos)
- ▶ Publicações Científicas
- ▶ Orientação de iniciação científica
- ▶ Disciplina: **Técnicas de programação em Games** (SENAC)
- ▶ Disciplina: **TCC 2** (SENAC)

Experiência de Mercado

- ▶ Programador na consultoria Arbit (2010-2011)
- ▶ Programador Itaú-Unibanco (2011-2013)
- ▶ Cientista de dados Sr. PagSeguro (2016 - 2018)
- ▶ Cientista de dados Sr. NuvemShop (Atual)
- ▶ Professor de Programação - SENAC (Atual)

E os senhores?

- ▶ Nome
- ▶ Graduação / pós-graduação
- ▶ Trabalho
- ▶ Qual sua experiência com os tópicos dessa disciplina?

Expectativas

- ▶ Quais expectativas?
- ▶ O que deve ser evitado?
- ▶ (E-Mail: 0800dirso@gmail.com)

O que é Robocode?

- ▶ É um jogo de programação cujo objetivo é codificar um robô (em linguagens como: Java e C#) que irá competir contra outros robôs
- ▶ O usuário não interage com o game durante a batalha, apenas durante a codificação do robô
- ▶ Universidades usam o robocode para ensinar programação e machine learning para os alunos [1]

História

- ▶ Desenvolvido por Mathew A. Nelson nos anos 2000 como um projeto pessoal. Levado para IBM pelo autor, foi visto como uma oportunidade divertida para incentivar as pessoas a aprenderem a programar em Java
- ▶ Foi inspirado pelo game Robot Battle desenvolvido por Brad Schick em 1992 o qual se inspirou no RobotWar um jogo desenvolvido para Apple II em 1980

História

- ▶ Se tornou um projeto *open source* em 2005
- ▶ Em 2006 virou um projeto no site SourceForge
- ▶ Em 2010 foi desenvolvido um plugin para .NET que permite desenvolver robôs nessa plataforma [2]

Instalação

- ▶ JRE ou JDK superior a versão 6, o robocode vem com um compilador interno rodando sobre o JRE
- ▶ A variável: JAVA_HOME tem que apontar para o diretório de instalação do JRE ou JDK
- ▶ Realize o download do arquivo jar aqui: [Source Forge](#)
- ▶ Na linha de comando, no mesmo diretório do arquivo jar, digite: **java -jar robocode-1.9.2.6-setup.jar**

Instalação

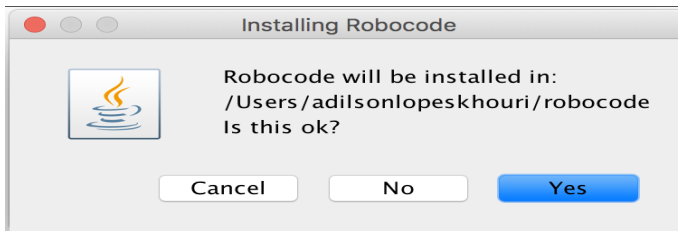


Figura: Passo 1 da instalação

Instalação

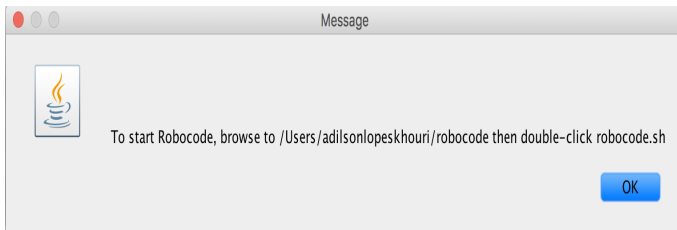


Figura: Passo 2 da instalação

Visão Geral do Jogo

- ▶ O campo de batalha funciona com um sistema de coordenadas cartesianas
- ▶ As rotações possíveis no jogo são sempre em sentido horário

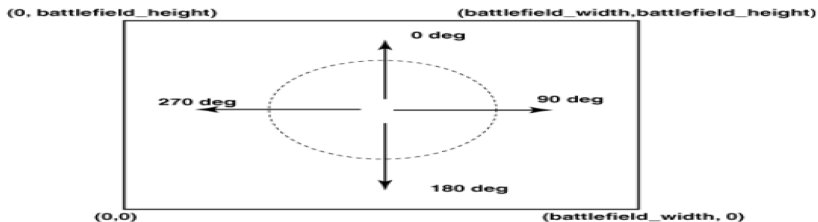


Figura: Campo de batalha baseado em sistema cartesiano [1]

Visão geral do Robô

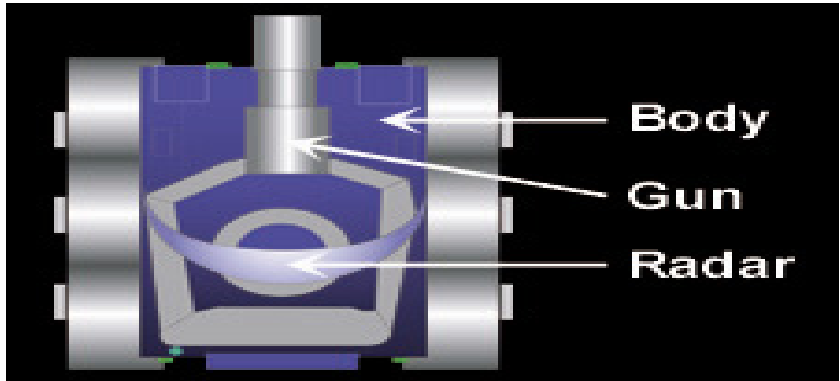


Figura: Anatomia do robô [1]

Visão geral do Robô

- ▶ *Body*: Sustenta a arma com o radar no topo, o corpo é usado para movimentar o robô para frente/trás e para rotacionar o mesmo
- ▶ *Gun*: Montada no *Body* é responsável por atirar bolas de energia. A arma pode ser rotacionada para: i) direta; ou ii) esquerda
- ▶ *Radar*: usado para escanear outros robôs durante o movimento do robô, pode ser movimentado para: i) direta; ou ii) esquerda. Esse radar lança o evento: *onScannedRobot()* [1]

Hands on

- ▶ Instale o Robocode em seu computador (consulte os slides de instalação se precisar)
- ▶ Execute o script: `Users \HOME_ DIR\robocode \robocode.sh`
- ▶ Clique em: Robot, source editor na janela principal.

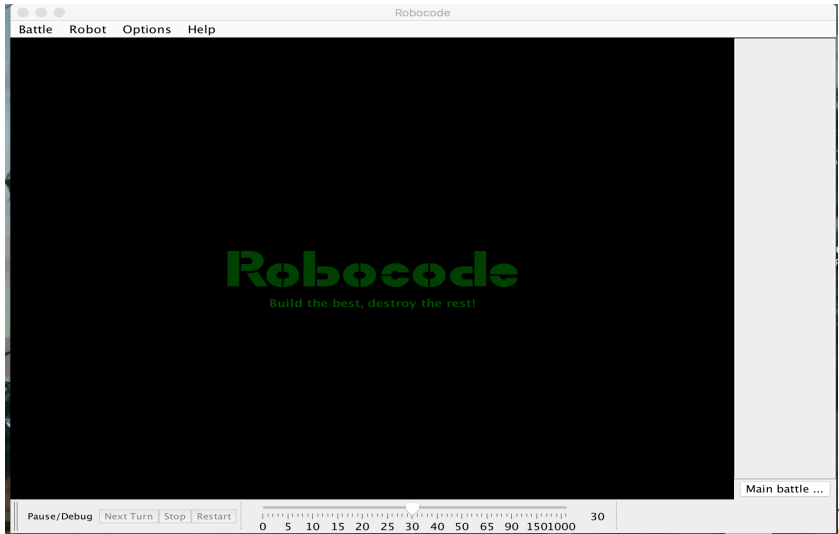
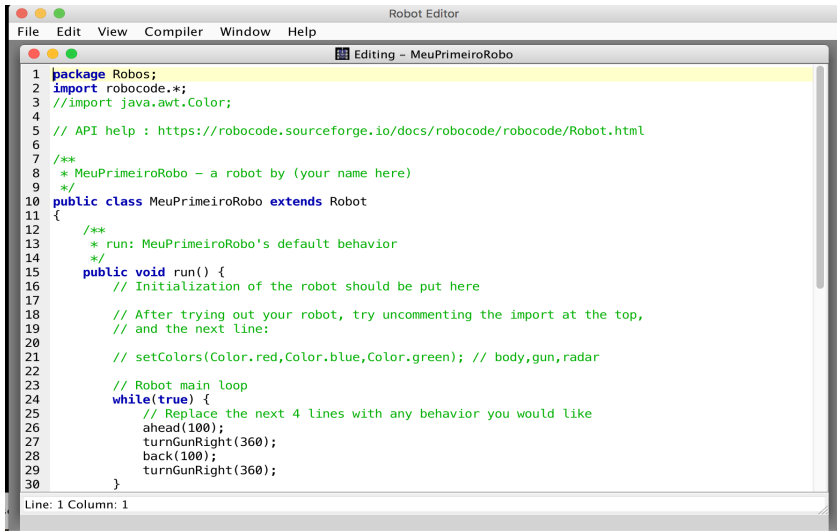


Figura: Tela Inicial

A screenshot of a Java IDE window titled "Robot Editor". The window has a menu bar with "File", "Edit", "View", "Compiler", "Window", and "Help". Below the menu bar is a toolbar with a small icon and the text "Editing - MeuPrimeiroRobo". The main area is a text editor showing Java code for a class named "MeuPrimeiroRobo". The code includes package declarations, imports, a class declaration extending "Robot", and a "run" method with various movement and color setting commands. Line numbers 1 through 30 are visible on the left side of the editor. The status bar at the bottom indicates "Line: 1 Column: 1".

```
1 package Robos;
2 import robocode.*;
3 //import java.awt.Color;
4
5 // API help : https://robocode.sourceforge.io/docs/robocode/robocode/Robot.html
6
7 /**
8  * MeuPrimeiroRobo - a robot by (your name here)
9  */
10 public class MeuPrimeiroRobo extends Robot
11 {
12     /**
13      * run: MeuPrimeiroRobo's default behavior
14      */
15     public void run() {
16         // Initialization of the robot should be put here
17
18         // After trying out your robot, try uncommenting the import at the top,
19         // and the next line:
20
21         // setColors(Color.red,Color.blue,Color.green); // body,gun,radar
22
23         // Robot main loop
24         while(true) {
25             // Replace the next 4 lines with any behavior you would like
26             ahead(100);
27             turnGunRight(360);
28             back(100);
29             turnGunRight(360);
30         }
```

Line: 1 Column: 1

Figura: Tela de codificação

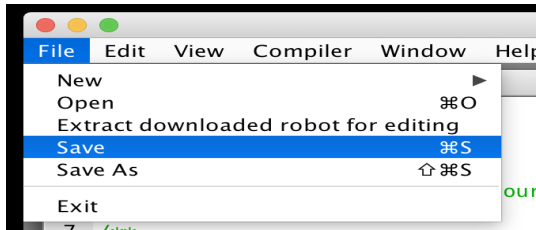


Figura: Tela para salvar

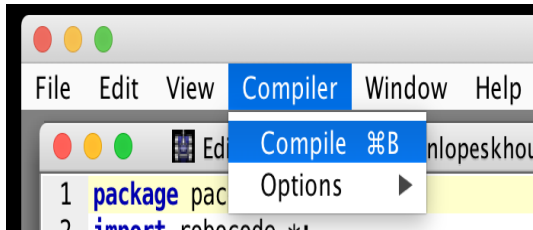


Figura: Tela para compilar

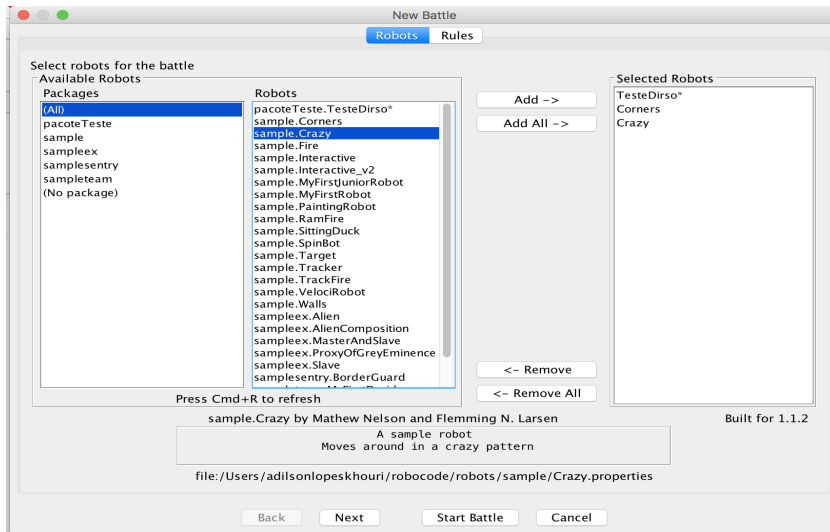


Figura: Tela para escolher robôs

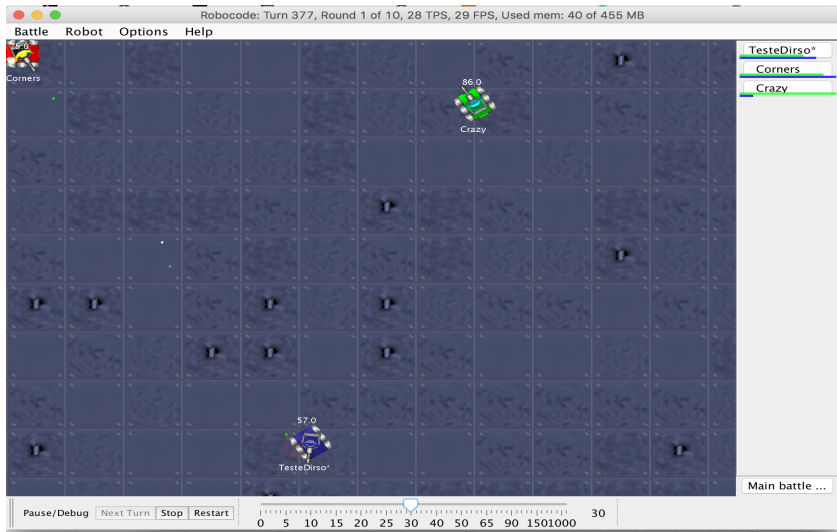


Figura: Campo de batalha


```
1 package Robos;
2 import robocode.*;
3
4 /*Essa classe eh seu roboh*/
5 public class MeuPrimeiroRobo extends Robot
6 {
7
8     public void run() {
9
10         // setColors(Color.red,Color.blue,Color.green);
11         // body,gun,radar
12
```

Code 1: Classe que representa o robô, herda da classe mãe *Robot*

```
13  // Robot main loop
14  while(true) {
15      // Replace the next 4 lines with any
16      //behavior you would like
17      ahead(100);
18      turnGunRight(360);
19      back(100);
20      turnGunRight(360);
21  }
22  }
```

Code 2: Laço principal que determina o comportamento do robô

```
24  /**  
25   * onScannedRobot: What to do when you  
26   *see another robot  
27   */  
28  public void onScannedRobot (ScannedRobotEvent e) {  
29   // Replace the next line with any behavior  
30   fire(1);  
31  }
```

Code 3: Evento do radar que identificou um robô

```
33  /**
34   * onHitByBullet: What to do when you're
35   * hit by a bullet
36   */
37  public void onHitByBullet(HitByBulletEvent e) {
38      // Replace the next line with any behavior
39      back(10);
40  }
```

Code 4: Evento do seu robô sendo alvejado

```
42  /**  
43   * onHitWall: What to do when you hit a wall  
44   */  
45  public void onHitWall(HitWallEvent e) {  
46   // Replace the next line with any behavior  
47   back(20);  
48  }  
49 }
```

Code 5: Evento do seu robô batendo na borda do campo de batalha

Exercício: Guerra de robôs!

- ▶ Codifique seu robô e faremos uma competição entre grupos de alunos!

Fontes de consulta robocode

- ▶ [README](#) do Robocode
- ▶ [Source Forge](#) do projeto
- ▶ [RoboWiki](#) com tutoriais
- ▶ [API](#) do Robocode
- ▶ [Competição de Robocode](#)
- ▶ [Livro](#) sobre Robocode

Dúvidas...

Alguma dúvida?

Contato

- ▶ E-mail: *0800dirso@gmail.com* (alunos SENAC)
- ▶ E-mail: *adilson.khour USP@gmail.com*
- ▶ Phone: +55119444 – 26191
- ▶ [Linkedin](#) do professor
- ▶ [Lattes](#) do professor
- ▶ Slides no [GitHub](#) do professor

Referências

- [1] F. N. Larsen. (2019) Robowiki. [Online]. Available: <http://robowiki.net/wiki/RoboWiki>
- [2] ——. (2019) Readme for robocode. [Online]. Available: <https://robocode.sourceforge.io/docs/ReadMe.html>