•万水编程革命系列•

Java语言学习利器

赵超鸿　周小刚　等编著

AI-CODE坦克机器人

中国水利水电出版社
www.waterpub.com.cn

万水编程革命系列

Java 语言学习利器
——AI-CODE 坦克机器人

赵超鸿　周小刚　等编著

中国水利水电出版社

内 容 提 要

本书是“万水编程革命系列”的第三本，把机器人设计与 Java 语言巧妙结合起来，让读者在躲避子弹、前进后退中全面掌握 Java 语言的变量、函数、容器、类、对象、继承、多态、文件流等各个要素，并学习到基础的数学和物理知识。本书由浅入深，侧重于 Java 的特性与机器人代码的实现。随着 Java 水平和机器人策略能力的提高，本书最后还介绍了人工智能及一些极具创意的机器人的 Java 设计。

本书章节设计合理，符合人脑思维定式，更易掌握各知识点。首先定出章节学习目标与任务，其次分析任务，穿插 Java 知识与机器人策略，并用代码实现，然后在任务的基础上进行知识点扩展，最后对各章节总结要点，并给出练习。

本书是编程爱好者、大中专学生加快学习的福音，更是计算机教师教学的强有力工具。本书可作为高等院校、高职高专、培训机构的教材。AI-CODE 已经被部分院校作为教学辅助软件使用。AI-CODE 必将改变传统教学，引领新的编程革命。

本书源代码可从中国水利水电出版社网站上免费下载，网址为：http://www.waterpub.com.cn/softdown/。

图书在版编目（CIP）数据

Java 语言学习利器：AI-CODE 坦克机器人 / 赵超鸿等编著. —北京：中国水利水电出版社，2007

（万水编程革命系列）

ISBN 978-7-5084-4530-4

Ⅰ. J... Ⅱ. 赵... Ⅲ. JAVA 语言—程序设计 Ⅳ. TP312

中国版本图书馆 CIP 数据核字（2007）第 044812 号

书　　名	Java 语言学习利器——AI-CODE 坦克机器人
作　　者	赵超鸿　周小刚　等编著
出版　发行	中国水利水电出版社（北京市　甲河路 6 号　100044） 网址：www.waterpub.com.cn E-mail：mchannel@263.net（万水） sales@waterpub.com.cn 电话　（010）63202266（总机）、68331835（营销中心）、82562819（万水）
经　　售	全国各地新华书店和相关出版物销售网点
排　　版	北京万水电子信息有限公司
印　　刷	北京市天竺颖华印刷厂
规　　格	787mm×1092mm　16 开本　24.75 印张　604 千字
版　　次	2007 年 4 月第 1 版　2007 年 4 月第 1 次印刷
印　　数	0001—4000 册
定　　价	42.00 元

凡购买我社图书，如有缺页、倒页、脱页的，本社营销中心负责调换

版权所有・侵权必究

前　　言

本书目标

至今还记得进入程序设计领域第一次在屏幕上用程序语言成功输出“Hello World”的喜悦；至今还记得无数日日夜夜枯燥的代码编写；却不记得一个小小的 Bug 伴随着自己度过了多少通宵。每每灯下感叹，这些代码能是游戏就好了。

终于过了编程设计的第一关，能独立写一些程序了，却要面对外面世界无数语言的选择，因为这些选择很可能就决定了自己未来的工作甚至幸福。然而铺天盖地的各方支持者都在自己的阵营摇旗呐喊：Java 好，C 好，C++好，C#……。于是每种语言都尝试了一下，几年也就过去了，仍然分不出谁好谁不好。现在终于明白具备扎实的基础和精通一门语言足矣。但心中还是偶尔会拿这些编程语言出来比比，有时还和其他阵营的高手过几招，但环境不同怎么也说不上谁写的程序更好，最后相互还是不服。

是否有一种软件能让大家边学边玩？是否有一种软件能让各语言高手同台竞技？实现我们征服程序世界的梦想？

AI-CODE 就是基于这种思想设计出来的。它以竞技游戏和比赛、任务驱动、寓教于乐的方式进行学习和思维训练。打破了传统教育的模式，利用游戏的互动性和操作性，让使用者在玩游戏、学编程中运用知识、提高能力。并且 AI-CODE 还实现了 Java、C、C++、C#等各种编程语言同台竞技。你有可能在闪躲炮弹和执行精确攻击的演练中学会数组、函数、对象、类。这个游戏即将为全世界的开发者实现这个愿望，它把游戏风潮变成了教学工具。

AI-CODE 为使用者提供了一个虚拟机器人的制作环境和竞技环境。首先，使用者通过自己编程，利用数学、物理等相关学科知识建立一个虚拟机器人。虚拟机器人通过图形化编写方式（机器人快车）或者纯代码编写方式（CodeCanvas）编写的机器人控制代码来体现。编写好的机器人控制代码经过编译以后，就可以在虚拟机器人运行平台（AIROBOT）竞技环境下和其它的机器人战斗到底。在这个过程中，为了取得胜利，你不断学习程序设计和机器人策略算法，通过学到的程序不断完善自己的机器人。在学习的同时感到了快乐，可是在娱乐的同时你发现自己目前的知识存储不能给你更高的支持，所以为了得到更多的快乐，你又投入学习。AI-CODE 就是这样一个系统，让你在玩—学习—玩—学习的循环过程中不断提高自己程序设计和策略算法设计的水平。AI-CODE 里有一些预先做好的机器人对手让你入门，一旦你不再需要它们，就可以把自己创建的机器人加入到正在世界范围内形成的某个联盟里去和世界最强手对阵。

编程革命系列打破了传统计算机程序设计教学方式的无趣、枯燥。利用 AI-CODE 游戏教育软件，让学习者在设计自己的坦克机器人的同时，学会 Java、C、C++、C#等编程语言。同时 AI-CODE 打破了语言的界限，各种语言设计的机器人可在同一平台下竞技，一决高低。学习者在与全国甚至全球 AI-CODE 爱好者的积极挑战中，不断改进完善自己的竞技机器人。在达到胜利的同时不知不觉牢牢地掌握了所用的程序设计语言，甚至包括数学、物理、人工智能

等各种知识。

希望本书帮助你通过 AI-CODE 这个游戏教育平台来进行一次愉快的程序和算法设计之旅，通过本书引导你体会到学习的乐趣，并在这个过程中学到所要学的东西，这就是我们的愿望。在这个过程中，我们会帮助你：

- AI-CODE 的使用——书中会简要地说明 AI-CODE 的使用。详细的内容读者可以参看 AI-CODE 的帮助文档，因为我们不想把这本书变成产品说明书。
- 通过设计 AI-CODE 机器人任务来学习 Java 语言的相关知识点。
- 学习机器人策略算法，并用 Java 语言详细实现每个讲解到的算法。
- 学习并了解 AI-CODE 所涉及的三角数学、几何数学、离散数学、物理及人工智能等综合知识。

阅读指南

本书章节设计合理，符合人脑思维定式，更易掌握各个知识点。只要是需要动手编程的章节，书中都会先定出章节学习目标与任务，其次分析任务，中间穿插 Java 知识与机器人策略，并用代码实现，然后在任务的基础上进行知识点扩展，最后对章节总结要点，给出练习题，并按 Java 语言和 AI-CODE 学习的难易程度分初级、中级、高级三大部分循序渐进。

本文是以 AI-CODE 两大主题之一——AI-TANK（智能坦克）进行讲解，所以在开始介绍部分用 AI-CODE 表示，后面都是以 AI-TANK 来说明。

第一部分为基础篇，全面介绍 AI-CODE 的基本特征，包括：游戏教育理念，AI-CODE 诞生、发展，AI-CODE 各种平台，AI-CODE 坦克主题涉及的物理和数学基础，AI-CODE 系统中的各种参数以及 Java 平台与环境的一些基本特性，最后带领读者快速体验 AI-CODE 的竞技平台并教读者设计自己的第一个机器人，体会到编程游戏所带来的无穷乐趣。

第二部分为中级篇，主要介绍 AI-TANK 中的三大基本运动方式和三大基本瞄准策略，并详细分析这些策略的原理与代码实现。同时还介绍 Java 语言的一些基本特性：运算符、变量、流程控制结构、表达式、函数。

第三部分为高级篇，介绍 AI-TANK 中的高级运动策略：反重力运动、振荡运动、因数避墙、躲避子弹，高级瞄准策略：模式匹配瞄准、振荡瞄准；信息收集与 Java 容器、类、对象、继承、多态、I/O 流、绘图。

第四部分为专家篇，介绍 AI-TANK 的一些专家特性，并对这些特性进行简单的分析和说明。如人工智能机器人中的强化学习、遗传算法等，机器人代码的混淆和破解，AI-TANK 中联盟机制的实现以及 AI-TANK 内核、接口的实现原理。

在附录中按章节给出例子机器人对应表、AI-TANK 对应的本章知识点，以方便读者随时查阅。

帮助资源

有关 AI-CODE 的详细使用，可以参考 AI-CODE 的帮助手册。在 AI-CODE 技术支持网站（http://www.ai-code.org）可以下载到 AI-CODE 最新的免费版本，同时在这个网站上每周都有新的技术文章发布。在学习的过程中还可以把自己写的机器人放到竞技平台上和全国乃至全世界的高手过招，从中发现自己的不足，不断改进。当然，如果你有任何问题都可以在 AI-CODE

试读结束，需要全本PDF请购买 www.ertongbook.com

技术支持网站的论坛上提问，上面有来自全国各地的 AI-CODE 爱好者帮你解答。

本书由赵超鸿、周小刚、蔡荣编写，参加编写工作的还有李杰、刘卓伟、刘婧、黄玲群、梁正武、王雪梅、梁湛业、李林玲、喻敬双、文惺、曹星、曹雪花等。

在此要感谢中南大学的周凤英老师，她不仅提出了宝贵意见，还亲自参与本书的编写工作。感谢广州中鸣数码科技有限公司的经理李军及所有员工对本书的支持，感谢 AI-CODE 工作室核心开发成员谢敏、蔡荣、陈静对本书的大力支持。更要感谢中国水利水电出版社计算机编辑室的编辑，感谢他们不断给我们提出宝贵的意见，以使本丛书能顺利出版。最后感谢 litter 对本丛书的默默支持。

由于作者水平有限，加之编写时间仓促，书中难免有疏漏之处，敬请读者批评指正。

编　者

2007 年 1 月

目　　录

前言

第一部分　基础篇

第 1 章　Java 平台与 AI-CODE 概述 1
1.1　教育理念 1
1.1.1　传统教育 1
1.1.2　游戏教育理念 1
1.1.3　游戏化学习 2
1.2　机器人编程游戏历史 2
1.2.1　AI-CODE 的诞生和发展 3
1.2.2　AI-CODE 游戏教育系统简介 4
1.2.3　AI-CODE 两大主题 5
1.3　程序和算法 6
1.3.1　面向对象程序设计思想 10
1.3.2　Java 语言简介及起源 11
1.3.3　Java 语言的概述 11
1.3.4　Java 平台、环境说明 14
1.3.5　Java 与 C 和 C++语言的异同 15
1.4　快速体验 15
1.4.1　下载、安装 AI-CODE 15
1.4.2　我的快乐竞技之旅 19
1.4.3　虚拟机器人运作平台——AIROBOT 20
1.4.4　图形编辑器——机器人快车 20
1.4.5　代码编辑器——CodeCanvas 21
第 2 章　AI-TANK 的数学与物理知识 22
2.1　数学知识 22
2.1.1　坐标系统 22
2.1.2　三角几何学与方向 23
2.1.3　离散数学 26
2.1.4　数学函数 27
2.2　物理知识 34
2.2.1　机器人解剖 35
2.2.2　机器人速度、距离、力 35

2.2.3 子弹能量、热、速度 38
2.2.4 Force 与动量守恒定理 39
2.3 基本参数 43
2.3.1 系统时钟 43
2.3.2 其他辅助参数 43
第 3 章 图形编程——快速步入程序之门 46
3.1 学习目标与任务 46
3.2 机器人快车概述 46
3.2.1 机器人快车的安装与卸载 47
3.2.2 机器人快车简介 47
3.2.3 流程模块拖拉编程 49
3.3 向战场中央靠拢机器人 54
3.4 机器人快车函数封装 57
3.5 小结与练习 59
第 4 章 我的第一个 Java 机器人 60
4.1 学习目标与任务 60
4.2 AI-CODE Java 环境设置 60
4.3 机器人开发 62
4.3.1 AI-CODE 开发目录设置 62
4.3.2 创建 First 机器人 62
4.3.3 机器人名称与 Java 包机制 68
4.4 FirstRobot 结构解析 69
4.4.1 Action 与 onTick 处理函数 69
4.4.2 常用函数 71
4.5 初识对象、类、与继承 72
4.5.1 面向对象思想 72
4.5.2 First 类及 AI-TANK 类图 73
4.6 知识扩展 75
4.6.1 关键字与注释码 75
4.6.2 经典例子机器人说明 75
4.7 小结与练习 76

第二部分 中级篇

第 5 章 基本运动与 Java 语言基础 77
5.1 学习目标与任务 77
5.2 基本运动策略 77
5.3 直线运动与 Java 基本元素 78
5.3.1 运动原理分析 78

5.3.2 绕墙走机器人剖析 79
5.3.3 数据类型、表达式、If-else 控制 81
5.4 指定方向或位置运动 84
5.4.1 运动原理分析 84
5.4.2 heading 与 bearing 方向解析 88
5.4.3 SuperCenter 机器人剖析 89
5.4.4 变量引用和代码规范 90
5.5 曲线运动 94
5.5.1 运动原理分析 94
5.5.2 圆周与倒 8 字运动机器人剖析 94
5.5.3 while 和 for 流程控制 98
5.6 扩展知识 100
5.6.1 变量与常量定义 100
5.6.2 三角函数与 Java 函数 101
5.6.3 switch 分支语句 102
5.6.4 运算符和赋值的概念 102
5.6.5 基本防御组合应用分析 103
5.7 小结与练习 104
第 6 章 基本瞄准与函数 105
6.1 学习目标与任务 105
6.2 基本瞄准策略 105
6.2.1 三大基本瞄准策略 105
6.2.2 相关事件的处理 105
6.3 定点瞄准 106
6.4 线性预测瞄准 107
6.4.1 提前量直线瞄准原理分析与正弦原理应用 108
6.4.2 提前量直线瞄准机器人剖析 110
6.4.3 直线迭代瞄准与 final 关键字 112
6.5 圆周预测瞄准与 Java 函数 114
6.5.1 圆周预测瞄准原理分析 114
6.5.2 圆周预测瞄准机器人剖析 117
6.5.3 Java 函数详析 119
6.6 小结与练习 121

第三部分 高级篇

第 7 章 战争情报员与容器 122
7.1 学习目标与任务 122
7.2 信息收集与数组 122

7.2.1 信息收集机器人 122
7.2.2 Java 数组 123
7.3 信息收集与集合 125
7.3.1 信息收集机器人 125
7.3.2 Java 集合 126
7.4 小结与练习 130
第 8 章 对象、类实现避弹避墙 131
8.1 学习目标与任务 131
8.2 躲避子弹机器人 131
8.2.1 躲避子弹原理 131
8.2.2 躲避子弹机器人剖析 132
8.2.3 子弹类实现 134
8.3 对象和类再析 135
8.3.1 类与对象 135
8.3.2 类的通用格式 136
8.3.3 深入研究 new 运算符 138
8.3.4 构造函数 139
8.3.5 方法重载 140
8.4 因数避墙机器人 143
8.4.1 因数避墙原理 143
8.4.2 因数避墙机器人剖析 144
8.4.3 因数避墙机器人的改进与扩展 145
8.4.4 final 关键字详析 147
8.4.5 static 关键字 149
8.5 扩展知识 150
8.5.1 Java jar 文件 150
8.5.2 对机器人进行管理 151
8.6 小结与练习 153
第 9 章 继承、多态实现高级机器人 155
9.1 学习目标与任务 155
9.2 高级运动策略与继承、多态性 155
9.2.1 高级运动策略分析 155
9.2.2 高级运动机器人实现 158
9.2.3 Java 继承与多态 164
9.2.4 反重力运动机器人剖析 168
9.3 高级瞄准策略 172
9.3.1 高级瞄准策略分析 172
9.3.2 振荡瞄准机器人 173
9.3.3 模式匹配瞄准机器人 178

9.3.4 统计学瞄准、虚拟子弹、波 184
9.3.5 其他瞄准算法 193
9.4 扩展知识 194
9.4.1 随机运动策略介绍 194
9.4.2 运动、瞄准简单组合再议多态 195
9.4.3 AI-TANK Vector2D 向量类 196
9.5 小结与练习 197
第 10 章 战略合成与 Java 接口、抽象 198
10.1 学习目标与任务 198
10.2 战略组合与 Java 接口和抽象 198
10.2.1 高手的秘诀：战略组合 198
10.2.2 设计可重用机器人结构 206
10.2.3 Java 抽象与接口 207
10.2.4 Java 机器人编程接口 213
10.3 Smoke 机器人与 Java 内部类 218
10.4 扩展知识——对象克隆 221
10.5 小结与练习 228
第 11 章 异常调试与 Record 机器人 229
11.1 学习目标与任务 229
11.2 调试机器人 229
11.2.1 AI-TANK 控制台 229
11.2.2 输出变量的值 231
11.2.3 AI-TANK 调试函数 234
11.3 Java 异常 234
11.4 Record 机器人 237
11.5 Java IO 238
11.6 小结与练习 241
第 12 章 团队作战实现 242
12.1 学习目标与任务 242
12.2 团队作战机制与通信兵 242
12.2.1 创建机器人团队 242
12.2.2 团队内的通讯 243
12.3 团队作战机器人 244
12.3.1 混战避敌原理 244
12.3.2 混战算法设计 246
12.3.3 混战避敌代码的实现 247
12.3.4 混战避敌算法改进与扩展 251
12.4 小结与练习 252

试读结束，需要全本PDF请购买 www.ertongbook.com

第 13 章　高级调试与绘图机器人 253
13.1　学习目标与任务 253
13.2　美丽的轨迹图 253
13.2.1　打开机器人图形控制 253
13.2.2　绘图机器人与监听器 254
13.3　机器人绘图类 258
13.3.1　AI-TANK 绘图类 258
13.3.2　Java 绘图类 Graphics2D 260
13.4　小结与练习 266

第四部分　专家篇

第 14 章　人工智能机器人与代码混淆 267
14.1　人工智能 267
14.1.1　人工智能概念 267
14.1.2　人工智能的研究和应用领域 268
14.1.3　AI-TANK 在人工智能领域的研究范围 270
14.2　强化学习机器人 271
14.2.1　原理概述 271
14.2.2　机器人设计分析 273
14.2.3　算法设计 278
14.2.4　代码实现 280
14.3　神经网络机器人 293
14.3.1　原理概述 293
14.3.2　机器人设计分析 298
14.3.3　反向传播算法设计 302
14.3.4　代码实现 308
14.3.5　与强化学习结合 319
14.3.6　神经网络 Java 公共包 319
14.4　遗传算法机器人 321
14.4.1　原理概述 321
14.4.2　预设策略进化机器人剖析 328
14.4.3　遗传操作机器人 330
14.4.4　中间解释程序进化机器人 332
14.5　机器学习机器人 336
14.5.1　机器学习原理 336
14.5.2　Bayesian 团队机器人设计 340
14.5.3　算法设计 341
14.5.4　代码实现 345

14.6 常用 Java 混淆与反编译 348
14.7 用 Eclipse 开发机器人 352
第 15 章 联赛系统、XML 与内部机制 354
15.1 联赛系统 354
15.2 机器人配置与启动原理 357
15.3 机器人编程接口与内核 359
15.4 XML 入门 367
第 16 章 AI-CODE 外传 370
16.1 策略流派 370
16.2 天才创意机器人 371
16.2.1 克隆机器人 Dolly 371
16.2.2 撞击攻击机器人 RamFire 372
16.2.3 跟踪者 Tracker 373
16.2.4 舞蹈机器人 Dancer 374
16.2.5 振动波绘图机器人 Shining 376
附录 1 章节机器人对照表 378
附录 2 知识点 379
参考文献 382

第一部分　基础篇

这部分全面介绍 AI-CODE 的基本特征，包括：游戏教育理念，AI-CODE 诞生、发展，AI-CODE 各种平台，AI-CODE 涉及的物理、数学基础，AI-CODE 系统中的各种参数以及 Java 平台与环境的一些基本特性，最后带领大家快速体验 AI-CODE 的竞技平台并教大家设计自己的第一个机器人，体会到编程游戏所带来的无穷乐趣。

第 1 章　Java 平台与 AI-CODE 概述

1.1　教育理念

1.1.1　传统教育

“传统教学法”是指以教师讲授为主的方式，在传统的教育体制下，师道尊严是维持师生关系的基础，老师高高在上，学生言听计从。几千年来的教育是以老师为主导的知识传授和学生被动接受的过程。在这种体制下，学生求真求新的天性被扼杀，创造力逐渐在负担沉重的课业中消磨殆尽。

传统教育注重的是传承人类创造的文明成果，而不是创造的过程，人们通过教育坐享前人的成果，受教育者是被动的知识接受者，而不是主动的探索者，教育与创造性几乎是毫不相干的，学生大多数是在被动地接受知识，而不是积极地发展智能。各种考试也主要是让学生回忆所学知识，因而学生常把记忆知识放在首位，这样他们就能顺利毕业。这种模式使学生习惯于教师用汤匙喂给他们知识，一旦独立面对问题，必然感到束手无策。这种教育模式往往存在着无趣、枯燥性，而且在很大程度上存在着单向性（也就是常说的填鸭式教学），教授者与被授者之间缺乏互动性，使得很大一部分的被授者对知识点的理解还停留在面上，普遍产生学习除了对考试有用外没有其他作用的厌学情绪，学完就又都交还给教授者了。

如何让被授者体会到学习的乐趣，并逐渐使被授者喜欢上学习呢？

1.1.2　游戏教育理念

网络游戏的迅速发展与教育软件的持续疲软形成明显的反差，人们一直在思考一个问题，为什么网络游戏有那么大的吸引力，为什么与孩子们不喜欢教育软件的现状有那么大的反差？诚然，网络游戏存在着一些如色情、暴力、游戏黏度过强导致玩家痴迷等问题，但是，在这些问题逐渐被社会所重视和逐步解决的时候，人们也在不断反思教育软件的死板、缺乏趣味性等缺点。如何把网络游戏的优点利用到国家的百年大计——教育上，成为新时期游戏设计者和教育工作者面临的挑战。

电脑游戏可以应用到严肃的教学中吗？英国学者朱迪通过 3 年的实验，证明了通过玩益智

类电脑游戏，孩子们的写作、沟通交流以及协作能力大大提高。

“游戏教育”是程艺软件提出的全新教育理念，旨在用一种寓教于乐的高效学习方法，利用人类“爱玩”的天性，让学习者在“玩”中学习，从而充分调动学习者的学习兴趣，使学习效果得到极大的提高。程艺软件公司积极把此理念溶入到产品、技术研发及团队管理中，努力帮助合作伙伴及顾客体会此理念带来的巨大学习效果，如图 1-1 所示。

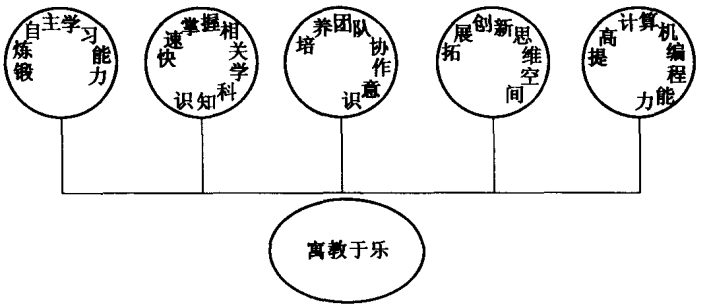

图 1-1 AI-CODE 涉及能力

1.1.3 游戏化学习

爱好游戏是人类的天性。但是沉溺于游戏，则会使人玩物丧志，影响学习、工作和生活，因此电子游戏曾经被教育界公认为“电子海洛因”。处理不好，网瘾也会成为社会问题。

如果将学生对游戏的痴迷转化为学习的动力，可以使教育的努力事半功倍。这正是游戏化学习研究的核心。

游戏化学习（Learn through play），又称为学习游戏化，就是采用游戏化的方式进行学习，是目前比较流行的教学理论和教育实践。游戏化学习是指在学习游戏化观念的指导下，在教学设计过程中就培养目标与发展、评价手段方面，就学习者年龄心理特征与教学策略等方面，借鉴游戏，设计、选择适当的发展工具、评价方法和教学策略。

目前可以归到游戏化学习范畴的教育教学实践活动，大致有这样一些：

- 幼儿阶段的各类品德发展活动、行为规范的学习活动、探索周围世界的学习活动。
- 小学低段（1～2 年级）的行为规范学习活动。
- 小学低段学科教学中的游戏性练习。
- 一些操作技能类学习的练习活动（如打字等）。
- 一些强调角色扮演的探究学习活动。
- 一些运用虚拟现实技术手段进行模拟实验的探究学习活动。

游戏化学习绝不仅仅是在学习过程中选用游戏性数码学习工具。有一位前苏联教育家曾经说过这么一句话“游戏对孩子而言就是学习，而学习对孩子而言是劳动”。

1.2 机器人编程游戏历史

机器人编程游戏指用各种编程语言设计自己的程序，并让这些程序在指定平台下以机器人或其他的外在形式表现出来，并在一定的规则下，让编程者可参与或观看到自己设计的机器人

程序在这个平台下竞技、交互或进行其他相关的游戏活动，从而达到游戏与学习的目的。

Worms 的 Commodore 64 版可以算是最早的一个编程游戏，它让使用者“教”一些东西在游戏中如何行动，然后看它们玩。其他的还有 C64 游戏 Mail Order Monsters。早期的这些编程游戏都很经典，但那时还没有图形，所以一切东西都必须可玩。这些编程游戏也成为后来图形编程游戏的规范：容易开始，玩起来有趣，并且让人很上瘾。

随着计算机软件硬件的发展，Brad Schick 在 1994 年创建了一个名为 Robot Battle 的经典图形编程游戏。它能让你创建你自己的机器人，并与其他机器人竞技，最终以胜负决定谁淘汰出局，这个平台利用了很多现实中的基本原理。比如数学中的三角学、物理中的能量定理等。而且平台给使用者提供了比赛排行榜，让每个人都可与世界各地的高手竞技比赛。

随着 Java 语言的出现，IBM 的 Mat Nelson 在这个语言上开发了 Robocode 这个 Java 平台的经典的编程游戏，从而把编程游戏带入了全新的时代。

早在 1995 年，也就是 Mat Nelson 开始在 IBM 做全职工作之前，他就开始使用 Java 语言了。随着这种语言的成熟，这个狂热的玩家看到了创建这个“他一直想玩”的游戏的机会。2000 年后期，Robocode 刚开始时只是个人的努力，2001 年 7 月以 alphaWorks 下载的形式带入 IBM 后就成了一种职业工作，短短半年这个游戏下载次数超过了 121000 次，它的成功也超乎作者的意料。

Robocode 是一个很容易使用的机器人战斗仿真器，可以在所有支持 Java 2 的平台上运行。你创建一个机器人，把它放到战场上，然后让它同其他开发者们创建的机器人对手拼死战斗到底。Robocode 里有一些预先做好的机器人对手让你入门，但一旦不再需要它们，就可以把你自己创建的机器人加入到正在世界范围内形成的某个联盟里去和世界最强手对阵。

每个 Robocode 参加者都要利用 Java 语言元素创建他的机器人，这样就使从初学者到高级黑客的广大开发者都可以参与这一娱乐活动。初级的 Java 开发者们可以学习一些基础知识：调用 API 代码、阅读 Javadoc、继承、内部类、事件处理等，高级开发者们可以在构建“最优品种”的软件机器人全球竞赛中提高他们的编程技巧。

1.2.1　AI-CODE 的诞生和发展

AI-CODE 最初是一些 Robocode 爱好者组成的兴趣小组，其前身为 RoboChina，即 Robocode 中华联盟。随着人们水平的提高，以及使用者的一些需求。RoboChina 成员发现 Robocode 是个很好的编程工具，真正能帮助大家学习到基础的编程知识。但是 Robocode 在国外比较流行，在国内还是一片空白，而且相关的中文支持也不够，最重要的是 Robocode 只支持 Java 一种语言的编程。以 Robocode 为思想开发出一种支持多种编程语言的编程游戏，让各种语言的爱好者在同一平台上竞技，学习。这将是很美好的事情。

于是 RoboChina 的核心成员：skyal.li、feynixs、iiley、xieming 开始了自己的编程游戏平台设计之路，也即 AI-CODE 的诞生。AI-CODE 延续了 Robocode 的思想，采用了全新的软件架构，一发布就得到了很多爱好者的支持，成功获得实体机器人厂家中鸣数码的支持，并成立了广州程艺软件有限公司。在技术上成功与中鸣数码的图形化编程平台机器人快车整合，完成了从图形编程到代码编程的初、中、高级学习过程，并同时支持 C、C++、C#、Java 等语言开发。

在 2003 年期间，此平台得到世界机器人足球竞赛组织大中华区负责人的认同。应他的要

求，AI-CODE 小组开发了坦克主题外的另一主题——足球，以适应世界足球比赛的要求。

现在 AI-CODE 开发小组正在全力打造新的内核，新功能以模块化管理，同时支持编程语言模块化、主题切换模块化。那时 AI-CODE 可轻松变换任何游戏主题，使用者更可把自己喜欢的语言对接到 AI-CODE 上。

1.2.2 AI-CODE 游戏教育系统简介

苏格拉底有句名言：教育不是灌输，而是点燃火焰！

AI-CODE 游戏教育平台的设计正是出于这种教育理念，它是一套全新的教育平台，以寓教于乐的游戏教育方式，打破了传统的教育模式，让大家在非压迫的竞争中不知不觉地提高自己的计算机程序设计水平和相关的数学、物理等学科知识的应用水平。用户通过机器人控制接口控制机器人的动作，编制好的机器人可以放到战场上和其他用户写的机器人同台竞技，在战斗的过程中感受到学习所带来的快乐，使用户在玩的过程中不断学习程序设计和机器人策略算法。在这个过程中，为了学习程序设计，你使用了 AI-CODE，因为它有趣、直观，在学习的同时你感到了快乐，可是在娱乐的同时你发现自己目前的知识存储不能给你更高的支持，所以为了得到更多的快乐，你又投入学习。AI-CODE 就是这样一种系统，让你在玩一学习一玩一学习的循环过程中不断提高自己程序设计和策略算法设计的水平。

这个过程是有趣而不是乏味的，由此带来与传统教学方式不同的效果是提高了被授者的学习兴趣并加深了对知识点的理解度。整个过程是一个自主的过程，而非被强迫的，如图 1-2 所示。

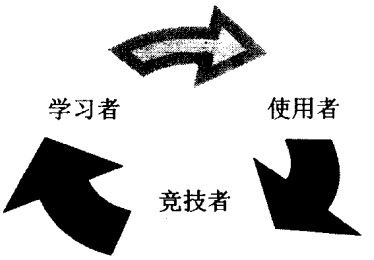

图 1-2 AI-CODE 学习三者的关系

- 学习者——学习程序和算法设计的人。
- 使用者——使用 AI-CODE 进行学习实践的人。
- 竞技者——将学习实践的成果发布出去与别的竞技者（有可能是你周围或者国内的其他选手，乃至世界各个国家的选手）进行比赛排名的人。

可以发现这是一个循环的过程，竞技者会再次变成学习者、使用者。是的，在和别人竞技的时候，你可能会发现别人成果的优点和自己的缺点，于是你通过再学习来对自己的成果进行改造或者重新设计，以期在下次比赛时取得更好的成绩。在竞技中提高自己知识水平、发现自己的不足。整个过程由于其主题的游戏性而充满了乐趣。

很多学生家长都有这样一种感觉，无论是电子游戏还是电脑游戏，都要让自己的孩子退而远之，因为觉得那是一种容易让人沉迷、玩物丧志的东西。其实，冷静地思考，我们能发现孩子们对游戏着迷的原因，就是因为游戏具有很大的互动性和操作性，游戏者能在游戏中展示自己。

试读结束，需要全本PDF请购买 www.ertongbook.com