

Wolfram 语言 入门



[美] 斯蒂芬·沃尔夫勒姆 (STEPHEN WOLFRAM) 著

WOLFRAM 传媒汉化小组 译

 科学出版社

Wolfram 语言 入门



[美] 斯蒂芬·沃尔夫勒姆 (STEPHEN WOLFRAM) 著

WOLFRAM 传媒汉化小组 译

科学出版社

北京

图字: 01-2016-4572号

内 容 简 介

Wolfram 语言代表了编程语言的一个巨大进步,使得每个人都能进行尖端的计算。因为拥有独特的构建丰富知识和自动化的方式,Wolfram 语言的规模可以是一行简单的交互式代码,也可以是拥有上百万行代码的产品系统。

本书提供了关于 Wolfram 语言和现代计算性思维的入门介绍。理解本书不需要编程相关的预备知识,适合于技术和非技术行业的大学生和高中学生,以及任何对最新科技和实际应用感兴趣的人士。

Authorized translation from the English language edition, entitled by Copyright © 2015.

All rights reserved. No part of this book may be reproduced or transmitted in any form or by any means, electronic or mechanical, including photocopying, recording or by any information storage retrieval system, without permission from the author.

CHINESE language print edition published by China Science Publishing & Media Ltd., Copyright © 2016.

图书在版编目(CIP)数据

Wolfram 语言入门/(美)斯蒂芬·沃尔夫勒姆(Stephen Wolfram)著;Wolfram 传媒汉化小组译.—北京:科学出版社,2016

书名原文:An Elementary Introduction to the Wolfram Language

ISBN 978-7-03-049962-2

I. ①W… II. ①沃… ②W… III. ①程序语言-基本知识 IV. ①TP312

中国版本图书馆CIP数据核字(2016)第227008号

责任编辑:赵彦超/责任校对:钟 洋
责任印制:肖 兴/封面设计:杰里米·戴维斯 直子·格洛瓦茨基

科 学 出 版 社 出 版

北京东黄城根北街16号

邮政编码:100717

<http://www.sciencep.com>

中国科学院印刷厂印刷

科学出版社发行 各地新华书店经销

*

2017年1月第 一 版 开本:787×1092 1/16

2017年1月第一次印刷 印张:21 1/4

字数:376 000

定价:98.00元

(如有印装质量问题,我社负责调换)

Trademarks: Wolfram, Wolfram Language, Wolfram|Alpha, Wolfram Cloud, Wolfram Programming Lab, Mathematica, Wolfram Workbench, Wolfram Knowledgebase, Wolfram Notebook, Wolfram Community, Wolfram Data Drop, Wolfram Demonstrations Project, and Tweet-a-Program.

该书是用 Wolfram 笔记本和 Wolfram 语言编写并制作。

前言

在过去 25 年多的时间里，我一直致力于为大家介绍 Wolfram 语言，终于是时候将我所掌握的知识精华撰写成书，以便读者自行阅读和学习。本书便是我努力的成果。

当我们在 1988 年首次发布 Mathematica（Wolfram 语言的前身）时，我出版过一本关于该系统的教程简介和参考指南。该书受到了广泛的好评，并且我认为其对 Mathematica 的早期成功贡献良多，这便是众所周知的 *The Mathematica Book*。在接下来的十多年中，该书经历了五次再版，内容也随之增加到了接近 1500 页的长度。

对于 *The Mathematica Book*，我的目标是能够系统性地覆盖 Mathematica 系统中所有的方面。但在 2007 年发布了一个大型新版本后，单独一本书显然已经不能容纳所有的内容。同时，我们的在线文档也在不断扩充，随着大量新范例在 2007 年的引入，印刷版将达上百万页。

Wolfram|Alpha 于 2009 年诞生，其独特的自然语言界面无需任何说明或文档。但接下来，对由 Mathematica 和 Wolfram|Alpha 融合而成的 Wolfram 语言，我们有必要再次给出文档并对其进行说明。

我相信 Wolfram 语言的在线文档（其打印后将超过 5 万页）对如何使用系统中所有各方面的功能给出了很好的解释。但对于初学者来说，他们更迫切地想要理解语言的原理——为了保持其连贯性和一致性，我付出了多年的心血。

Wolfram 语言在编程语言中独树一帜，并且在诸多方面与众不同。在不久前，我撰写了一本约 30 页的《快速编程入门》(wolfr.am/fifp)，为现代程序员讲解了 Wolfram 语言最基本的原理基础。

那么，完全不懂编程的人该怎么办？Wolfram 语言不仅为其打开了步入编程殿堂的大门，而且为其向当今的计算前沿快速跨越提供了难得的机会。

这也许是我们致力于 Wolfram 语言技术开发近 30 年全部努力的成果。我的目标是创建一种语言，使得任何人都能够轻松指定其想做的事情，而所有内部细节都会自动完成。

对于 Wolfram|Alpha 的快速问答而言，用日常英语提出你想知道的问题即可。但如果你想要完成更多系统性任务时，就需要用清晰准确的方式对其进行表述。Wolfram 语言便可在发挥功效。

那么,大家该如何学习 Wolfram 语言呢?沉浸式是其中一种方法:将自身置于使用 Wolfram 语言的环境中,探索可运行的程序,并将其作为范例从中学习。据我观察,当遇到问题时,只要有人可以从旁解答,哪怕只是偶尔能够帮助解释一下原理,你也会受益匪浅。

但如果想完全通过自学掌握 Wolfram 语言呢?对此,我认为需要进行系统性的介绍,渐进式地构建各个概念,并在学习过程中对常见问题给予回答。这便是我想在本书中做到的。

学习 Wolfram 语言与学习人类语言有些相似,必须把指令词汇和原理结合在一起进行学习。Wolfram 语言比人类语言更加系统化:无需记忆不规则动词,但仍然需要通过不断的练习才能熟练掌握。

我曾经有过疑惑,不知该如何撰写本书,但最终我还是决定大致按照我小时候学习拉丁语用的教科书那样编写。与日常语言不同,拉丁语无法沉浸式学习,除了逐步构建之外别无选择,这点和这本书一样。

在某些方面学习编程与学习数学有些类似,两者都有一定的准确性:结果只有对或错。但对于 Wolfram 语言,编程则更加具体:在每一步你都能够看到发生了什么,并知道你所做的是否正确。这里没有隐藏的、无法直接看到的、需要从外部进行抽象解释的概念。

尽管如此,数学教育历经两千年的发展,对算术、代数等的讲解顺序已经被逐步优化。而讲解 Wolfram 语言时遇到的问题却都是全新的,需要从零开始。现存的编程教育并没有太多帮助,原因是其教授的大部分内容都是在 Wolfram 语言中已被自动执行的非常底层的结构。

我把本书视为一次尝试:试图为学习 Wolfram 语言提供一种特定的途径。我不想涵盖语言中的所有内容,那将至少需要 5 万页才能完成。相反,我试图通过用一定数量的具体范例来解释语言的基本原理。

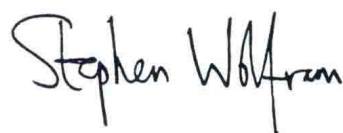
我选择了一些在实践中寓教于乐的范例,但重点在于这些范例覆盖了 Wolfram 语言中几乎所有的核心原理。并且,在了解这些原理后,你将可以通过查看具体的参考资料来理解语言在特定方面的功能。

毋庸置疑，Wolfram 语言拥有大量复杂的功能。其中一些，如图像中的对象识别，内部虽然复杂，但是解释起来并不难。不过其他一些内容，如计算格罗布纳基（Gröbner basis）——仍旧难以讲解，可能需要很多数学和计算机科学之外的知识。

我的目标是使读者可以完全自学这本书，不需要任何超越常识的知识。我尽量避免使用明显高于基础算术的数学，但是学习过高等数学的读者会注意到很多数学概念和本书概念中的关联。

当然这不是唯一一本关于 Wolfram 语言初级入门的书籍，我希望还会有更多的书籍。这本书循着一个特定的，同时许多方面是随意的路线，遍览了 Wolfram 语言的各种功能，并特意讲解了其中某些特性，但是还有许多同样重要、值得提及的特点没有涉及。

无论如何，我殷切期望这本书能够展现出我倾注了大半生心血的 Wolfram 语言的强大功能和魅力，同时让拥有不同背景的学生和各界人士，通过这本书开始了解 Wolfram 语言，并参与到定义当今时代特征的计算思维中来。

A handwritten signature in black ink that reads "Stephen Wolfram". The signature is written in a cursive, flowing style with a large initial 'S'.

什么是 Wolfram 语言？

Wolfram 语言是一种计算机语言 (computer language)，它为你提供了与计算机交流的方式，尤其是你可以告诉它要做什么。

计算机语言有很多种，比如 C++、Java、Python 和 JavaScript。Wolfram 语言基于知识 (knowledge based) 的特性让其独树一帜。这意味着它已经拥有了许多知识，因此你只需告诉它很少的内容就可以让它完成你想做的事情。

在这本书中，你将了解如何使用 Wolfram 语言来完成很多任务，并将学习如何用计算思维来思考要做的事，以及如何用 Wolfram 语言与计算机对话。

为什么不能用日常英语描述你想要做的事情呢？在 Wolfram|Alpha 中你就可以这样做，而且它对简短的提问非常有效。但对更复杂的任务而言，用日常英语来描述所有的内容便会变得不够实际。这正是 Wolfram 语言的用武之地。

Wolfram 语言的设计理念是充分利用内置的丰富知识，把描述想要完成的任务这件事变得尽量简单。更重要的是当你用 Wolfram 语言提出问题时，计算机马上能够理解你的意思并给出想要的答案。

我认为 Wolfram 语言是将构思变成现实的最佳工具。从你最初的构思开始，到将其用计算术语进行公式化，然后用 Wolfram 语言进行表述，最终让 Wolfram 语言尽可能自动化地完成。

你可以创建视觉、文字、互动或其他任何形式的内容；可以进行分析或探索；可以编写代码、创建应用程序和网站；在计算机、网页、电话、微型嵌入设备和更多地方实现各种不同的构思。

大概从 30 年前，我开始了对当今的 Wolfram 语言的构建。在整个过程中，特别是在 Mathematica 方面，Wolfram 语言得到了世界各国的研究机构 and 学校的广泛应用，并借助它实现了大量的卓越创新和发现。

如今的 Wolfram 语言已经具有崭新的意义：它是一种新型的通用计算机语言，重新定义了计算机的实际功能。在 Wolfram 语言的早期用户中，有许多是世界顶尖的开发人员和技术机构。还有许多大型的重要系统也都是用 Wolfram 语言构建的，比如 Wolfram|Alpha。

所有这些令 Wolfram 语言如此强大的知识和自动化，同时也使得这种语言可以为每个人所用。你不需要知道计算机如何工作，或关于技术以及数学的概念，这些都是 Wolfram 语言的职责。你唯一要做的就是了解 Wolfram 语言，以便告诉计算机你想要完成的任务。

通过学习本书，你将会了解 Wolfram 语言的原理，学习如何用 Wolfram 语言进行编程，并领会到其中蕴含的计算思维。但最重要的是，你将学习到一套将构思转换为现实的强大技术。Wolfram 语言在未来具有无限的创造力，让我们一起来期待——充分利用本书中学到的知识，尽情挥洒你的精彩创意，与我们一同开创未来！

实际运用 Wolfram 语言

实际运用是学习 Wolfram 语言的最好方法. Wolfram Programming Lab 是为能够方便学习该语言而特别设置的, 当然, 你也可以使用其他交互式 Wolfram 语言环境.

在这些环境中, 用 Wolfram 语言进行输入, 系统会马上对其进行计算并得到输出结果. 可以在桌面、网页或移动设备上输入. 在桌面和网页上, 一般按 `Shift + Enter` 代表你结束了输入; 在移动设备上, 一般按  按钮. 你的输入和输出序列, 连同你加入的任意文本, 全都存在 Wolfram 笔记本中.

In[1]:= `2 + 2`  输入
Out[1]= `4` 输出

在 Wolfram 笔记本中, 会有各种提示来帮助你输入 Wolfram 语言.

自动补全

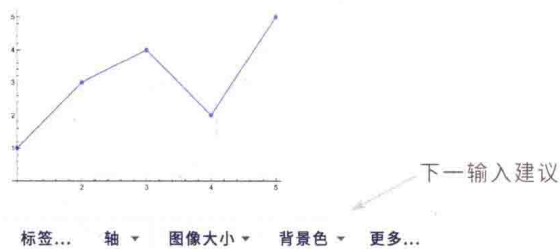
ListPl|

ListPlot (绘制点集)  参考信息

ListPlot3D (点集三维图)

ListPlay (点集播放)

 plot the sequence 1,3,4,2,5 自然语言输入



Table[Column[Range[n]], {n, 8}]

表格 列 范围 非英语环境中的代码说明

Wolfram 笔记本及其输入和输出的交互式序列, 是学习、探索和运用 Wolfram 语言编程的理想方式. 但是 Wolfram 语言也可以在自身交互界面之外的其他各种软件工程配置下进行操作. 而在内部, Wolfram 语言依然使用我们在本书中谈到的交互式配置进行相同的计算.

常见问题

在阅读这本书前我需要懂编程知识吗？

完全不需要。这是一本独立的编程入门书籍。

这本书适合于什么年龄的人阅读？

从经验上来看，凡 12 岁以上的读者都适合。我尽力选择了各年龄段读者都能理解且有趣的范例，成人也包括在内。

我需要掌握多少数学知识才能阅读这本书？

这本书没有任何超过基本算术的内容。这是一本关于用 Wolfram 语言编程的书籍，而不是一本数学书。

在阅读这本书的同时我需要使用电脑吗？

阅读这本书虽然不需要计算机，但通过使用 Wolfram 语言进行交互式尝试将会有很大的帮助，比如在 Wolfram Programming Lab 中进行练习。

我必须按顺序阅读这本书吗？

这样当然更好。这本书的结构是循序渐进地不断拓宽基础原理。如果进行跳跃性阅读，你很可能经常需要返回阅读前面的内容。

这本书的主题为何与其他编程书籍大相径庭？

因为 Wolfram 语言是一种与众不同且更高级的语言，自动摒弃了其他编程语言书籍专注的许多细节。

Wolfram 语言是一门教育性语言吗？

它确实被运用在教育（Mathematica 在各大校园中都被普遍使用）中，但同时在工业界也被广泛使用。它很适用于教育是因为其功能强大且简单易懂。

学习 Wolfram 语言会帮助学习其他语言吗？

当然会。Wolfram 语言会帮助你理解较高级的概念，之后你会发现这些概念以较低级的方式出现在其他语言中。

Wolfram 语言看起来太简单了，它真的可以用来编程吗？

当然可以。正是由于它自动化处理了在编程中可能遇到的枯燥操作，你将能够走得更远，理解得更深。

可以在教学中使用这本书吗？

完全可以。查看这本书的网页 (wolfr.am/ciwl) 了解更多辅助资料。

在教学中可以省略这本书中的某些章节吗？

这本书内容是以循序渐进的方式撰写的，因此如果省略了中间的章节，可能会需要补充部分内容。

习题是如何与正文相互关联的？

习题帮助检查和巩固理解，并且展示内容中的更深含义。

阅读这本书需要使用哪个版本的 Wolfram 语言？

10.3 以上的任何版本。请注意即使相对简单的范例（例如：`Table[x, 5]`），在更早版本中也有不能运行的可能。

这本书中的代码是“产品级别”吗？

通常是的。偶尔有些代码比实际需要稍微复杂些，因为写那些代码时，还没有讲到能使其简化的概念。

其他资源

Wolfram 语言主页

Wolfram 语言相关资源集锦

wolfram.com/language

Wolfram 参考资料中心

Wolfram 语言中所有函数的参考文档，并含有大量范例

reference.wolfram.com/language

Wolfram Programming Lab

提供对 Wolfram 语言的在线和桌面访问，并含有教育探索实例

wolfram.com/programming-lab

快速编程入门

面向有编程经验人士的 Wolfram 语言简要教程

wolfram.com/language/fast-introduction-for-programmers

Wolfram 竞赛

Wolfram 语言在线编程竞赛的动态集锦

challenges.wolfram.com

Wolfram Tweet-a-Program

大量少于 140 个字符的 Wolfram 语言程序范例

wolfram.com/language/tweet-a-program

Wolfram 演示项目

用 Wolfram 语言编写的 1 万多个交互式演示项目

demonstrations.wolfram.com

Wolfram 社区

学习和讨论 Wolfram 技术的在线社区

community.wolfram.com

Wolfram 主页

Wolfram Research 的官方主页，Wolfram 语言的缔造者

wolfram.com

Stephen Wolfram 的个人网页

本书作者的个人网页

stephenwolfram.com

目录

前言	iii	25 函数的应用方法	141
什么是 Wolfram 语言?	vii	26 纯匿名函数	147
实际运用 Wolfram 语言	ix	27 重复应用函数	153
其他资源	xi	28 判定和条件	163
1 开篇: 初等算术	1	29 纯函数详细介绍	171
2 函数简介	3	30 重新排列列表	177
3 初步认识列表	7	31 部分列表	187
4 显示列表	11	32 模式	193
5 列表的操作	15	33 表达式及其结构	197
6 制表	19	34 关联	205
7 颜色和样式	25	35 自然语言的理解	211
8 基本图形对象	29	36 创建网站和应用程序	217
9 交互式操作	33	37 布局和显示	227
10 图像	39	38 命名	233
11 字符串和文本	45	39 立即运算和延迟运算	239
12 声音	53	40 自定义函数	243
13 数组或由列表组成的列表	57	41 关于模式的更多内容	249
14 坐标与图形	63	42 字符串模式和模板	255
15 Wolfram 语言的范围	73	43 存储	261
16 现实世界数据	77	44 导入和导出	265
17 单位	87	45 数据集	271
18 地理计算	93	46 写出好的代码	283
19 日期和时间	101	47 调试代码	289
20 选项	107	更多实用功能	293
21 图和网络	115	后记: 作为一个程序员	301
22 机器学习	123	习题答案	303
23 关于数字	129	索引	311
24 更多可视化格式	137		

1 | 开篇：初等算术

作为了解 Wolfram 语言如何运转的第一个示例，让我们先来看看初等算术。

两个数相加：

```
In[1]:= 2+2
```

```
Out[1]= 4
```

```
In[2]:= 1234+5678
```

```
Out[2]= 6912
```

两个数相乘：

```
In[3]:= 1234*5678
```

```
Out[3]= 7006652
```

指令词汇

2+2	加法
5-2	减法
2*3	乘法（2 3 也可以）
6/2	除法
3^2	乘方（如平方）

习题

- 1.1 计算 1+2+3.
- 1.2 将 1 到 5 之间的整数相加.
- 1.3 将 1 到 5 之间的整数相乘.
- 1.4 计算 5 的平方（即 5×5 或 5 的 2 次方）.
- 1.5 计算 3 的 4 次方.
- 1.6 计算 10 的 12 次方（1 太拉）.
- 1.7 计算 2 的平方的平方的平方.
- 1.8 给 4-2*3+4 添加括号，使计算结果为 14.
- 1.9 计算 12 万与 123 之积.

常见问题

在 $2+2$ 等算式中，如何告诉 Wolfram 语言我的输入已经完成？

在计算机上，同时按住 `Shift + Enter`。在移动设备上，按  按钮。详情请见实际运用 Wolfram 语言一章。

为何乘法用 `*` 表示？

因为 `*`（“星号”，通常用 `Shift + 8` 键入）很像乘号。在 Wolfram 语言中，也可以在数字之间加空格表示相乘；Wolfram 语言将自动插入乘法符号 `x`。

“乘方”（`^`）是什么意思？

6^3 表示 $6 \times 6 \times 6$ （即 6 和它自身相乘 3 次）； 10^5 表示 $10 \times 10 \times 10 \times 10 \times 10$ 等等。

在 Wolfram 语言中的最大数是多少？

想多大就多大，只要计算机的内存允许。

Wolfram 语言中的运算顺序是怎样的？

与普通数学相同：乘方，乘法，加法。因此 $4*5^2+7$ 表示 $(4*(5^2))+7$ 。你可以像在数学中一样使用括号。（在数学上，人们有时会使用 `[...]` 或者 `(...)`。而在 Wolfram 语言中，`[...]` 另有其他意义。）

在做除法时，如何避免在答案中出现分数？

如果你输入的数字中有小数点，那么得到的数字中也会有小数点。也可以使用 `N`，我们将在第 23 章中讨论。

在结果中，如 `7 006 652`，出现了一些小空格，是什么意思？

这些空格是为了读数方便而显示的；并不是数的一部分。

如何输入一个大数？

只需键入数字，而不加任何逗号、空格或其他分隔符（如 `1234123511415223`）。

如果计算 $1/0$ ，会发生什么？

试试吧！你将得到无穷的符号表示，在它的基础上 Wolfram 语言可以进行更多计算。

更多资源

Wolfram Programming Lab 入门指南 (wolfr.am/eiwl-1-more)

2 | 函数简介

当你键入 $2 + 2$ 时，Wolfram 语言会把它理解为 `Plus[2, 2]`。Plus 是一个函数 (function)。Wolfram 语言中有超过 5000 个内置函数。算术使用的仅仅是其中的一小部分。

使用函数 Plus 计算 $3+4$ ：

```
In[1]:= Plus[3, 4]
```

```
Out[1]= 7
```

使用 Plus 计算 $1+2+3$ ：

```
In[2]:= Plus[1, 2, 3]
```

```
Out[2]= 6
```

函数 Times 可进行乘法运算：

```
In[3]:= Times[2, 3]
```

```
Out[3]= 6
```

可以把函数嵌套在其他函数内部：

```
In[4]:= Times[2, Plus[2, 3]]
```

```
Out[4]= 10
```

Wolfram 语言中的所有函数均使用方括号，且名称以大写英文字母开头。

函数 Max 求的是一组数的最大值。

这三个数的最大值是 7：

```
In[5]:= Max[2, 7, 3]
```

```
Out[5]= 7
```

函数 RandomInteger 随机取一个整数，范围在 0 到你所指定的大小之间。

在 0 到 100 之间取一个随机整数：

```
In[6]:= RandomInteger[100]
```

```
Out[6]= 71
```

每次运算，都会得到一个不同的随机数：

```
In[7]:= RandomInteger[100]
```

```
Out[7]= 1
```

指令词汇

Plus[2, 2]	2+2	相加
Subtract[5, 2]	5-2	相减
Times[2, 3]	2*3	相乘 (2 3 也可以)
Divide[6, 2]	6/2	相除
Power[3, 2]	3^2	乘方
Max[3, 4]		最大值
Min[3, 4]		最小值
RandomInteger[10]		随机整数

习题

- 2.1 使用函数 Plus 计算 7+6+5.
- 2.2 使用 Times 和 Plus 计算 2×(3+4).
- 2.3 使用 Max 求 6×8 和 5×9 的较大值.
- 2.4 给出一个 0 到 1000 之间的随机数.
- 2.5 使用 Max 和 RandomInteger 生成一个 10 到 20 之间的数.

常见问题

是否必须键入 Plus、RandomInteger 等函数名称中的大写字母？
是的. 在 Wolfram 语言中, plus 与 Plus 不同. Plus 中的大写字母意味着你使用的是内置 (“官方”) 加法函数.

在使用函数时, 是否必须键入方括号 [...]？
是的. 方括号 [...] 用于函数; 圆括号 (...) 用于分组, 如在 2*(3+4) 中, 而不用于函数.

如何朗读 Plus[2, 3]？
通常读作 “plus of 2 and 3”; 有时读作 “plus of 2 comma 3”. “[” 可以读作 “open bracket (左方括号)” ; “[” 读作 “close bracket (右方括号)” .

为何使用 Plus[2, 3] 而不是 2+3？
对于 Plus 来说, 这并不是必须的. 但对于绝大多数函数, 比如 Max 或 RandomInteger, 不存在类似于 + 的特殊形式, 因此必须给出它们的名称.

Plus[...] 和 + 能混合使用吗？
可以. Plus[4+5, 2+3] 的输入是有效的, Plus[4, 5]*5 也是如此.

如果 Wolfram 语言将输入变成了红色, 这是什么意思？
这意味着你键入了 Wolfram 语言不能理解的内容. 更多信息, 请参阅第 47 章. 首先要检查的是左右方括号是否匹配.

技术要点

- Wolfram 语言中的表达式(expression) (参阅第 33 章) 由函数的嵌套树组成.
- Plus 可以将任意数目的数相加, 但 Subtract 只能将一个数从另一个数中减去 (以避免将 $(2-3)-4$ 和 $2-(3-4)$ 混淆).
- 与传统数学或计算机科学相比, 在 Wolfram 语言中函数的概念大为拓展, 更具一般性. 例如, $f[anything]$ 可以被看作是一个函数, 无论它有一个明确的计算结果, 还是仍保持符号形式.

更多资源

Wolfram 语言: 数学函数 (wolfr.am/eiwl-2-more)