

信息技术

五年级 上册

主 编：陈军

副主编：赵飞

清华大学出版社
北京

学术顾问：雍俊海
总主编：刘军
副总主编：谢作如 夏正仁 陈军
本册主编：陈军
本册副主编：赵飞
主要编写人员：赵立双 蒋碧莹 商亮 罗柳垠 艾瑞婷 杨雪生 孙磊

版权所有，侵权必究。举报：010-62782989，beiqinquan@tup.tsinghua.edu.cn。

图书在版编目(CIP)数据
信息技术·五年级上册 / 陈军主编. — 北京：清华大学出版社，2024.8. — ISBN 978-7-302-66892-3
I. G624.581
中国国家版本馆 CIP 数据核字第 2024BU8294 号

责任编辑：赵轶华
封面设计：王静 薛玉斌 张思宇
责任校对：赵琳爽
责任印制：沈露

出版发行：清华大学出版社
网 址：<https://www.tup.com.cn>，<https://www.wqxuetang.com>
地 址：北京清华大学学研大厦 A 座 邮 编：100084
社总机：010-83470000 邮 购：010-62786544
投稿与读者服务：010-62776969，c-service@tup.tsinghua.edu.cn
质量反馈：010-62772015，zhiliang@tup.tsinghua.edu.cn
课程资源：<https://wqbook.wqxuetang.com/qhytl/>

印装者：新疆新华印务有限公司
经 销：全国新华书店
开 本：185mm×260mm 印 张：5.75 插 页：4 字 数：87 千字
版 次：2024 年 8 月第 1 版 印 次：2024 年 8 月第 1 次印刷
定 价：11.34 元

产品编号：109047-01

前言



亲爱的同学们：

你们对计算机、手机和平板电脑里呈现的神奇世界感到好奇吗？那些精美的图文、有趣的视频和丰富的知识都是怎么来的？悄悄告诉你们，这一切都离不开信息科技！

在这个信息爆炸的时代，信息科技如同“魔法”般改变着我们的世界，它让学习变得更加便捷，也让生活变得更加多彩。从早上起床的手表提醒，到上学路上的导航指引；从课堂上的“双师”教学，到放学后的在线学习；从周末与家人的视频通话，到节日里的电子贺卡传递祝福……信息科技的身影无处不在。

作为互联网时代的“原住民”，同学们需要掌握一定的信息科技知识，才能更好地适应这个时代的快速发展。因此，我们精心编写了这套《信息科技》教材，希望它能够成为你们探索信息世界的“指南针”，带领你们走进信息科技的殿堂，解锁生活中的无限可能，发现更多令人惊喜的奥秘。



这套充满乐趣、智慧和挑战的“魔法”之书，将带领你们在探索中发现，在发现中成长。无论你是对互联网感兴趣的小小网民，还是对编程充满好奇的小小程序员，或是想探索虚拟世界的小小探险家，这里都能满足你的需求。你们将像“探险家”一样，跟随我们的脚步，踏上一场充满乐趣和惊喜的“冒险”之旅。本册教材包括初识算法、描述算法、算法的控制结构三个单元，通过学习，你们将了解算法在生活中的应用，学会用多种方式描述算法，理解算法的三种基本结构，运用算法思维解决问题。

信息科技并不只是冷冰冰的计算机和代码，它更是一门充满活力和创造力的学科。学习信息科技，是一件既有趣又有意义的事情。在这本书中，你们或许会遇到一些困难和挑战，但只要保持好奇心和求知欲，勇敢地面对，就一定能够找到解决问题的方法，收获满满的成就感。

亲爱的同学们，让我们一起开始这段奇妙的旅程吧！愿你们在学习中收获知识、收获快乐、收获成长！在学习的过程中，你们一定会成为掌握信息科技的小小“魔法师”，用你们的智慧和勇气创造出更加美好的明天！

——编委会的大朋友们

目 录



1 初识算法 1

- 第 1 课 走进算法.....3
- 第 2 课 算法的工作过程和特点.....10
- 第 3 课 算法的应用.....16
- 第 4 课 活动：计算大比拼.....21

2 描述算法 29

- 第 1 课 用自然语言描述算法.....31
- 第 2 课 用流程图描述算法.....37
- 第 3 课 用程序描述算法.....45
- 第 4 课 活动：数字加密.....51

3 算法的控制结构 61

- 第 1 课 顺序结构.....63
- 第 2 课 分支结构.....68
- 第 3 课 循环结构.....74
- 第 4 课 跨学科活动：绘制莲花.....80

清华大学出版社

1

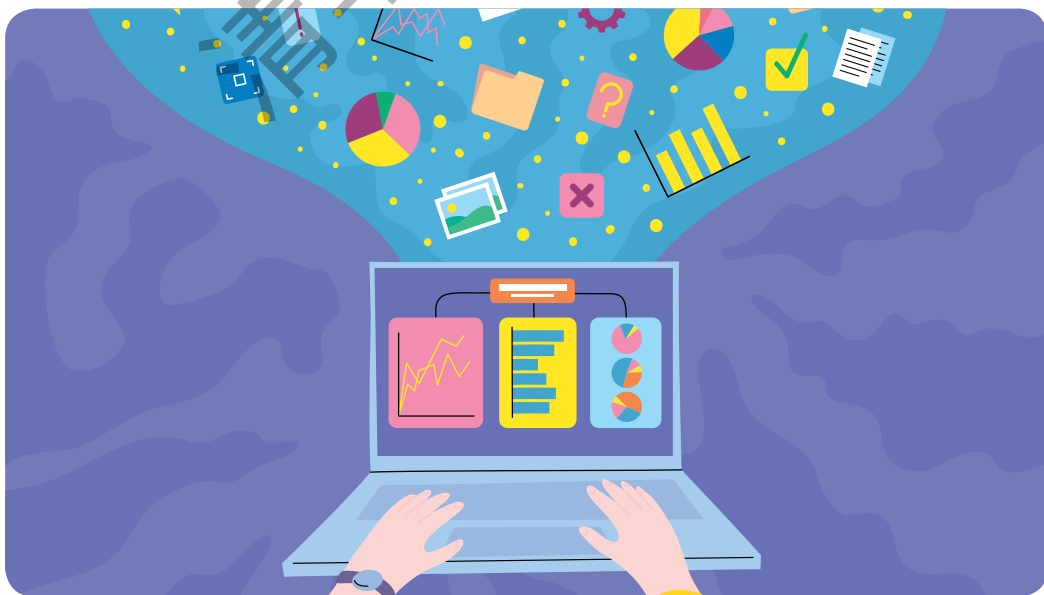
初识算法

📖 本单元你将学习：

- 什么是算法
- 计算机如何应用算法解决问题
- 生活中常见的算法
- 程序如何执行指令、验证算法

计算机最初从处理数字信息开始，再经过编码使得所有人类的信息都可以进行“计算”，成为传递和处理信息最核心的工具。那为什么计算机能够达到人脑处理信息的能力呢？它是做到像人一样思考的呢？这是人类通过“算法”告诉它的。

在本单元的学习中，我们将初步认识算法，理解算法对于计算机工作的意义，并探索生活中的算法。通过学习，我们将能够借助算法的帮助解决实际问题，并尝试编写简单的程序。



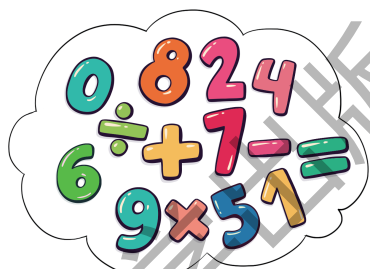
学习热身

双人计算大比拼（100 以内的整数四则运算）。

规则：

- 一方出题，另一方分步骤描述计算过程。
- 若回答正确，互换角色，继续出题作答。
- 一方答错，游戏结束，出题方获胜。

计算
算法



你知道吗？

算法就是我们解决问题的操作步骤。算得又快又准确的同学，一定是发现了更为简便的计算方法，能够省去更多重复的操作步骤，也就是“算法”更优。



第1课

走进算法



本课中你将学习：

- 算法是如何被提出的
- 算法给生活带来了什么变化
- 人工智能中的算法

班主任提出了布置班级环境的任务，身为班长的小清看着杂乱的教室、空空的墙壁，不知从何下手。小华来帮忙，她先环顾四周，表示第一步肯定是收拾教室卫生，然后将教室的三面墙壁按区域分主题进行设计。随后，他们根据班委们各自擅长的领域分工合作，很快就有序地完成了任务。

小青非常佩服小华，小华却说其实只是因为自己的“想法”很清晰、有步骤，再按照计划让大家分工完成各自的工作而已。这在计算机科学里就是“算法”。

同学们，如果你是小清，你能提出比小华更高效、优质的算法，去解决你们班级教室的开学环境布置的问题吗？

一、算法的诞生

“算法”一词最早源于数学中的计算，原本的含义是“数学的运算法则”。公元前1世纪成书的《周髀算经》，是中国最古老的天文学和数学著作（图1.1.1）。其中介绍的“勾股定理”就被称为一种“算法”。公元1世纪左右成书的《九章算术》（图1.1.2），其中提到的四则运算，也被书中表述为多种“算法”。



周髀算经

中国最早的数学及其“测天量地”之用

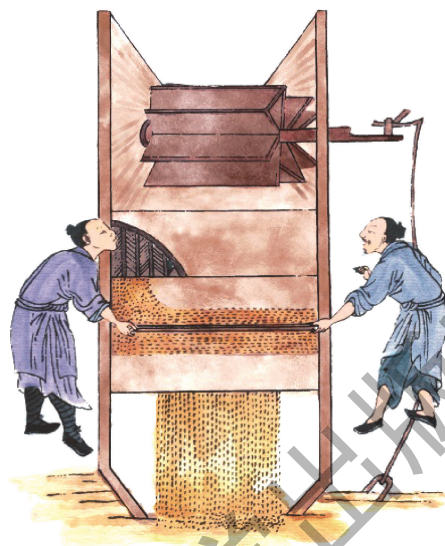


图 1.1.1 《周髀算经》

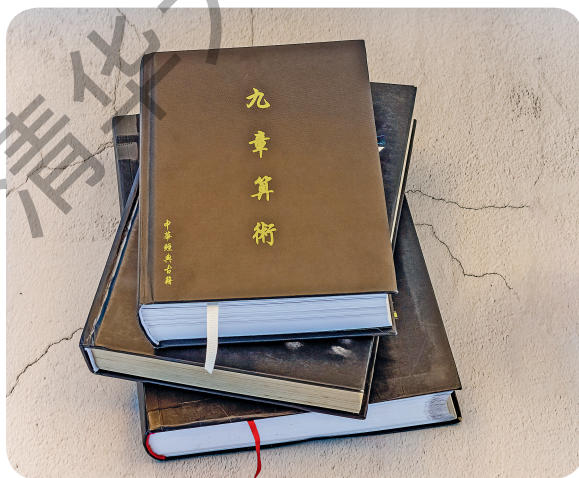


图 1.1.2 《九章算术》

随着数学的不断发展和人们计算能力的不断提高，越来越多的复杂计算方法在四则运算的基础上诞生，人们开始思考，如果能有机器的帮助我们完成运算，那是不是我们就能节省更多的时间去解决更关键的问题了呢？

阅读

20 世纪的英国科学家艾伦·麦席森·图灵（图 1.1.3）提出了著名的图灵测试，他被誉为“计算机科学之父”。他用一种抽象数学模型设计了一台假想的计算机，被称为“图灵机”（图 1.1.4）。图灵机通过一组有限的规则用机器来模拟人们用纸笔进行数学运算的过程。这使得数学中基本的数的计算可以交由机器处理，而人类则通过发明更精妙、更高效的计算方法，命令机器替我们完成工作。所以，计算机科学家们将算法看作是计算机执行计算任务，也就是处理信息的过程。

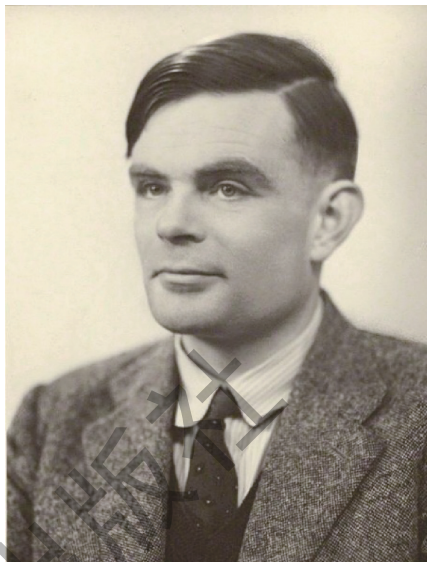


图 1.1.3 艾伦·麦席森·图灵

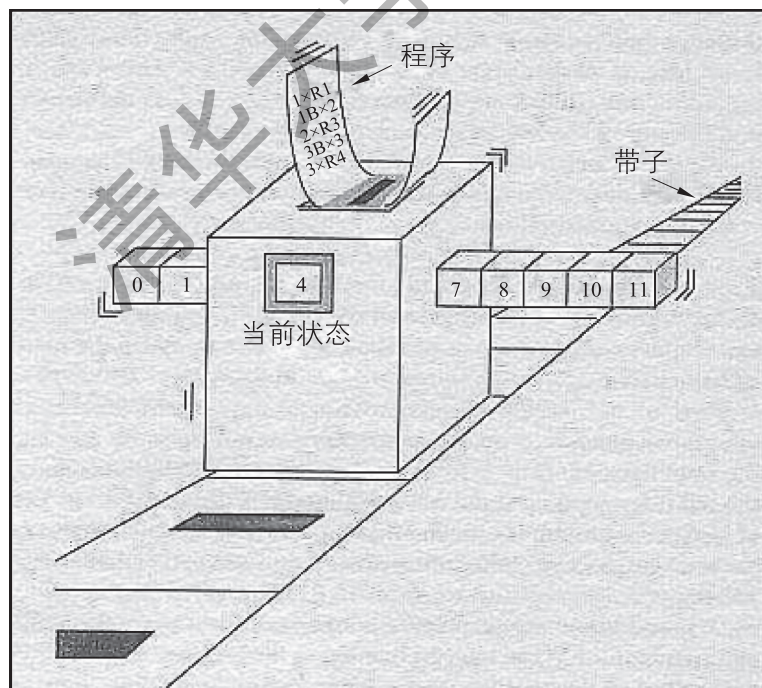


图 1.1.4 图灵机模型



随着图灵机模型的提出，用机器替代人脑进行运算成为人类的奋斗目标，计算机科学应运而生，计算机科学家们对于算法进行了重新定义。

总的来说，算法指的就是计算的方法，计算机通过程序实现，数学通过公式推导实现。

二、生活中的算法

算法作为一种思维方法，它的运用早已潜移默化地渗透到我们的日常生活中。比如，给物品分类、按顺序排队、玩各种策略游戏或者制订各类调研计划等，通常也是依据某种方法或策略，分步骤完成任务。

同学们熟悉的猜数字游戏，是在猜数字和判定大小的过程中，不断缩小可能的范围，最终猜中谜底中的数字，这就是一种比较再判定的算法。

给物品分类也包含很多种不同的算法，比如按照物品的作用分、按照物品的大小分、按照物品的颜色分、按照使用的频率分等。

针对同一道数学题目有不同种的求解方法，也是不同的算法。

探索

整理班级教室，将现在教室里的物品利用不同的分类方法进行整理，分小组提出方案，比一比谁的算法最合理、有序。

随着越来越多高级复杂的算法的诞生，提升了计算机能够处理更多复杂信息的能力，使得计算机能够帮助人类完成更多的工作，也更接近于人类的思维水平，这都是计算机科学家们在不断提升算法水平的功劳。

三、人工智能里的算法

算法是计算思维的核心要素之一，也是人工智能得以普遍应用的三大支柱（数据、算法和算力）之一。人工智能看似高深莫测，然而再高深的技术，也是从基础原理开始的。人工智能领域中的有些基础算法，原理浅显，很早就生活中被人们用来解决生活中的实际问题了。

决策树，是人工智能中的基础算法之一，其实也是我们每天遇到问题时，自己进行判定、做出决定的思维过程。举个简单的例子，学校要选一

个地点组织全校秋游，要从距离、花费、教育意义、场地面积、场地资质等方面共同考虑得出结论，最终决定是否可以选择该场地，会得到图 1.1.5 所示的决策树。

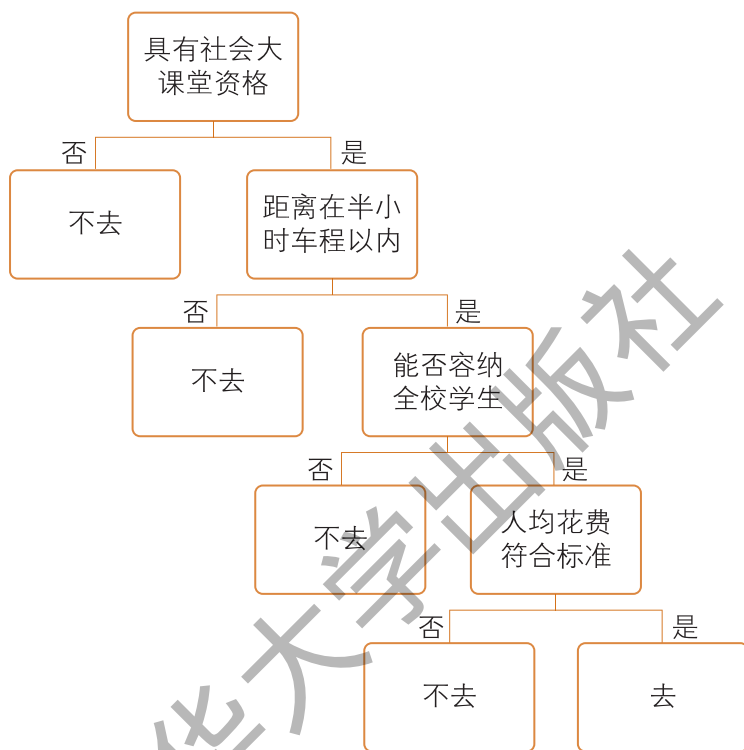


图 1.1.5 决策树

探索

同学们，你们认为优秀文明班级必须具备哪些条件呢？利用决策树，设计一套优秀文明班级的评价方案，同学们和老师门一起评判，这套方案是否公正、公允、可行，并在年级试运行。

人工神经网络，是模仿人脑的神经元工作原理建立的算法，也是人工智能得名的由来。就像人的大脑一样，人工智能不断处理着接收到的信息，进行处理后输出，寻求人类的判定和反馈，最终得到正确的处理结果，完成自我学习。我们生活中常见的图像识别，就是神经网络中的一个著名应用。



阅读

神经元是动物大脑中的神经元细胞（图 1.1.6），是动物神经系统最基本的结构和功能单位。生物科学家们发现了动物以及人脑的生物结构之后，破译了神经系统的工作原理。1943 年，美国科学家麦卡洛克（W. McCulloch）和皮茨（W. Pitts）提出了利用神经网络对信息进行处理数学模型，神经网络算法由此诞生。

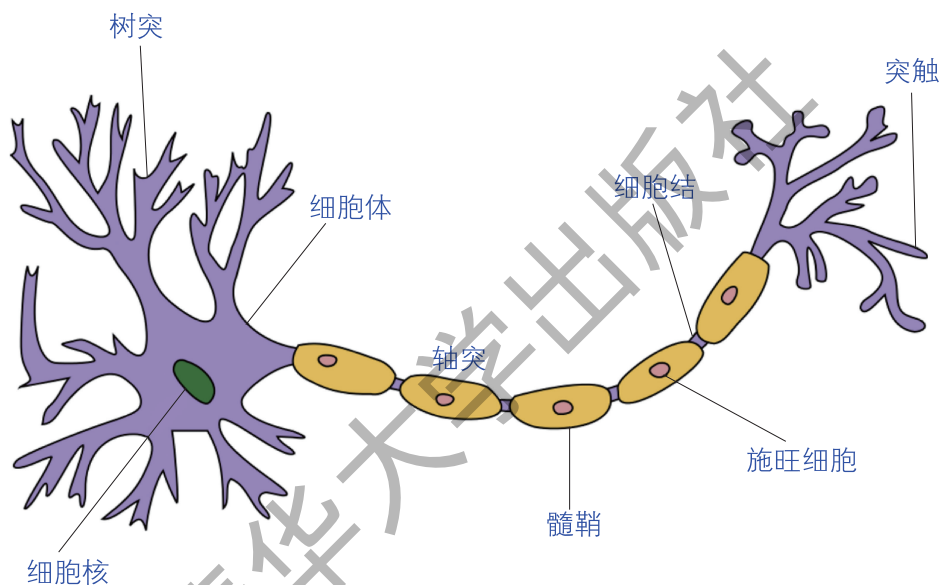
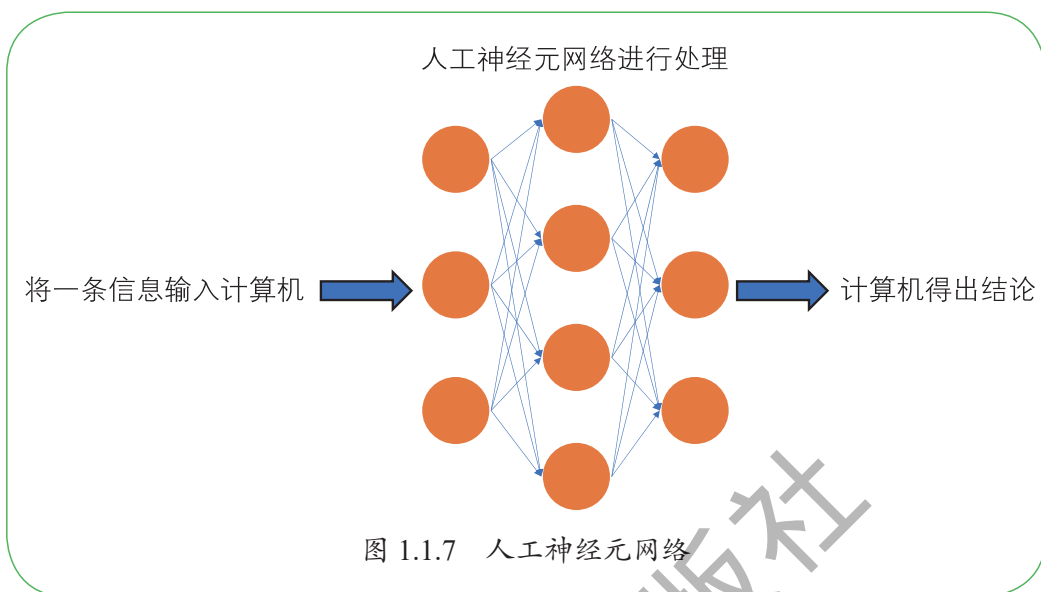


图 1.1.6 神经元细胞

应用在人工智能领域的人工神经网络算法（图 1.1.7），是 20 世纪 80 年代以来人工智能领域兴起的研究热点。它从信息处理角度对人脑神经网络进行抽象，建立成用符号和流程表示的模型，可以按不同的连接方式组成不同的网络，从而具备处理各种不同类型问题的能力。它就像是人工智能的大脑，是计算机的神经中枢。



人工智能技术其实就是利用仿生学原理，计算机科学家们参照人的思维模式，解析出人类解决问题的过程，设计出相应的算法，从而实现让计算机模仿人类的思维方式，像人类一样处理信息、解决问题。

挑战

和父母长辈聊一聊，随着算法的不断优化，同一个 App 或者其他智能技术经过这些年的更新和升级，在使用体验上有什么变化？感受算法的进步给信息科技支持下的生活带来的质量提升。

第2课

算法的工作过程和特点



本课中你将学习：

- 了解算法的工作过程
- 掌握算法的特点
- 能使用算法解决实际问题

随着科技的发展，计算机的功能越来越强大，计算机在我们的生活中发挥的作用也日益增强。从家中的智能扫地机器人，到出行游玩的好伙伴智能导航系统，计算机似乎变得有了“智慧”，能够“思考”。而这拥有“十八般武艺”的计算机的“智慧”都源于它背后的算法，只有在算法指挥下的计算机才能成为我们的好帮手。那么，算法是如何快速解决问题的呢？

一、算法的工作过程

算法的出现最初就是为了解决难易程度不同的问题，确定解决一类问题或者完成某件事情的步骤。算法还清晰地描述了在按照步骤执行的过程中需要遵循的规则。只要按照算法的步骤与规则一步一步操作，就能够使问题得到解决。

对于计算机而言，算法的使用亦是如此。算法就像一份精妙的设计图纸，规划了计算机每一步应该做什么、要如何做。只要依照算法编写出正确的计算机程序，再由计算机执行程序，便能得到预期的结果。

算法的工作过程可以简单概括如下（图 1.2.1）。

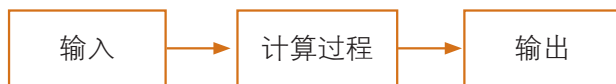


图 1.2.1 算法的工作方式

当我们准备去比较远的地方旅游时，做好规划后的第一件事通常是订购火车票或飞机票，智能购票系统可以快速为我们提供有效的购票信息。在我们输入与出行相关的各种信息（如起点、终点、乘车时间等）后，购票系统会经过一系列计算，为我们推荐满足条件的所有车次（图 1.2.2）。



图 1.2.2 购票系统

购票系统之所以可以为我们提供可购买车票信息，是因为它有一套完善的算法（图 1.2.3），只需要按照算法一步一步执行，计算机就可以将最终的车票信息提供给我们。



图 1.2.3 购票系统的算法

其中，购票信息是我们给计算机输入的内容，如果我们不提供起点、终点、乘车时间等信息的话，计算机就不知道要向我们提供哪些内容。



因此，在使用算法时，首先要有部分确定的内容作为解决问题的已知条件，这部分已知条件其实就是算法中的输入部分。算法的输入其实就是问题的初始条件，不同的问题会有不同的输入，没有输入将不知道从何开始。

最后一步“提供详细的车票信息”就是计算机输出的内容，也是最终实现我们向计算机提出的“我们可以乘坐哪些车次出行？”这一问题的结果。不论中间的步骤多么复杂，最终我们都会得到一个结果。如果没有输出，那么这一算法将毫无意义。

因此，一个算法只有包含确定的输入和输出后，才是有价值的算法，才能发挥算法的作用（图 1.2.4）。

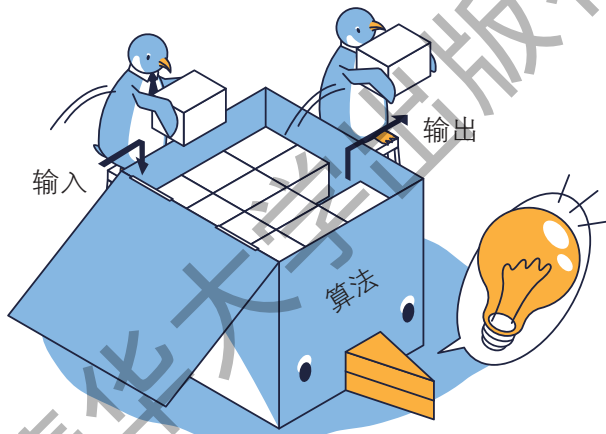


图 1.2.4 算法的输入与输出

在购票系统的算法中，中间的两步就是计算部分。当算法为计算机描述了一条准确的、清晰的、可以执行的计算方法后，计算机就能够根据输入，经过计算，实现我们所期待的结果的输出。

这就是精妙的算法，是它让计算机变得“智能”又“博学”，成为了我们的好帮手、好朋友。

探索

请你使用算法的工作方式，按照“输入—计算—输出”描述一下购物网站是如何实现为我们展示搜索内容的。

二、算法的特点

算法的出现是为了帮助人们精准地解决问题，那么，算法要能够被每个人使用，需要满足一些要求，这些要求也就是有价值的算法所共有的特点。我们以乘坐地铁（图 1.2.5）为例，乘坐地铁的算法可以描述为：提供购票信息——购买地铁票——进入地铁站——找到正确的车辆——找到正确的乘车方向——等待地铁到站并开门——上车——站立或找到座位——到达目的地后下车——出站。



图 1.2.5 乘坐地铁

1. 可执行性

在乘坐地铁的算法中，不论是购买地铁票、等待地铁，还是上下车，其中的每一步都是我们可以完成的，不论是谁，拿到这个算法都能够顺利乘坐地铁，这就是我们能够使用的算法。

如果我们将这一算法中“找到正确的车辆”变为“找到不存在的车辆”，这一步我们就无法执行，这一算法自然就无法进行下去。

算法的每一步都应该是可以做的，如果使用现有知识不能做就不是算法，这就是算法的可执行性，它是保证算法能够顺利实现的基础。

2. 有穷性

在乘坐地铁的算法中，我们最终的结果是到达目的地并出站，中间不论经过多么复杂的乘坐地铁步骤，即使有换乘，我们最终也能够到达目的地。在有限的时间里让我们解决了乘坐地铁去目的地的问题，这就是一个有意义的算法。



如果这一算法让我们反复乘坐地铁，但没有到达目的地和出站的步骤，我们将永远被困在“乘坐地铁”这一步骤中无法结束，也就永远无法到达目的地。没有结束，就没有解决我们的问题，这个算法就是无效的。

因此，算法一定要是可以结束的，如果不能结束就没有结果，就没有意义，这就是算法的有穷性。

3. 确定性

我们再次观察这一算法，“找到正确的车辆”“找到正确的乘车方向”……这些步骤在执行的时候给了我们非常明确的要求，这就使我们可以顺利到达目的地。如果，我们将它的描述改为“找到一辆车”和“随便选择一个方向”，在执行的过程中就会因为没有办法确定到底如何坐车而产生混乱。

因此，我们要求一个算法的每一步描述都必须是精准的、确定的、没有分歧的，它只能有一种明确的操作方法，这就是算法的确定性。

只要做到上面三点，再结合算法的输入、计算、输出三大步骤，该算法就是一个有价值的算法（图 1.2.6）。

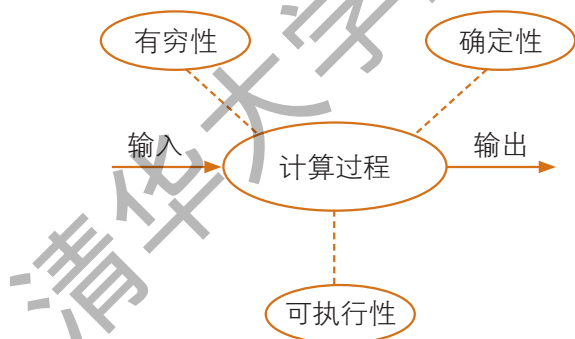


图 1.2.6 算法的特点

根据算法的工作过程和特征，一个有意义的算法，是按照以下方式生成的。

- (1) 有一个或几个输入。
- (2) 需要明确一共需要执行几步，每一步都要有准确且唯一的指令，并确认这些指令是可以被完成的（这一部分就是算法的计算过程）。
- (3) 算法要能够结束，并在结束的时候输出结果。

生活中，很多问题的解决过程都可以用算法思想进行整理，比如日常学习计划、家庭劳动计划、数学解题方法等，使用算法思想思考和解决问

题可以让一切变得井井有条。

挑战

某同学制订了一份新学期的英语学习计划，阅读 5 本书，大概 500 页，摘抄英文句子 100 条，每周写至少 3 篇英文日记。

将“新学期的英语学习计划”制作为算法，根据算法的特征，请说一说，这份计划中是否存在问题？问题在哪里？

空白区域，用于思考或记录。

请你使用算法生成步骤，制订一份属于你的本月学习计划（任选一个学科），并明确指出计划中的输入、计算过程、输出分别是什么？

输入

空白框，用于填写输入内容。

计算过程

空白框，用于填写计算过程。

输出

空白框，用于填写输出内容。

第3课

算法的应用



本课中你将学习：

→ 生活中计算机算法的常见应用

在数字时代，无论是使用智能手机还是计算机，无论是在网上购物、线上学习，还是进行智能出行、网络社交，我们每一分每一秒都在使用算法，甚至可以说算法已成为生活的一部分。那么我们身边究竟有哪些算法呢？

一、在线地图中的算法

在现代生活中，算法的应用大多基于计算机、智能手机等设备。比如，小华的家在北京，想要去往中国美术馆进行参观，在查询路线的时候，在导航软件中输入起点和终点信息后，地图导航经过计算，就会推荐一条能够最快到达中国美术馆的路线（图 1.3.1）。如表 1.3.1 所示，根据算法的步骤，一步步执行，最终找到用时最短的路线，并将结果展示在小华面前（图 1.3.2）。

表 1.3.1 小华家到中国美术馆路线表

路线	用时	距离	途 经
方案①	15 分钟	5.9 公里	19 个红绿灯，地安门西大街、地安门东大街
方案②	16 分钟	5.9 公里	17 个红绿灯，地安门西大街、北河沿大街
方案③	17 分钟	6.1 公里	18 个红绿灯，赵登禹路、地安门西大街

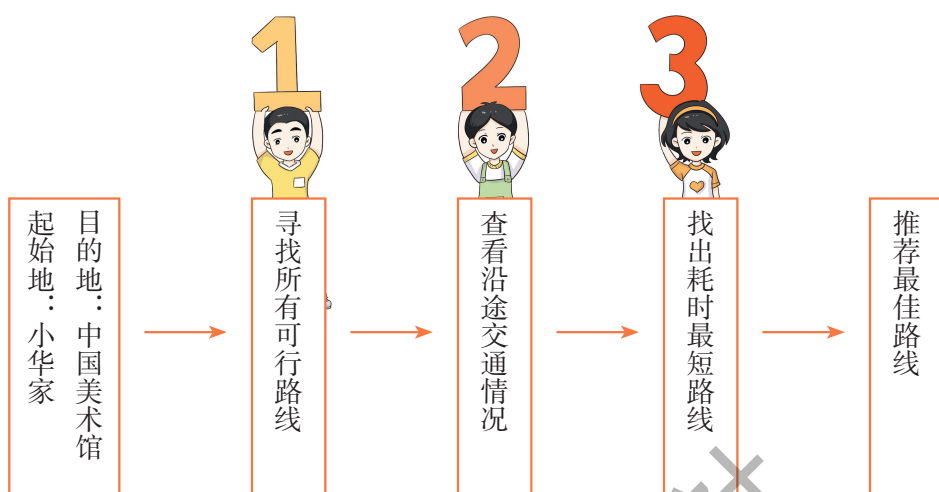


图 1.3.1 路线算法推荐流程



图 1.3.2 最佳导航路线方案

探索

利用在线地图，搜索从你家到学校的路线，从地图推荐的方案中分析当前推荐的最优路线是计算机基于什么条件推荐得出的。再尝试修改你的出行偏好（比如：出行时间、红绿灯数量、是否经过高速等），看看如果条件发生改变，最优路线方案推荐会不会也随之发生改变。

二、图像识别中的算法

人脸识别技术现在已经应用到了生活中的方方面面，比如：火车站人



脸识别进站、面容支付、小区面部识别门禁系统等。智能手机上的图像识别用途更广泛，不仅能识别人脸，还能拍照识物。公安机关借助精密图像处理技术复原案发现场的蛛丝马迹，还原事实真相，找到重要物证以及抓捕犯人。

“看图识物”对于人脑来说似乎是一件极其简单的事（图 1.3.3），而对于图像识别算法来说，却也经过了一系列复杂的辨别流程（图 1.3.4）。

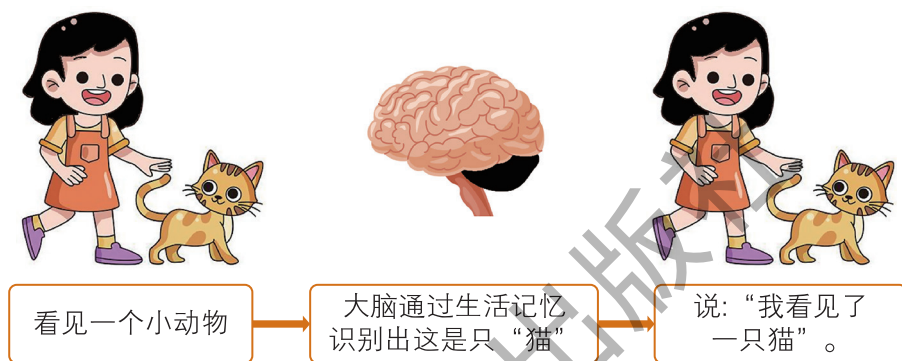


图 1.3.3 人脑判断物体的过程

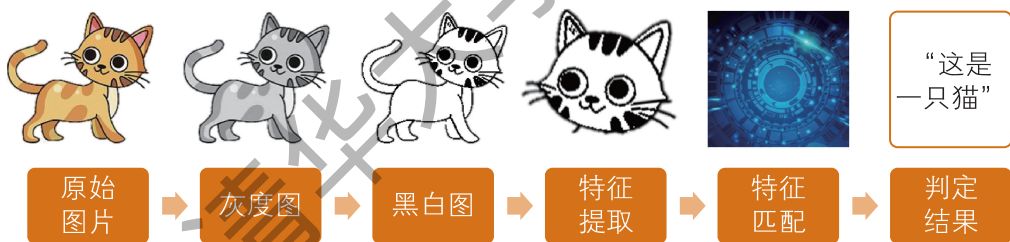


图 1.3.4 计算机图像识别算法

探索

和同桌一起玩一玩“读心游戏”，同学 A 心中想一件物品，同学 B 猜一猜物品是什么。同学 B 通过提问的方式缩小物品可能的范围，同学 A 只能回答“是”或者“不是”。看看你们能不能在 20 个问题之内得出答案。

三、智能产品中的算法

我们身边的许多硬件和软件产品，譬如智能手环、扫地机器人、视频

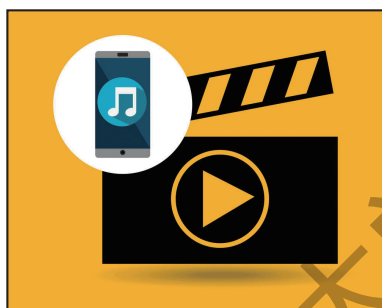
编辑软件、线上音乐平台（图 1.3.5）等，背后都有不同的算法执行着。这些智能产品的算法都能够获取数据，准确、快速地分析出有用的信息，实现不同功能，帮助人们解决问题。



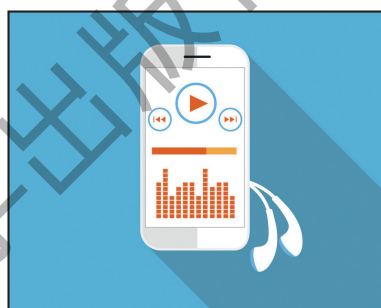
智能手环



扫地机器人



视频编辑软件



线上音乐平台

图 1.3.5 应用计算机算法的智能产品

例如，一辆无人驾驶汽车，如何从 A 点安全地行驶到 B 点？行驶过程中，计算机需要使用各种传感器、卫星定位信息等来感知周围的行车环境、交通状况等，计算机程序根据所获得的信息，规划路线并且判断该如何驾驶，最终发出正确的指令控制汽车行进（图 1.3.6）。

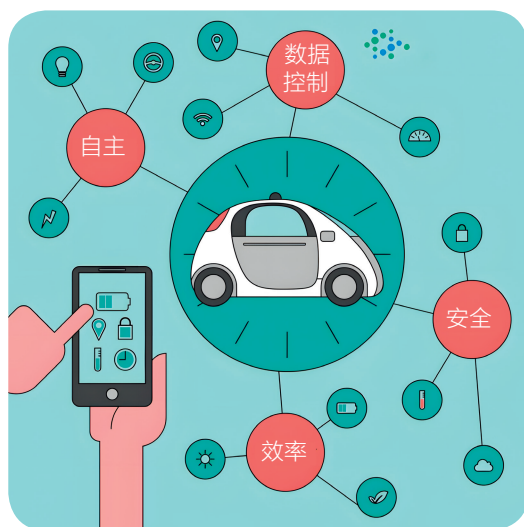


图 1.3.6 无人驾驶汽车算法



阅读

智能手环的工作主要依赖于其内置的传感器。传感器通过感知人运动的动作得到一些基础数据，再结合用户个人身体的基本信息，根据一些特殊算法，得到针对个人的个性检测数据，如运动步数、距离和消耗的热量等。由于每个人身体体征不同，运动将产生不同数据，因而用户在使用智能手环进行监测前，需要在手机软件中录入自己的性别、年龄、身高、体重等信息，信息同步到智能手环中，通过传感器感知运动动作，经过特定算法最终实现运动监测的功能。



挑战

智能电子产品在复杂算法的支持下为我们的学习和生活带来了很大的帮助。请同学们分组讨论，仿照示例，根据生活经验、借助网络搜索，将产品信息填写在表 1.3.2 中，并和全班同学分享调研成果，说一说这些产品的工作流程是怎样的。

表 1.3.2 智能产品调研表

产品	输入	输出	使用方法
智能手环	运动数据 身体数据	运动记录 身体状态 运动建议 健康提示	佩戴在手腕上，手环背面有传感器，可自动记录运动、身体等方面数据

第4课

活动：计算大比拼



本课中你将体验：

- 设计一个可以实现四则运算的程序
- 如何使用图形化编程软件编写程序
- 怎样设计算法来帮助编写程序
- 制作并保存好程序，并分享给小伙伴

我们一起利用计算机来实现本单元学习热身中的计算大比拼游戏吧！看看计算机是不是可以在我们的算法控制下，实现任何数的四则运算？借助图形化编程工具，通过程序实现四则运算，设计一个简易的计算程序（图 1.4.1）。



图 1.4.1 四则运算



想一想

我们先从实现加法开始，想一想，实现两个数的相加，算法步骤是什么？请写在下面的横线上。



算法步骤：

- ① _____
- ② _____
- ③ _____
- ④ _____

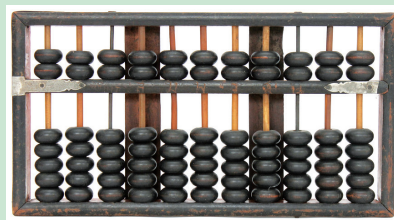


读一读

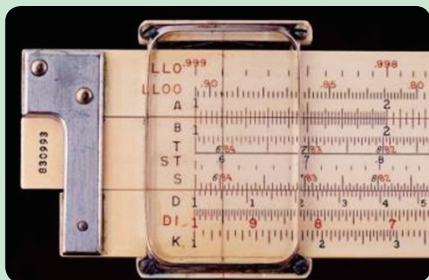
算法的出现最初是为了解决各种各样的计算问题，但是随着算法变得越来越复杂，单纯使用口算、心算或使用纸笔计算开始变得较为困难。于是，对于一些复杂烦琐的计算问题，我们就需要借助一些工具来解决，这种工具称为计算工具。从数学产生之日起，人类就在不断发明能方便和快速计算的工具（图 1.4.2）。



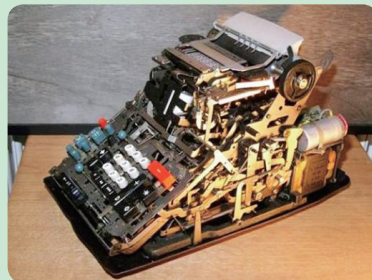
算筹



算盘



计算尺



机械计算器

图 1.4.2 计算工具



你能用图形化编程工具进行两个数的加法计算吗？可以按照程序算法的输入、处理、输出流程来制作作品。

1. 程序的输入

①在图形化编程工具中获取输入的数据，可以使用“外观”模块中的“询问【 】并等待”积木（图 1.4.3）。

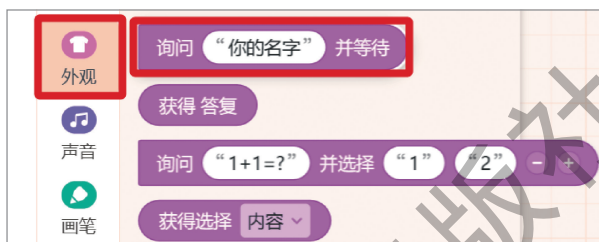


图 1.4.3 “询问【 】并等待”积木

②在“变量”模块中，单击 **+ 变量**，创建变量“加数 a”和“加数 b”（图 1.4.4）。



图 1.4.4 创建变量



③设置变量“加数a”和“加数b”的值为询问后的“获得答复”(图 1.4.5)。



图 1.4.5 获得答复

2. 程序的处理

成功获取加数a和加数b的数据后,运用“运算”模块中“四则运算积木块”,按照加法运算法则(图 1.4.6)进行运算。

四则运算积木块



图 1.4.6 加法计算两个数的和

3. 程序的输出

要输出数据，可以使用“对话【】”或“新建对话框【】”积木（图 1.4.7）。



图 1.4.7 输出两个数的和



议一议

请让小伙伴尝试设置不同的“加数 a”“加数 b”的值。将值记录在表 1.4.1 中，算一算，验证编写的程序是否准确。

表 1.4.1 记录表

加数 a	加数 b	和



做一做

改编程序，分别实现 $+$ 、 $-$ 、 \times 、 \div 的计算（图 1.4.8）。



图 1.4.8 四则运算程序



评一评

设计完程序后，注意保存并分享给其他的小伙伴，展示自己的作品吧！

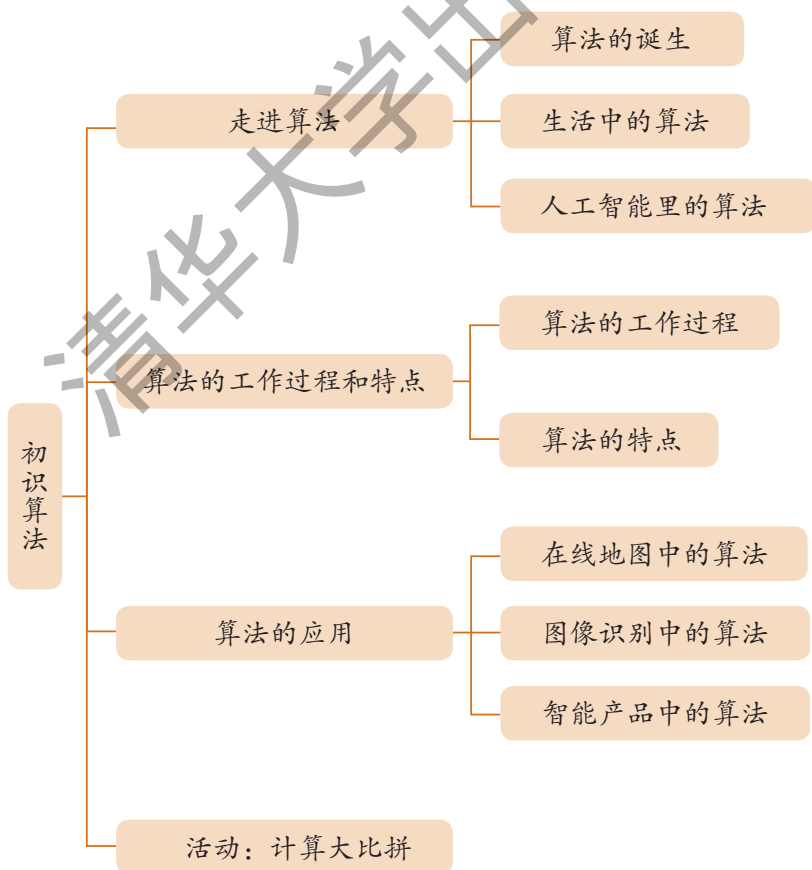
内容	遇到的问题	解决方法	感悟收获	总体评价
想一想				☆☆☆☆☆
读一读				☆☆☆☆☆
做一做①				☆☆☆☆☆
议一议				☆☆☆☆☆
做一做②				☆☆☆☆☆

小结与评价

本单元你学习了：

- ◆ 什么是算法
- ◆ 用算法描述事物
- ◆ 生活中的算法
- ◆ 编写四则运算的程序

知识梳理





学习检测

1. 你知道什么是算法吗?
2. 你能举出生活中算法应用的例子吗?
3. 你能说出算法是如何影响计算机工作的吗?
4. 算法的主要特征有什么?
5. 请分步骤描述从早上起床到上学的过程。



反思评价

在本单元学习过程中,肯定少不了与他人互动交流、参与作品制作等活动。现在请就此进行总结与反思,以更好地促进自身成长。

1. 从同伴那里学到过什么?
2. 向同伴分享过哪些观点?
3. 所运用的工具和方法是否得当?
4. 设计的作品是否可以推广至其他问题的解决?

2

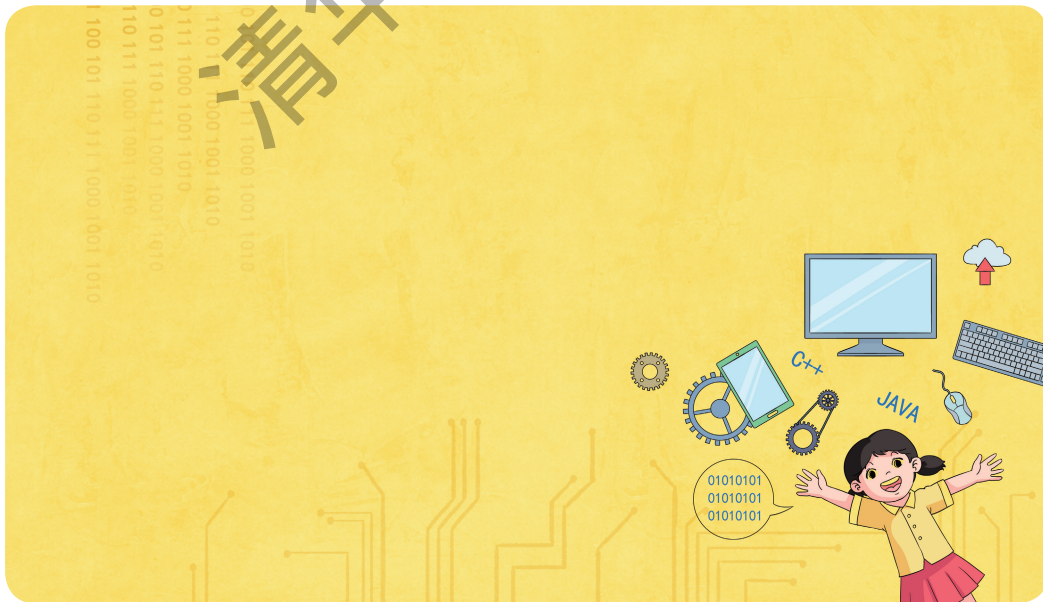
描述算法

📖 本单元你将学习：

- 如何用自然语言、流程图、编程语言描述算法
- 如何用程序实现基本的算法

要用算法解决问题，需要先描述算法。描述的目的是方便认识和理解，如在游览景区时，会看到各式各样的导览牌，有用文字描述的，有用路线描述的，也有用图示描述的，都是为了方便游客观光游览。想要描述算法，首先，要明确算法的目标是要分析解决什么问题。其次，根据目标选择描述方式。最后，用相应的方式描述算法。

描述算法的方式有很多种。在本单元的学习中，我们将重点学习用自然语言、流程图和编程语言来描述算法。



学习热身

思考并讨论，计算机能直接理解用文字表述的算法吗？根据所学知识，尝试说一说，计算机能理解哪些指令？

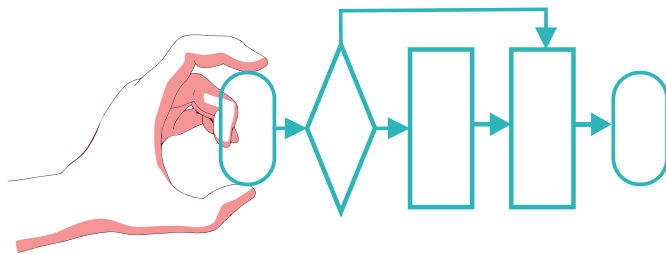


自然语言
流程图
编程语言



你知道吗？

算法描述就是通过描述解释说明算法的形式、原理等，以便人们理解、交流算法。算法的描述很关键，可用自然语言、流程图、编程语言等进行描述。



第1课

用自然语言描述算法



本课中你将学习：

- 什么是自然语言
- 如何用自然语言描述事物以及问题的解决步骤
- 如何用自然语言描述算法问题

自然语言包括我们在生活中交流沟通所说的话、写文章所用的语句，可以用来表达观点、描述事物。同学们，你们知道吗？用自然语言还可以描述算法呢！

一、用自然语言描述事物

用自然语言描述事物，我们可以使用常用的词汇、句子等来描述事物的外观、功能、用途等信息。例如，如图 2.1.1 所示的一辆汽车，使用自然语言对其进行描述，一般可以从它的外观、品牌、颜色、型号、用途等方面进行描述，可以描述为“这是一辆黄绿相间的红旗牌出租车”。当然，我们还可以描述汽车的某些行为，如加速、刹车、转弯等，“这是一辆停在路边的出租车”。



图 2.1.1 出租车



二、用自然语言描述问题解决的步骤

在学校或公共场所的洗手池处，通常会有洗手的步骤描述（图 2.1.2）。这其实就是在用自然语言描述具体问题的解决步骤。



图 2.1.2 七步洗手法的步骤

七步洗手法的步骤包括：

- (1) 洗手掌：双手掌心对掌心，手指并拢相互揉搓。
- (2) 洗背侧指缝：掌心对手背，手指交错相互揉搓，然后两手交替。
- (3) 洗掌侧指缝：双手掌心对掌心，手指交叉相互揉搓。
- (4) 洗指背：双手互握，揉搓指背，然后两手交替。
- (5) 洗拇指：拇指在另一手掌中转动揉搓，然后两手交替。
- (6) 洗指尖：指尖并拢在另一手掌中搓擦，然后两手交替。
- (7) 洗手腕：双手交替清洗手腕。

有了图示，对于同学们而言，读懂这些步骤将会变得更加容易。这种图文搭配的描述方式也是自然语言的描述方式。

用自然语言描述问题的解决步骤需要注意：

第一，要简要明确地描述问题是什么。

第二，描述具体步骤时，语言要明确和精练，避免使用模糊不清的表达方式。

阅读

采用不同的步骤解决问题的结果可能是一样的，但效率却可能有很大不同。如何规划步骤才能让做事更有效率，这是一个值得我们反复思考的问题。

比如日常生活中，为了节省时间，我们会先煮米饭，然后在煮米饭的同时去洗菜、切菜、炒菜，这样等到菜做好，米饭也煮熟了。通过有序地安排步骤，就能达到提升效率的目的。

探索

规划一下从早上起床到入校的步骤，争取做到最高效，将规划的步骤用自然语言描述出来，写在下面的空白处，并讲给老师、同学们听一听。



三、用自然语言描述算法

使用自然语言描述算法，可以通过文字或数字表达式来阐述解决问题的步骤和过程。

1. 用自然语言描述算法的“格式”

在日常的生活和学习中，写日记、写信、写作文都有固定的格式，比如写信的时候要有“称呼”“问候语”“正文”“署名和日期”等。用自然语言描述算法也有一定的“格式”。

以下是一个用自然语言描述 $s = a + b$ 求和算法的例子。

算法名称：求两个整数的和。

输入：整数 a 和 b ；

输出：它们的和 s 。

计算步骤：

- (1) 读取输入的两个整数 a 和 b 。
- (2) 将 a 和 b 相加，将结果赋值给 s 。
- (3) 返回 s ，输出结果。

目的：计算给定的两个整数 a 和 b 的和。

结果：将 a 和 b 相加，输出它们的和 s 。

一般算法的描述要包括“算法名称”“输入”“输出”“计算步骤”“目的”“结果”“适用性”等方面。一般来说，算法名称、输入、输出、计算步骤、目的、结果为必要元素，其余的如“适用性”等则需根据实际情况进行描述。

探索

用自然语言描述算法的形式描述“求长方形的周长”的算法。

2. 自然语言描述算法的优势和劣势

用自然语言描述算法通俗易懂，适合描述一些简单的问题或算法，但是对于复杂算法来说，用自然语言描述可能会显得冗长且容易产生歧义，从而导致对算法的理解有误。在使用自然语言描述算法和事物时，要避免

产生歧义。例如，“她弟弟和我说的一模一样”这句话就可能有几种含义。

同时，一些算法需要基于特定的领域知识，如像人脸识别一样的各类图像处理算法（图 2.1.3）、语音识别类的音频处理算法等。这些算法的原理和过程可能需要专业领域的知识才能理解，使用自然语言描述会存在一定的困难。



图 2.1.3 人脸识别

探索

请查阅相关资料，写出用自然语言描述算法有什么优势和劣势，填写表 2.1.1。

表 2.1.1 用自然语言描述算法的优势和劣势

优势	
劣势	



挑战

校门口红绿灯的运行过程如下：当红绿灯开始运行时，绿灯亮，持续 30 秒；黄灯亮，持续 3 秒；红灯亮，持续 30 秒；然后重复执行这个过程。

如果将红绿灯的运行过程对应的算法叫作“红绿灯”算法，你能用明确的步骤将算法描述出来吗？填写在表 2.1.2 中。

表 2.1.2 “红绿灯”算法

步骤	描 述
1	
2	
3	
4	
5	
.....	

第2课

用流程图描述算法



本课中你将学习：

- 什么是流程图
- 如何用流程图描述身边的事物
- 如何用流程图描述算法

流程图是描述事物次序与逻辑的常用方法，通过流程图能够直观地看出完成某项工作的具体步骤。在算法描述中也是一样，流程图可以直观地将算法的逻辑很好地呈现出来。

一、流程与逻辑

生活中，我们做事情要分步骤，其中每个步骤都有着各自的顺序，这个顺序要符合一定的逻辑，最终形成一种“流程化”的模式。

下面我们以“哥哥做木马玩具”为例，来体会“流程化”的工作。

哥哥给妹妹做了一个木马玩具，以下是一些操作内容，如图 2.2.1 所示。

结合我们之前学习的自然语言描述，可以描述把“哥哥做木马玩具”的步骤描述为：先买一些颜料，找到一些木材，然后把木材雕刻成马的形状，再给木马涂上颜料，最后等到颜料晒干。

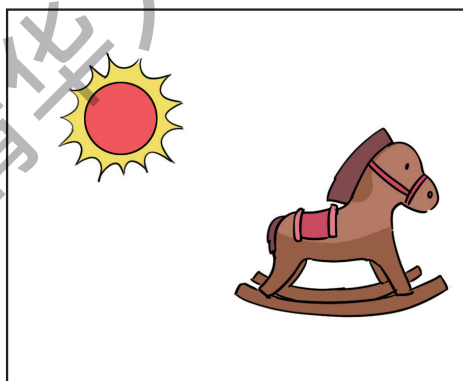
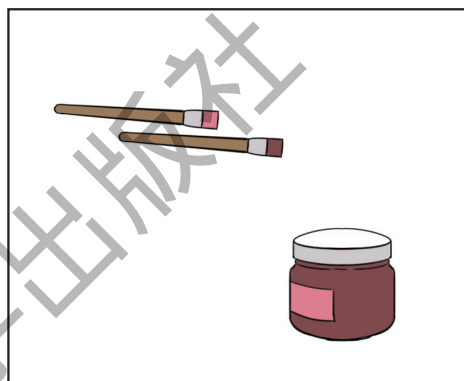
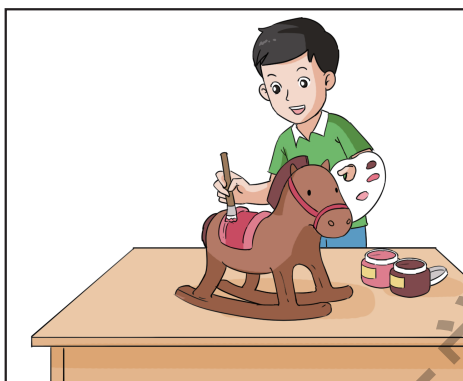
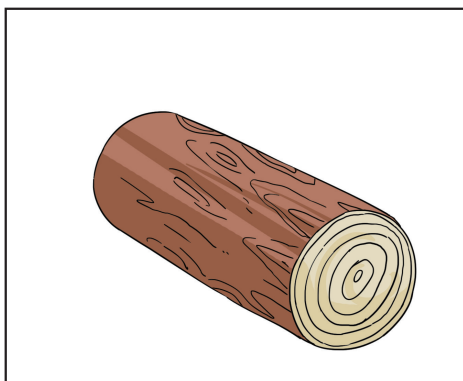


图 2.2.1 哥哥做木马玩具的操作内容

为了让整个过程更加直观，我们可以用图示的形式呈现出整个“木马玩具制作”的顺序，如图 2.2.2 所示。

其中，“买一些颜料”和“找到一些木材”的顺序是可以调换的，无论谁先谁后都是符合逻辑的。

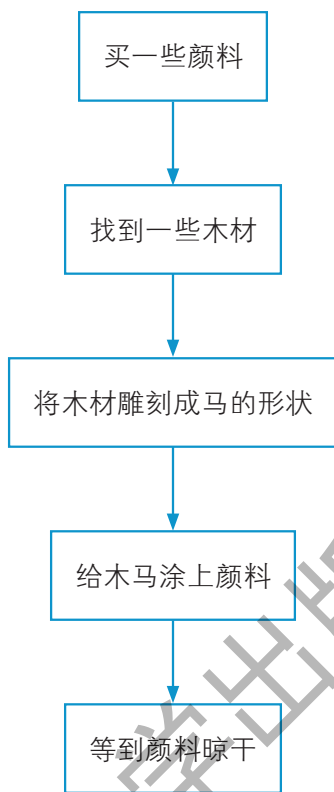


图 2.2.2 哥哥做木马玩具的流程图

以上用简单图示表示一定次序与逻辑的方式，就是流程图描述，相比自然语言描述而言，要更加清晰一些。

探索

用纸和笔，规划一下从早上从起床到入校的步骤与流程，用简单的图示呈现出来。



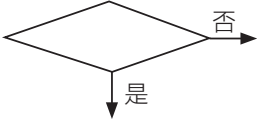


二、流程图

流程图是一种用于展示流程、步骤、决策和条件的图形化工具。它使用不同的符号和箭头来表示不同的操作和流程，可以帮助人们更好地理解复杂的流程，从而提高工作效率和准确性。

在具体的流程图绘制中，最为常用的流程图元素如表 2.2.1 所示。



表 2.2.1 常用流程图元素的样式

图形	名称
	起止框
	处理框
	判断框
	流程线
	输入 / 输出框

以下是几个简单的、不同形式的流程图。

例 1 给水池中注满水的流程图（见图 2.2.3）。

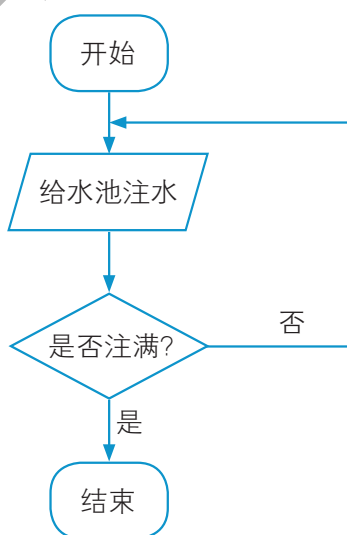


图 2.2.3 给水池注满水的流程图

流程图中从开始到给水池注水，是用流程线直接相连的。给水池注水后，判断是否注满，如果判断结果为“是”，则结束；否则，返回到“给水池注水”这一步骤，然后继续判断水池是否注满，循环往复。

例 2 计算正方形面积的流程图（见图 2.2.4）。

在计算正方形面积的流程中没有判断框，不过有“输入正方形的边长”的输入框和“输出正方形的面积”的输出框。

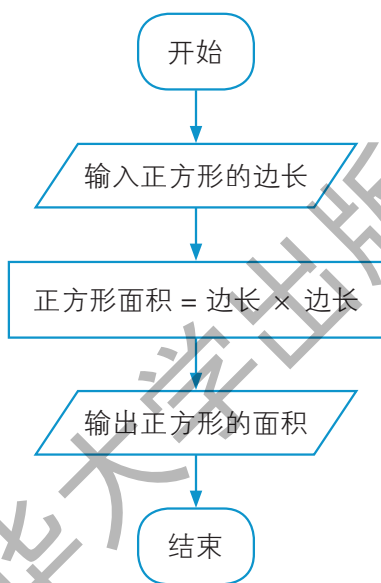


图 2.2.4 计算正方形面积的流程图

探索

如果要求三角形的面积，你会修改图 2.2.4 的哪些步骤？怎么修改？

三、用计算机软件绘制流程图

我们既可以用纸笔来绘制流程图，也可用一些计算机软件工具来辅助完成流程图的绘制。

在常用的办公软件中，就有相应的流程图绘制工具，以常用的文字处理软件为例，单击“插入”菜单后如图 2.2.5 所示。



插入菜单→在线流程图

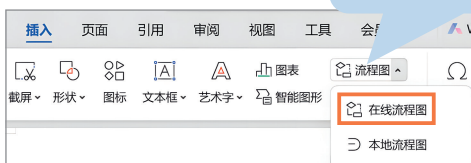


图 2.2.5 插入在线流程图

单击在线流程图，选择新建空白。进入绘图工具中，可以利用基础的流程图节点图形，帮助我们在计算机上绘制出标准、美观的流程图。

以给水池注满水的流程图为例，我们怎样在计算机上绘制流程图呢？

首先打开文字处理软件，在菜单栏中单击“插入”菜单项，在下拉菜单中选择“在线流程图”，进入流程图的绘制界面，新建一个空白页，开始绘制流程图，如图 2.2.6 所示。

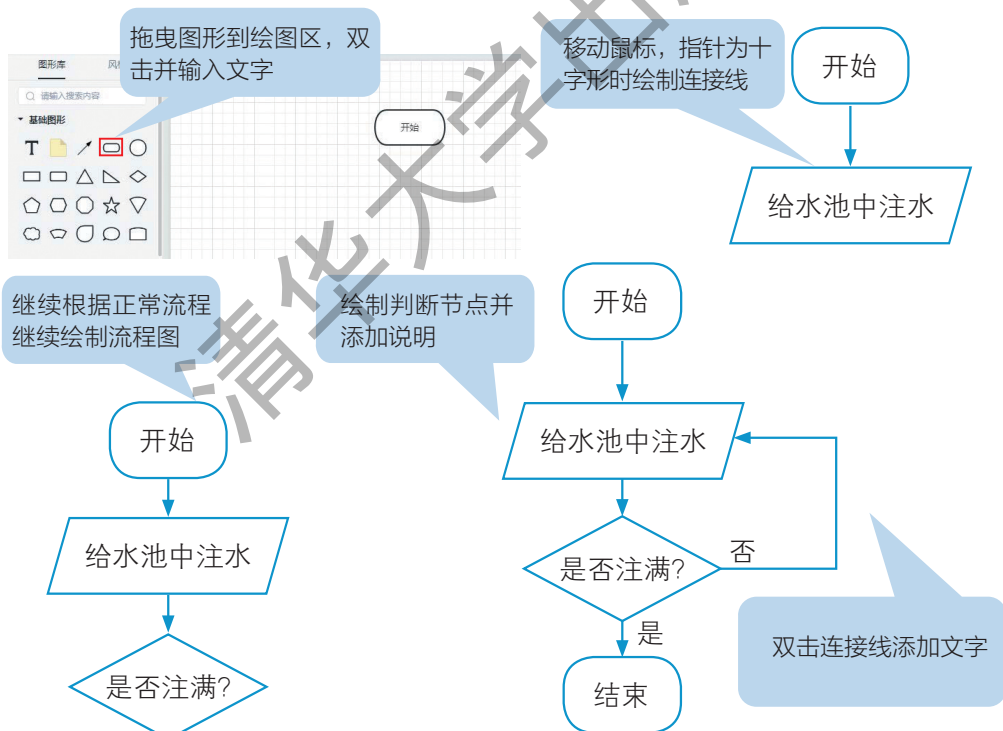


图 2.2.6 绘制流程图

探索

在计算机上利用流程图绘制工具，绘制出求长方形面积的流程图。

四、用流程图描述算法

利用流程图描述算法，是算法描述的常用方式之一。

我们以“输入三个数，找到其中最大的那个数”这一问题为例。

这个算法的自然语言描述如下。

算法名称：输入三个数，找到其中最大的数

输出：最大的数（max）

步骤：

- （1）输入三个数，分别放到 a ， b ， c 中。
- （2）将 a 与 b 进行比较，将其中较大的数放入 \max 中。
- （3）将 c 与 \max 进行比较，将其中较大的数放入 \max 中。
- （4）输出 \max 的值。

目的：找到输入的数 a ， b ， c 中的最大值，并放在 \max 中。

结果：输出最大的数（ \max ）。

用流程图描述这一算法如图 2.2.7 所示。

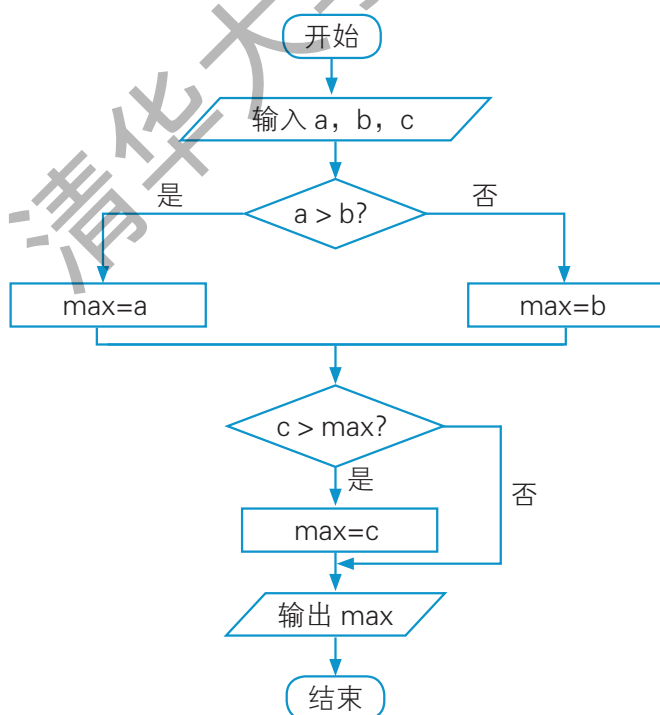


图 2.2.7 “找出三个数中最大的数”算法流程图



在这个流程图中，进行数据输入时，输入的三个数分别为 a ， b ， c ，最大值用 \max 表示。我们首先可以将 a 和 b 进行比较，把其中较大的数放入 \max 中，再将 \max 与 c 比较，将两者中较大的数放入 \max 中。最后将 \max 输出，此时 \max 中的值就是 a ， b ， c 中的最大值。



挑战

找出三个数字中的最大值（数），比较容易，那么要是 50 个数、100 个数甚至更多个数呢？

我们以 50 个数为例，求解的步骤如下：

(1) 最大的数为 \max （初始值为 0），初始化变量 $i=1$ 。

(2) 若 $i \leq 50$ ，执行下一步，否则执行 (6)。

(3) 输入 x 。

(4) 若 $\max < x$ ，则 $\max = x$ ；否则 \max 不变。

(5) $i = i + 1$ ，转 (2)。

(6) 输出 \max 。

参考以上描述，在给定的流程图框架（图 2.2.8）中完成“输入 50 个数，找到最大数”的流程图。

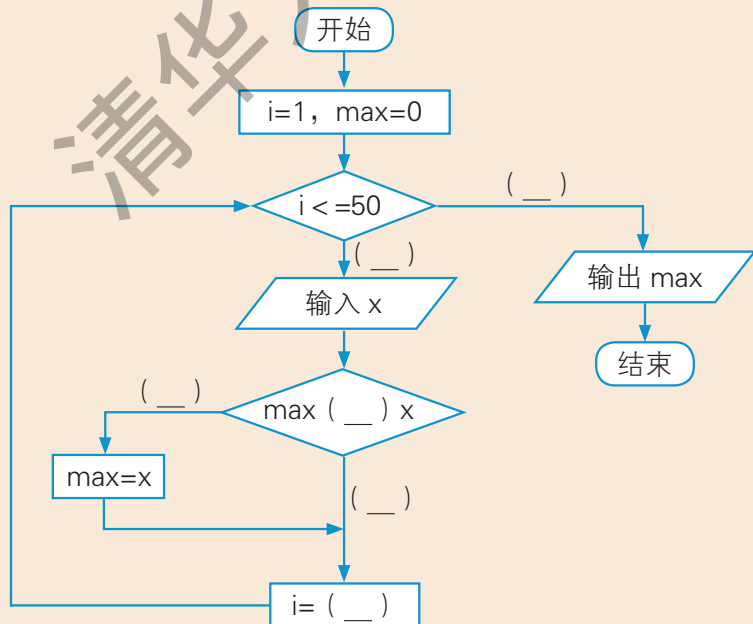


图 2.2.8 “输入 50 个数，找到最大数”流程图

第 3 课

用程序描述算法



本课中你将学习：

- 如何用伪代码描述算法
- 如何用程序描述算法
- 如何用程序实现简单的算法

我们常用计算机代码来描述算法。除此之外，我们还常用图形化程序的形式来描述并实现算法（图 2.3.1）。

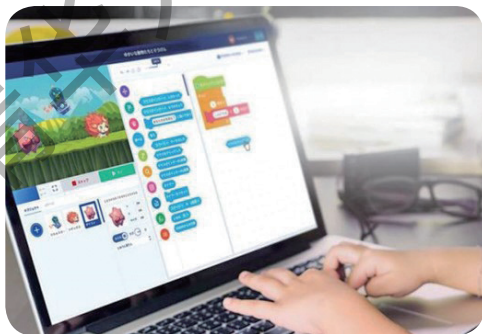


图 2.3.1 图形化程序

一、用伪代码描述算法

伪代码是一种用于表示程序和算法的非正式语言。一般情况下伪代码是无法执行的，它们的存在只是为了帮助我们理解算法，其本身并不能运行或得到结果。



以下是一个计算正方形面积的“图形化伪代码”（图 2.3.2）。

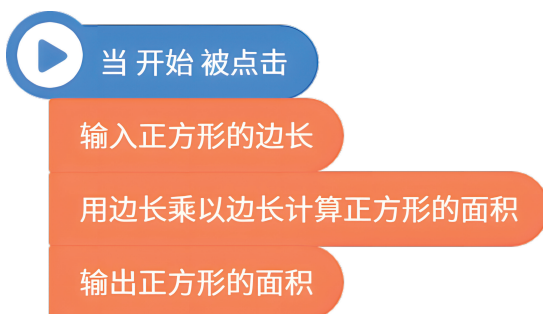


图 2.3.2 计算正方形面积的“图形化伪代码”

探索

思考并讨论，以下哪些是伪代码的特点？

- (1) 能够描述算法；
- (2) 能够运行得到结果；
- (3) 是正式的编程语言；
- (4) 能够实现算法；
- (5) 能够帮助我们理解算法。

二、用计算机程序描述算法

计算机程序依赖算法进行设计、编写，程序也能实现算法、描述算法。

1. 计算机程序

计算机程序是一系列指令的集合，用于告诉计算机执行特定的任务或完成特定的操作。它是实现特定功能的一种方式，可以通过编程语言编写。

阅读

计算机程序需要依靠一些计算机代码来编写，这些代码也被称为计算机语言。计算机语言是人工语言的一种，不同的语言有不同的应用场景。图 2.3.3 列出了几种常用的计算机语言。

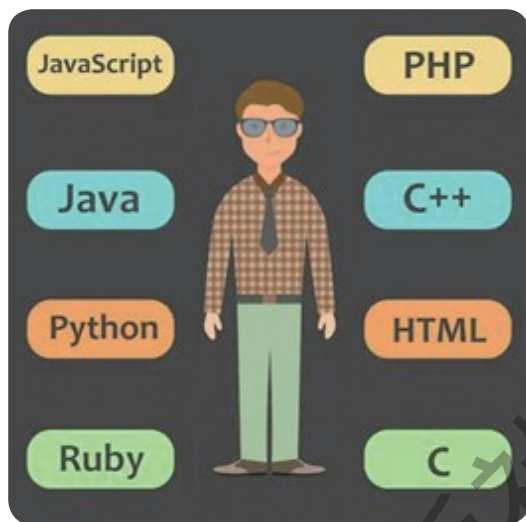


图 2.3.3 常见的几种计算机语言

2. 用计算机程序描述算法

计算机程序可以用来描述和实现算法。前面已经学习过用自然语言描述“求两个整数的和”，本课我们在此基础上作具体描述。

(1) 将 a 和 b 相加，结果赋值给 sum ；

(2) 返回 sum 作为输出。

我们可以用 Python 语言写的程序，来描述这“求解整数和”的过程。

```
# 从键盘上输入 a
a=input(" 输入变量 a")
# 从键盘上输入 b
b=input(" 输入变量 b")
# 对 a、b 进行数据类型转换
a=int(a)
b=int(b)
# 用 sum 变量存储 a+b 的结果
sum=a+b
# 输出 sum
print(sum)
```



探索

借助图形化的编程工具编写“输入 10 个数字，输出最小值”的程序。

3. 用图形化编程实现简单算法

用图形化编程工具绘制一个正方形（图 2.3.4），打开编程工具页面，将背景和角色隐藏起来，删除角色中的示例代码块。

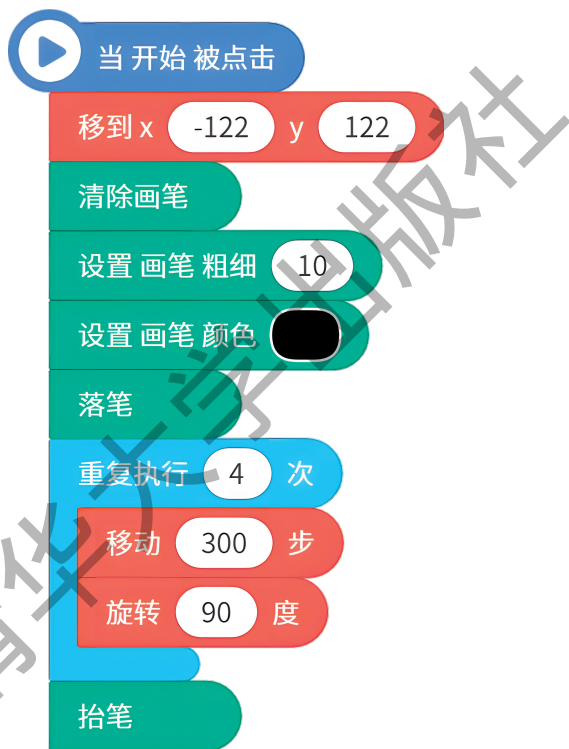


图 2.3.4 绘制正方形代码块

编写以上代码，然后单击“开始”按钮，程序就会运行画出一个正方形。

绘制正方形程序可以分解如下。

（1）确定初始位置

首先设定一个初始位置，也就是正方形左下角的那个顶点的位置，将初始位置设置为（-122，122）。在动作盒子中找到这一积木块，然后修改参数（图 2.3.5）。



移到 x -122 y 122

图 2.3.5 设定初始位置

(2) 初始化画笔

在画笔盒子中设置画笔粗细、颜色等参数（图 2.3.6）。



图 2.3.6 初始化画笔

(3) 绘制形状

移动 300 步 表示画出一条步长为 300 的线段。

旋转 90 度 表示画完直线后调整下一次画笔的方向，这和我们在纸上

绘制一个正方形类似。

因为绘制的是一个正方形，所以共需要重复执行 4 次（图 2.3.7）。



图 2.3.7 绘制形状

(4) 完成绘制

抬笔 表示最后执行抬笔的操作，完成绘制。

所绘制的正方形如图 2.3.8 所示。



图 2.3.8 正方形

同学们有没有发现，用计算机图形化编程工具绘制正方形的过程与在纸上画一个正方形是类似的。



挑战

借助图形化的编程工具绘制出一个等边三角形。

第 4 课

活动：数字加密



本课中你将体验：

- 设计一个数字加密的小程序
- 用自然语言、流程图、图形化编程描述算法



活动任务

在生活中密码无处不在（图 2.4.1），打开手机、计算机需要密码，取款、转账也需要密码。密码安全关系着用户的信息和隐私以及财产的安全，因此设置一个安全的密码是非常重要的。

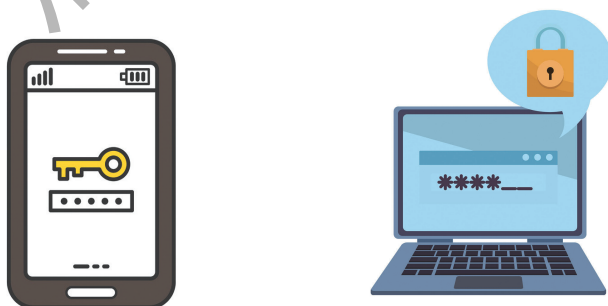


图 2.4.1 密码无处不在



想一想

生活中，还有哪些需要应用密码的场景？

数字密码是生活中最常见的一种密码形式，如何提高密码的安全性呢？小华在分析“数字加密”问题时，决定尝试用“随机密钥”的加密方法进行算法设计，这种方式需要获得一串随机密码，相当于一把钥匙，使用这把随机密码的钥匙，通过加密算法得到加密的数字，从而提高密码的安全性（图 2.4.2）。



图 2.4.2 数字加密



读一读

数据加密是通过加密算法和加密密钥将明文转变为密文，而解密过程则恰好相反，是通过解密算法和解密密钥将密文恢复为明文。

数据加密是计算机系统保护信息的一种最可靠的办法。它利用密码技术对信息加密，这样在信息传输的过程中，即使信息中途被窃取，也不会轻易被读取和破译，从而起到保护信息安全的作用。



议一议

随机密钥的加密算法步骤：首先输入要加密的数字，生成随机密钥，通过数字与密钥之间的运算进行加密，这些运算包括数字的加减乘除等，最终得到密码结果。

请仿照图 2.4.3 左图中的加密过程，试写出右图加密后的密码。

要加密数字: 3 5 1

+

随机密钥: 1 3 6

数字加密: 4 8 7

加密后密码: 4 8 7

要加密数字: 5 7 9

+

随机密钥: 3 6 5

数字加密:

加密后密码:

图 2.4.3 数字加密

思考一下，我们要读取原本的数字（如 3 5 1、5 7 9 等），应该怎么办？



做一做

设计算法：试着分步骤写出数字加密的设计思路。

1. _____
2. _____
3. _____
4. _____



读一读

随机密钥可以通过随机数来生成（图 2.4.4），随机数是指计算机随机生成的数字，就像掷骰子时，无法事先判断掷出的骰子的点数是多少，这就是随机数字。



图 2.4.4 投掷骰子模拟随机密钥生成



做一做

画出“随机密钥加密”算法设计的流程图。

①在图形化编程工具中获取输入的数据。选择外观盒子中的“询问【 】并等待”积木，并输入“请输入需要加密的数字：”（图 2.4.5）。



图 2.4.5 询问积木

②创建变量“密码”、“密钥”和“数字”，并对新建变量的名称、属性和主题进行设置（图 2.4.6）。



图 2.4.6 新建变量

③使用函数盒子，在定义好的函数中找到“生成随机密钥”“数字加密”（图 2.4.7 和图 2.4.8）。



图 2.4.7 函数声明

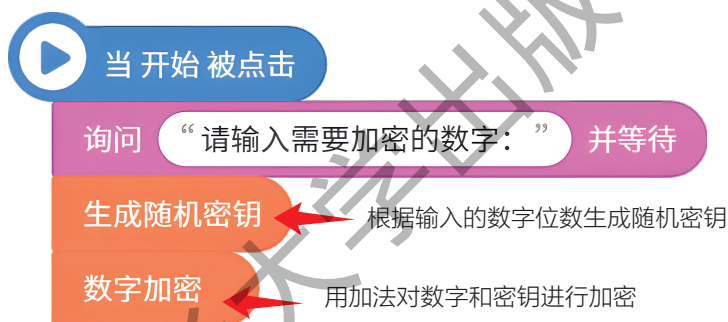


图 2.4.8 “生成随机密钥”和“数字加密”函数

④定义函数“生成随机密钥”(图 2.4.9)。设置变量“密钥”由 0~9 间的随机整数组成，并根据输入的数字位数生成随机密钥。



图 2.4.9 定义函数“生成随机密钥”

⑤定义函数“数字加密”(图 2.4.10)。先将变量“密钥”转换为数字类型，

再设置变量“密码”的值为“数字”+“密钥”。



图 2.4.10 定义函数“数字加密”

⑥输出数据。可以使用外观盒子中的“对话【】”或“新建对话框【】”积木，输出数据是“密码”变量。可以根据实际需要编写输出的内容，比如把输出的文本改为“加密后的密码是：”，将输出文本与“密码”变量放在一起（图 2.4.11）。

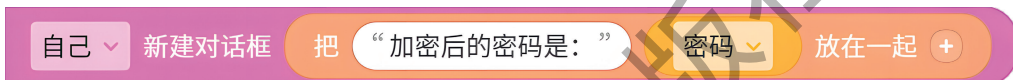


图 2.4.11 输出密码



议一议

请你在表 2.4.1 中记录要加密的数字、随机密钥和生成的密码。选择其中一组加密数字和随机密钥，让小伙伴尝试生成密码，看一看是否能得到相同的结果。

表 2.4.1 数字加密表

加密的数字	随机密钥	生成的密码



做一做

围绕“数字加密”问题，设计一款自己的“数字加密”小机器人来提高安全性吧！



评一评

设计完程序记得保存并分享给你的小伙伴，展示你的作品吧！

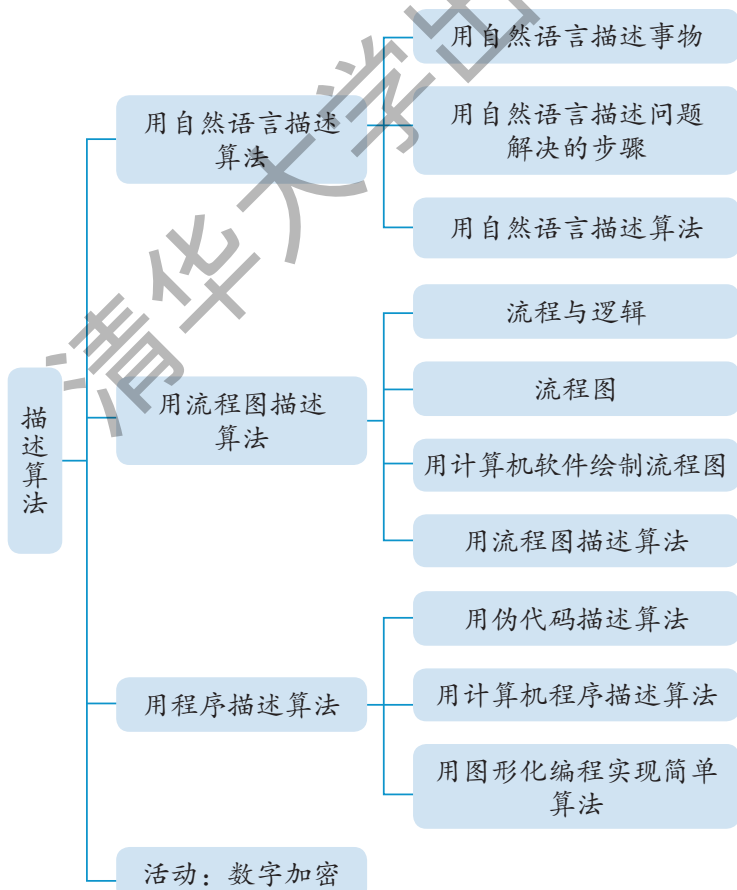
内容	遇到的问题	解决方法	感悟收获	总体评价
想一想				☆☆☆☆☆
读一读①				☆☆☆☆☆
议一议①				☆☆☆☆☆
做一做①				☆☆☆☆☆
读一读②				☆☆☆☆☆
做一做②				☆☆☆☆☆
议一议②				☆☆☆☆☆
做一做③				☆☆☆☆☆

小结与评价

本单元你学习了：

- ◆ 算法思想
- ◆ 用算法描述事物
- ◆ 生活中的算法
- ◆ 编写测算面积的程序

知识梳理





学习检测

1. 常用的算法描述方式有哪些？

2. 用自然语言描述制作蛋糕的步骤。



反思评价

在本单元学习过程中，肯定少不了与他人互动交流、参与作品制作等活动。现在请就此进行总结与反思，以更好地促进自身成长。

1. 从同伴那里学到过什么？
2. 向同伴分享过哪些观点？
3. 所运用的工具、方法是否得当？
4. 所设计的作品有哪些可以优化的地方？

3

算法的控制结构

📖 本单元你将学习：

- 了解和运用顺序、分支、循环三种基本的算法结构
- 发现生活中三种算法控制结构的简单案例并对其进行描述
- 如何分析算法的执行过程与结果，并能通过编程验证

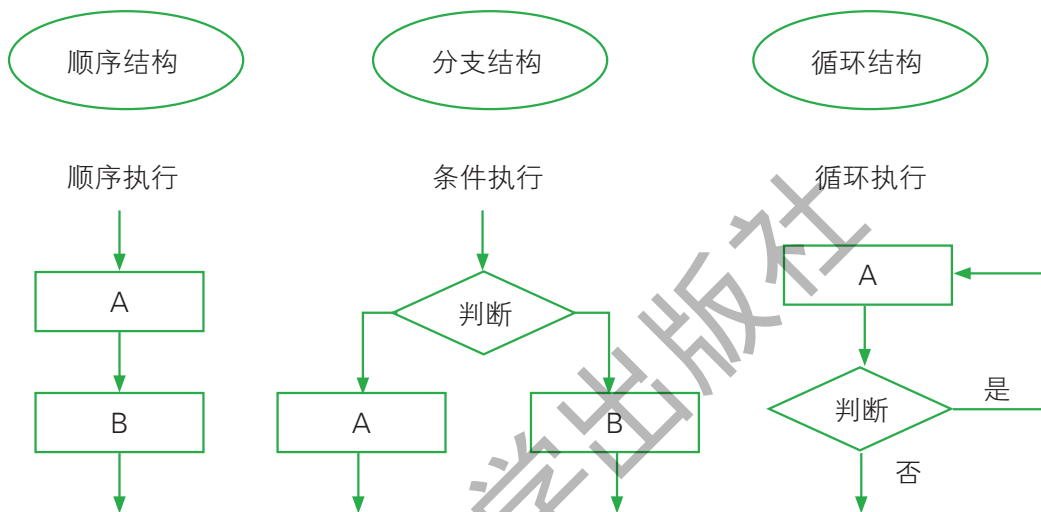
算法的控制结构是按照某种顺序执行的一系列动作，用于解决某个问题。无论多么复杂的算法均可通过顺序、分支、循环这三种基本控制结构来实现。

本单元通过算法实例，帮助我们理解算法的三种基本结构——顺序结构、分支结构和循环结构，了解它们的应用，学会分析算法的执行过程与结果，并能够利用这三种基本控制结构来解决问题。



学习热身

算法的控制结构有哪些？根据你所学的知识，尝试说一说，这些结构在生活中有哪些实例？



你知道吗？

用算法解决某个问题时，有时一步一步按顺序执行就能完成，有时执行到某一步后要根据条件判断下一步该干什么，有时还需要重复执行某些步骤。对应上述三种情况，算法的控制结构可分为顺序结构、分支结构和循环结构三种。理论和实践证明，无论什么算法都可以由顺序、分支和循环这三种基本控制结构及它们的组合构造出来。

算法效率
算法思维
算法应用



第1课

顺序结构



本课中你将学习：

- 什么是顺序结构
- 如何用顺序结构解决生活中的实际问题
- 顺序结构的描述与程序实现

生活中，很多问题都可以通过执行一系列的步骤来解决。例如，汉字的书写应遵循笔画顺序，机场的安检应遵循安检顺序，去医院看病应遵循就医流程，工业的生产应遵循工艺流程等。同样，在算法中也有这样的顺序控制结构。

一、认识顺序结构

顺序结构是指算法中各个步骤按照先后顺序依次执行的结构，是最简单的一种算法结构。计算机按顺序处理步骤，当一个步骤处理完毕，自动转到下一个步骤。顺序结构流程图如图 3.1.1 所示。

妈妈让悦悦把新买的老抽与料酒倒入空的调料瓶中。悦悦一不小心，将老抽倒进了料酒瓶里，将料酒倒进了老抽瓶里。同桌之间互相说一说，悦悦用什么方法能将两种调料调换过来？

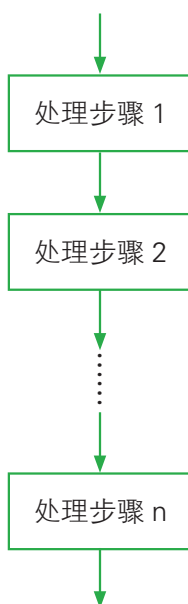


图 3.1.1 顺序结构流程图



该怎么办呢？

可以利用空杯子进行调换（图 3.1.2）。



图 3.1.2 怎样调换瓶子中的老抽和料酒

请在如图 3.1.3 所示的交换调料流程图中填空，形成完整的流程图。

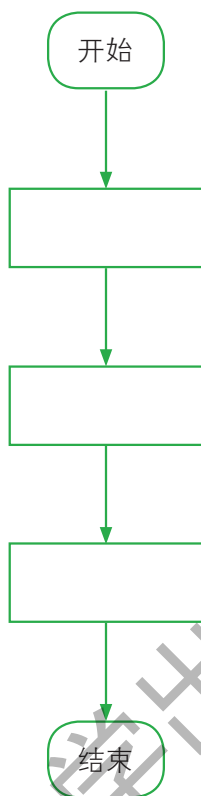


图 3.1.3 交换调料流程图

阅读

变量，顾名思义，就是变化的量。它是计算机程序中储存计算结果的一个空间。在建立变量时，我们要给它起一个名字，例如“序号”“总数”或“a”等，但最好有特定的含义，以方便理解。我们可以把变量想象成计算机中临时用来存储数据的盒子。

二、顺序结构在计算机中的实现

交换“老抽”与“料酒”可以用图形化的程序实现，在这里需要新建三个变量 a, b, c 并将其初始化（图 3.1.4），即变量 a 用来存储“老抽”，变量 b 用来存储“料酒”，变量 c 则充当“中转”的杯子（默认存储数据为 0，也就是空）。绘制一个“交换”按钮，并为交换按钮添加脚本。



图 3.1.4 新建变量并将其初始化

要想让老抽和料酒互换, 只需要让 $c=a$ 、 $a=b$ 、 $b=c$ 即可, 具体实现过程如图 3.1.5 所示。

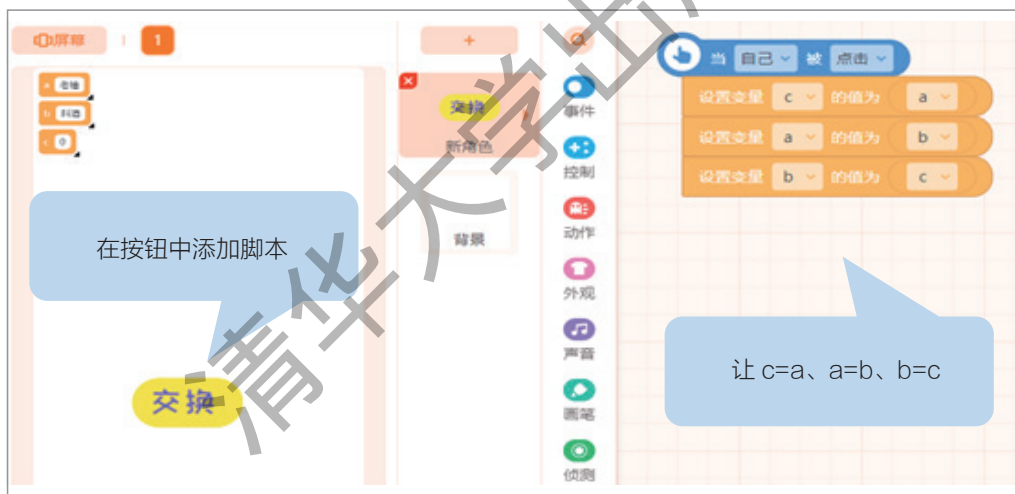


图 3.1.5 执行交换过程

挑战

1. 利用计算机流程图绘制工具, 绘制出“老抽”与“料酒”互换的流程图。
2. 在“计算三角形的面积”这一程序中, 主要就是一个基本的顺序结构(图 3.1.6)。

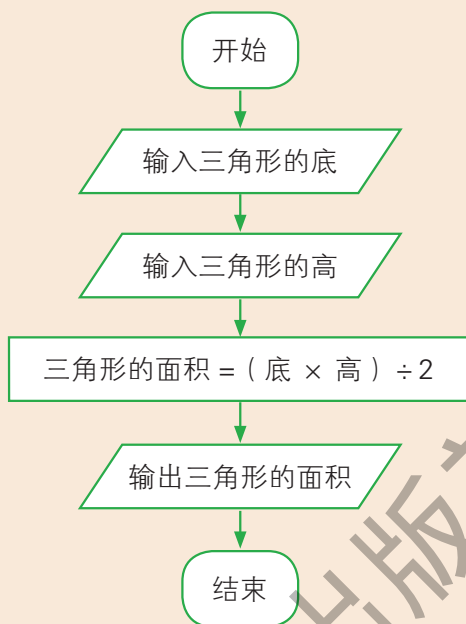


图 3.1.6 计算三角形面积的流程图

请用图形化的编程工具来实现。

第2课

分支结构



本课中你将学习：

- 什么是分支结构
- 如何用分支结构解决生活中的实际问题
- 分支结构的描述与程序实现

在生活中，我们经常会遇到有多种选择的情况，例如学校会根据天气选择是否组织室外活动：天气晴朗则开展室外活动，否则就会停止室外活动。

在算法中，也经常会遇到要进行选择的情况，要通过判断特定的条件，选择不同的执行路径。

一、认识分支结构

分支结构也称为选择结构，表示处理问题的过程中需要根据特定的条件选择多个分支中的某一个分支执行。分支结构有单分支、双分支和多分支三种情况。

冬季的早上，小明上学前在穿好外套后都要查看一下天气情况（对于计算机程序而言，小明查看天气情况相当于获得了天气情况信息，在流程图中表现为“输入天气情况”），如果发现降温了，就把外套换成羽绒服，再出门上学。这个算法包含的分支结构是单分支结构，它的流程图如图 3.2.1 所示。

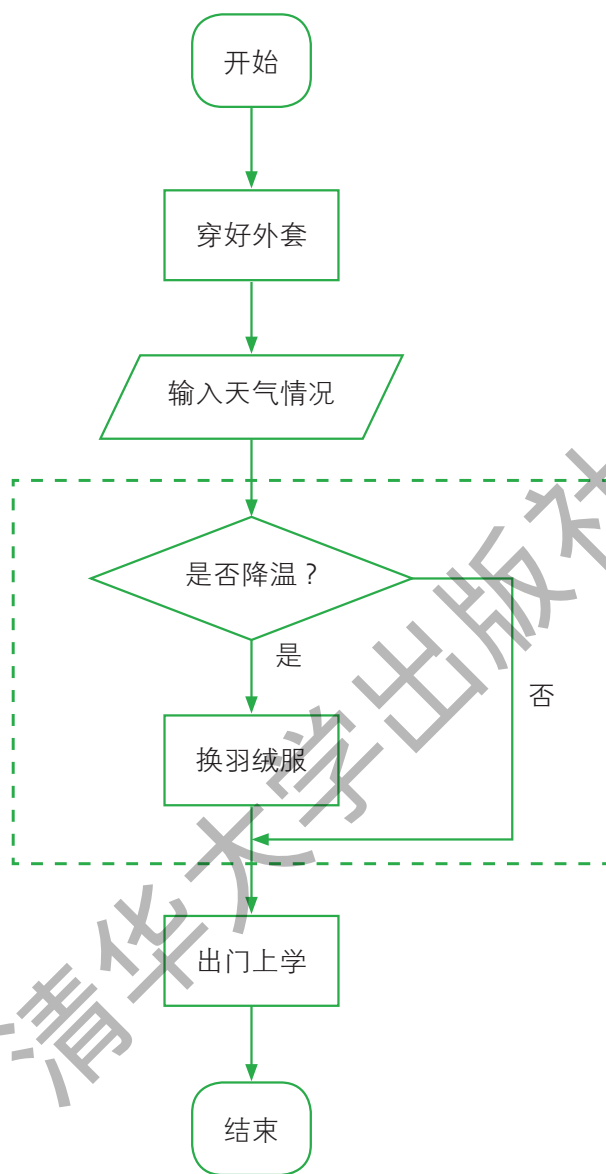


图 3.2.1 单分支分支结构流程示意

执行单分支结构时，先判断条件是否成立：如果条件成立，则执行某项操作，然后执行后续的操作；如果条件不成立，则直接执行后续的操作。

上述情况也可以用另外的方式实现：小明在上学前查看一下天气情况，如果发现降温了，就加穿羽绒服；如果没降温，就只穿外套。穿好外套后，再出门上学。这个算法中包含的分支结构是双分支结构，它的流程图如图 3.2.2 所示。

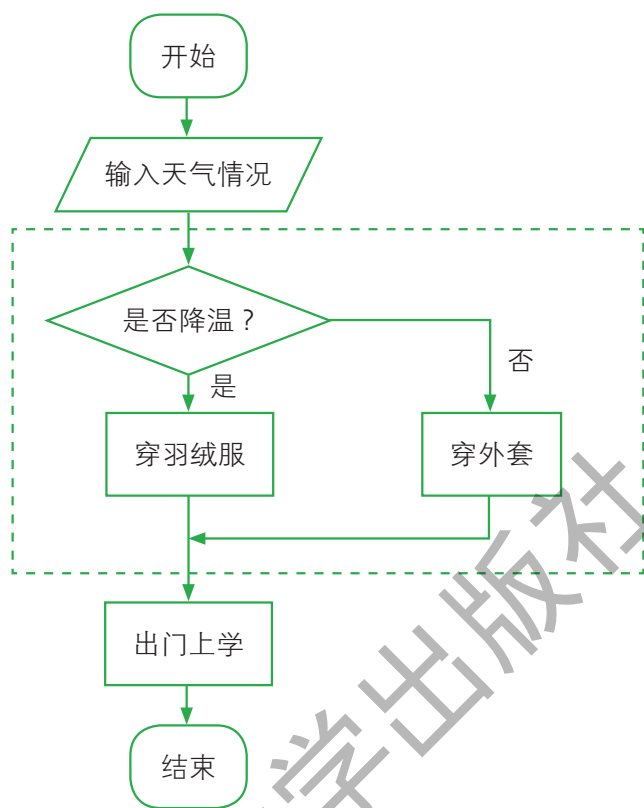


图 3.2.2 双分支分支结构流程示意

执行双分支结构时，先判断条件是否成立：如果条件成立，则执行某项操作，然后执行后续的操作；如果条件不成立，则执行另外一项操作，然后执行该操作紧连的后续操作。

无论是单分支、双分支还是多分支结构，判断/决策节点（表示根据条件进行判断或决策的步骤，通常用菱形表示，图 3.2.3）都是必需的，其是典型的分支结构标志。

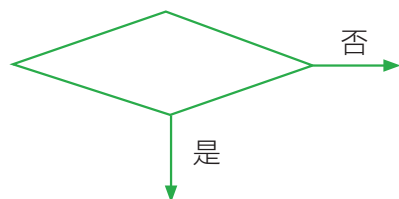


图 3.2.3 判断/决策节点

探索

任意给出三个正数 a , b , c , 它们分别代表三条线段的长度, 初步判断这三条线段能否构成一个三角形。

算法分析: 判断条件为三条线段中的任意两条线段长度之和大于第三条线段的长度, 这里只判断 “ $a + b > c$ ” 的情况。如果条件成立, 则显示 “长度为 a , b , c 的三条线段有可能构成一个三角形”; 如果条件不成立, 则显示 “长度为 a , b , c 的三条线段不能构成一个三角形”。

请完善下面的流程图 (图 3.2.4)。

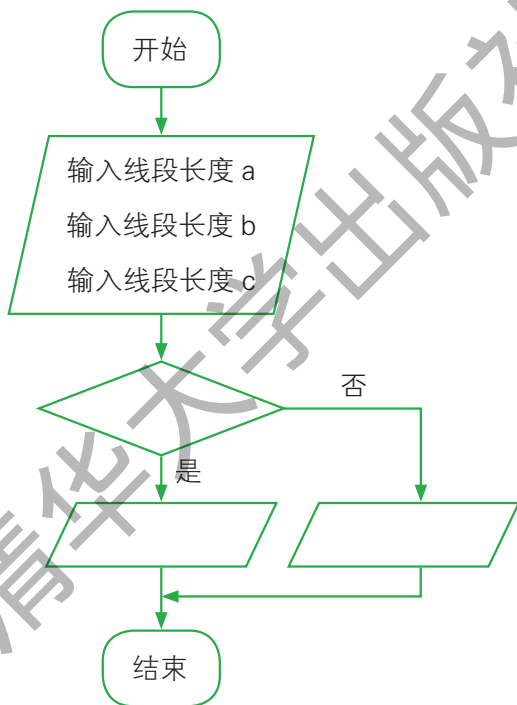


图 3.2.4 初步判断三条线段是否有可能构成一个三角形的流程图

二、分支结构在计算机中的实现

判断三条线段能否构成三角形的条件为: 三角形的任意两条边的长度之和大于第三条边的长度。也就是说, 三条线段如果能构成三角形, 必须同时满足三个条件, 即 $a + b > c$, $a + c > b$, $b + c > a$ 。

请在图 3.2.5 所示的流程图中填空并完善该流程图。

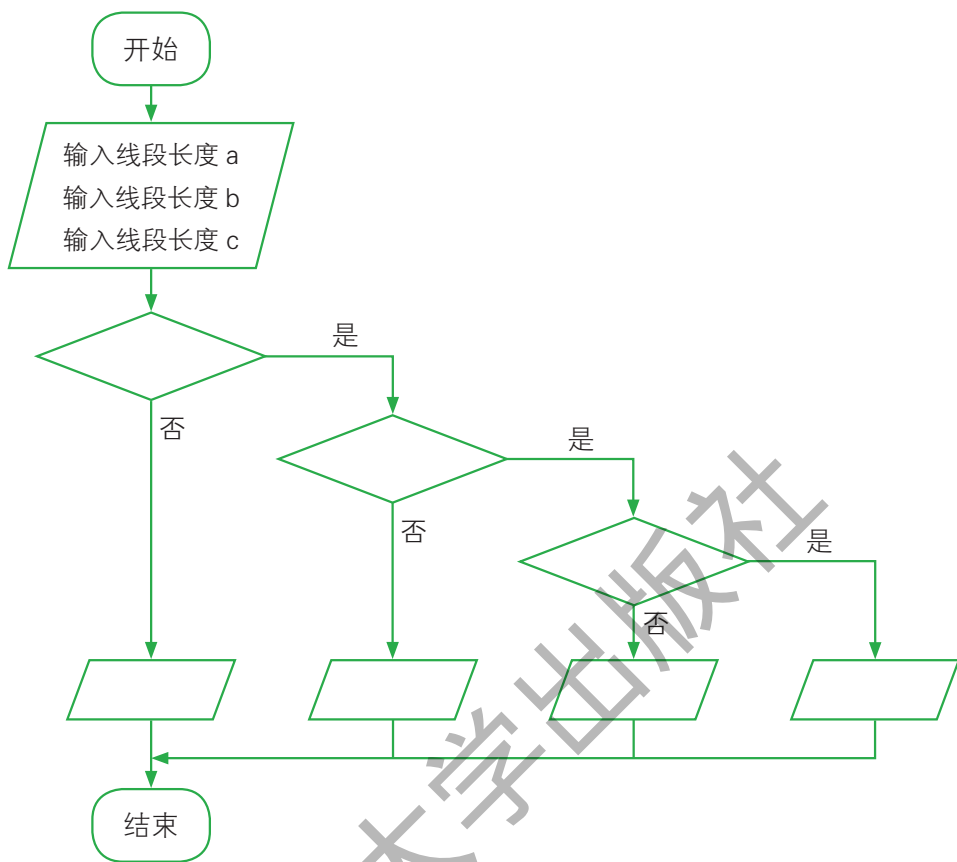


图 3.2.5 判断三条线段 a , b , c 能否构成一个三角形的流程图

如图 3.2.5 所示，构成三角形的条件中，要求同时满足三个条件，我们通常使用连接词将条件相连。例如，把构成三角形的条件表示为“ $a + b > c$ 且 $a + c > b$ 且 $b + c > a$ ”这样的形式。在图形化编程工具中也有对应的积木块，可以用来连接不同条件的积木，如图 3.2.6 所示；当选择“且”选项时，若两边输入的条件同时都满足，则执行“是”分支下的语句，否则执行“否”分支下的语句；当选择“或”选项时，两边只要有一个条件满足，就执行“是”分支下的语句，否则执行“否”分支下的语句。

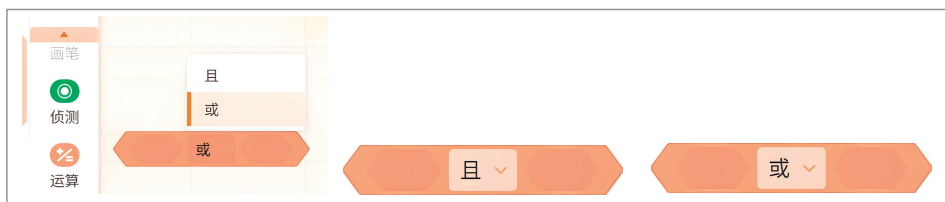


图 3.2.6 连接不同条件的积木

下面是具体的图形化程序实现过程，如图 3.2.7 所示。



图 3.2.7 输入 a , b , c 三条边的边长

可以通过如图 3.2.8 所示代码去直接判断是否 $a + b > c$ 且 $a + c > b$ 且 $b + c > a$ 。



图 3.2.8 用“且”积木块实现 a , b , c 为三边长能否构成一个三角形的判断

为了简化，我们还可以采用分支“嵌套”的形式去实现“且”这个操作，具体如图 3.2.9 所示。

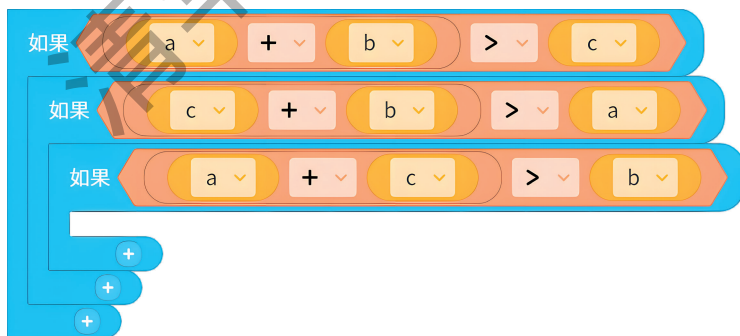


图 3.2.9 分支结构的“嵌套”

挑战

用图形化编程工具实现“判断一个数是不是偶数”。

第3课

循环结构



本课中你将学习：

- 什么是循环结构
- 如何用循环结构解决生活中的实际问题
- 循环结构的描述与程序实现

生活中有许多重复的事物：太阳东升西落，日复一日；钟表指针一圈圈地旋转；工厂流水线上—批批产品按照同一工艺流程加工生产。在算法中可以用循环结构提高解决重复性问题的效率。

一、认识循环结构

循环结构是一种在程序中重复执行一段代码的结构。它允许程序多次执行同一段代码，从而实现对重复任务的处理。

循环结构是计算机程序能够比人类处理事物更加高效的关键之一，计算机最擅长的就是多次、重复地处理某一事物，这和我们人类有很大的不同。当然循环结构可以是有条件的有限循环（循环有限的次数），也可以是无限循环。

电水壶烧水（流程图如图 3.3.1 所示）的例子，就包含了循环结构。

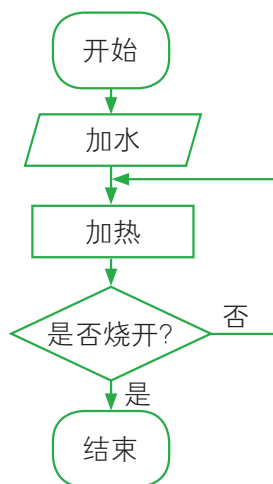


图 3.3.1 电水壶烧水的流程图

在注水和判断是否烧开这个过程中就有一个循环存在，在没烧开的情况下，“给水加热”这一动作就要重复进行。若条件满足“水烧开”，则结束循环，按照顺序继续执行后面的流程。

在生活中经常也有这样的例子：我们在马路上经常会遇到红绿灯。一般重要路口的红绿灯都是 24 小时工作的，图 3.3.2 是用顺序结构流程图描述的一组红绿灯的工作过程，通过观察可以发现：全部工作过程基本上是前三个步骤的重复。



图 3.3.2 用顺序结构描述红绿灯的工作过程



用顺序结构描述红绿灯的工作过程非常烦琐，如果采用循环结构，就能简洁、明了地描述这个工作过程，如图 3.3.3 所示。

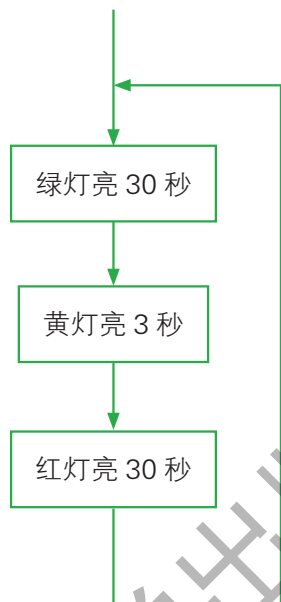


图 3.3.3 用循环结构描述红绿灯的工作过程

循环结构一般由条件和循环体两部分组成。通过判断条件是否满足，决定是否执行循环体，循环体是循环执行的操作步骤，如图 3.3.4 所示。

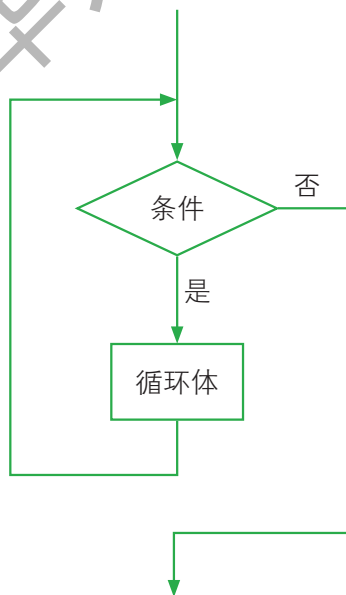


图 3.3.4 循环结构示意

二、循环结构在计算机中的实现

循环结构分为两种：一种是当型循环，另一种是直到型循环。

(1) 当型循环。如图 3.3.5 所示，先判断条件是否满足，若条件满足，则执行循环体操作后，再判断条件；当条件不满足时终止循环。因此，当型循环的循环体有可能一次也执行不到。

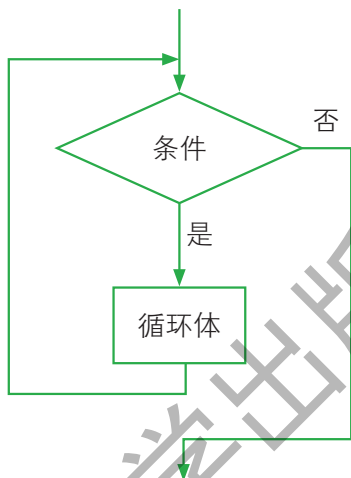


图 3.3.5 当型循环结构

(2) 直到型循环。先执行循环体的操作，然后判断条件，若条件不满足则返回，重复执行循环体，直至条件满足才终止循环，如图 3.3.6 所示。因此，直到型循环的循环体至少要执行一次。

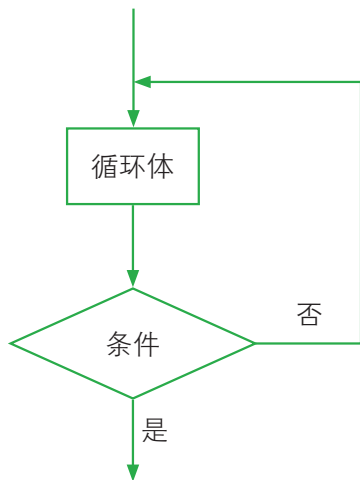


图 3.3.6 直到型循环结构



我们用循环结构实现 100 以内整数累加和的程序。

算法分析：求 100 以内整数累加和。设 s 为累加和的变量，100 以内的每个加数可以用变量 i 代替。如图 3.3.7 所示，一开始让 s 和 i 都等于零，循环执行 $i = i + 1$ 和 $s = s + i$ ，直到 i 大于 100 结束循环；最后输出 s 即为“100 以内整数的累加和”。

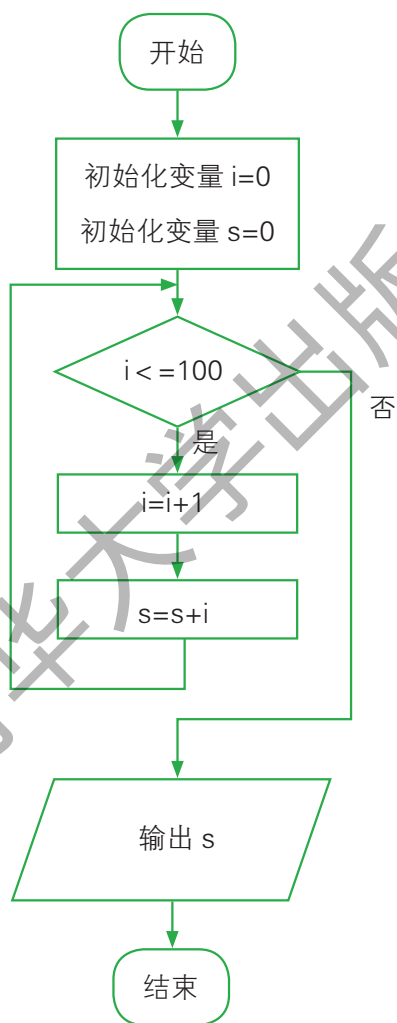


图 3.3.7 计算 100 以内整数累加和流程示意

在图形化编程软件中如何编写计算 100 以内整数累加和这个程序呢？请你参照流程图，根据图 3.3.8 所示的核心图形化程序，设计并运行整个程序，验证结果。

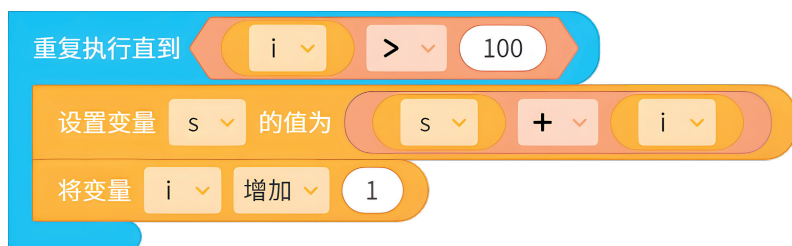


图 3.3.8 求 100 以内整数累加和的核心程序

不同的编程语言中，循环结构的格式也不同。在图形化编程语言中，一般有如下三种循环结构格式：

(1) 永久循环（见图 3.3.9 (a)）。反复执行直到强制结束，一般只在一些特殊情况下使用，如前面提到的路口红绿灯。

(2) 按条件循环（见图 3.3.9 (b)）。反复执行，直到满足指定的条件后结束循环。

(3) 按次数或数值区间循环（见图 3.3.9 (c)）。执行指定的次数或从开始数据起按某种条件到终止数据后结束循环。

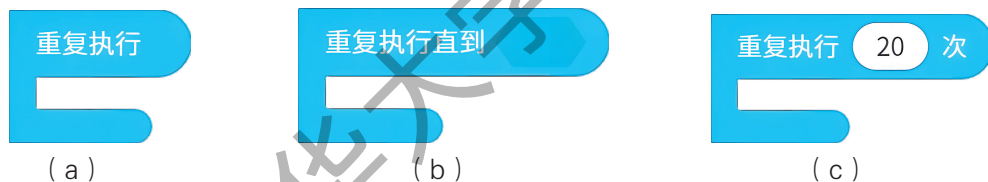


图 3.3.9 三种格式的循环结构

挑战

画出求 100 以内奇数和的流程图，请尝试用图形化编程工具编写程序，并运行程序验证结果。

第4课

跨学科活动：绘制莲花



本课中你将体验：

- 设计绘制莲花的小程序
- 通过美术、数学等学科知识了解莲花的外形
- 综合运用顺序、分支、循环结构优化程序
- 制作并保存好你的程序，并分享给小伙伴



活动任务

假期，小明一家出去旅游，在湖中看到了一片盛放的莲花（图 3.4.1），莲花有很多花瓣，每片花瓣都伸展开来，好看极了。小明被这些莲花深深地吸引了，他想利用程序画出一朵好看的莲花来让大家欣赏。如果你是小明，如何利用图形化软件绘制莲花呢？



图 3.4.1 莲花



想一想

小明根据美术课上学到的画莲花知识，仔细观察莲花的外形（图 3.4.2），发现莲花是一个三层结构，每一层由 10 片花瓣构成。画好花瓣是绘制莲花的关键，一个花瓣画好了，然后再画出一层的花瓣，这样每层的花瓣都画好了，莲花就画好了。



图 3.4.2 莲花的外形

通过分析，可以确定：

①莲花的层数：_____；

②莲花每层由多少花瓣组成：_____。

请用自然语言描述绘制莲花的步骤：

首先_____，其次_____，最后_____。



做一做

按照步骤，画出“绘制莲花”的流程图。



做一做

利用图形化编程软件，绘制出你心中最美的那朵莲花吧！

①确定初始位置。首先设定一个初始位置作为“花心”，这里将初始位置设置在 $(0, 0)$ ，并且面向 90° 方向（图 3.4.3）。



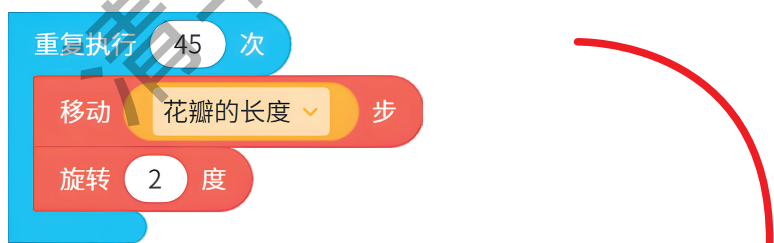
图 3.4.3 确定初始位置

②初始化画笔。在画笔盒子中，找到设置画笔颜色和粗细的积木块，将画笔颜色设置为红色，画笔粗细设置为 5（图 3.4.4）。



图 3.4.4 初始化画笔

③绘制 $\frac{1}{4}$ 圆弧。花瓣是由两个 $\frac{1}{4}$ 圆弧组成的，先来绘制一个 $\frac{1}{4}$ 圆弧（图 3.4.5）。

图 3.4.5 绘制 $\frac{1}{4}$ 圆弧

利用数学课上所学的计算角度的知识，为了能够画出 $\frac{1}{4}$ 圆弧，我们需要确保“重复次数 \times 旋转角度 $= 90^\circ$ ”。绘制 $\frac{1}{4}$ 圆弧程序中，旋转角度是 2° ，重复执行 45 次， $2^\circ \times 45 = 90^\circ$ 。其实，旋转的角度并不是固定的，可以设置大一点，也可以设置小一点，但是要确保“重复次数 \times 旋转角度 $= 90^\circ$ ”。

④绘制一片花瓣。接下来再绘制另外一个 $\frac{1}{4}$ 圆弧，拼成一片由两个圆弧组成的花瓣。要先旋转 90° ，否则就和之前的 $\frac{1}{4}$ 圆弧连在一起形成一个半圆了（图 3.4.6）。

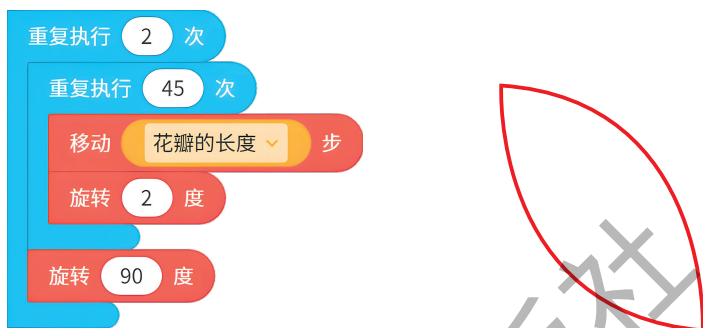


图 3.4.6 绘制一片花瓣

⑤绘制外层花瓣。每层花瓣有 10 片，需要使用循环结构。10 片花瓣，需要循环 10 次，旋转 $\frac{360^\circ}{10}=36^\circ$ 就可以绘制出另一片花瓣（图 3.4.7）。

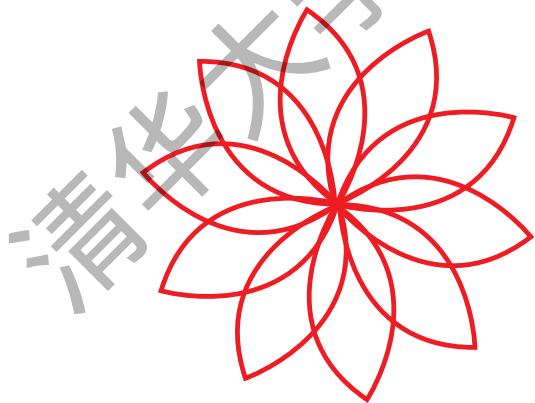


图 3.4.7 外层花瓣效果示意

⑥绘制莲花。莲花有三层，可以使用循环结构。每一层花瓣的大小、颜色都不一样，这就需要重新设置画笔的颜色并减小花瓣的长度。花瓣的长度是一个变量，需要新建变量“花瓣的长度”。每循环一次将变量“花瓣的长度”减少 1（图 3.4.8）。再调整画笔的颜色，一朵莲花就绘制好了（图 3.4.9）。

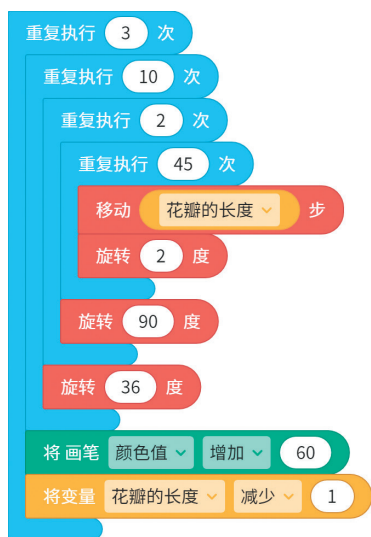


图 3.4.8 绘制莲花程序



图 3.4.9 莲花效果示意



议一议

小明如果想绘制一朵每层有 12 瓣的莲花，每次需要旋转多少度？应该如何改写这个程序呢？



评一评

设计完程序记得保存并分享给你的小伙伴，展示你的作品吧！

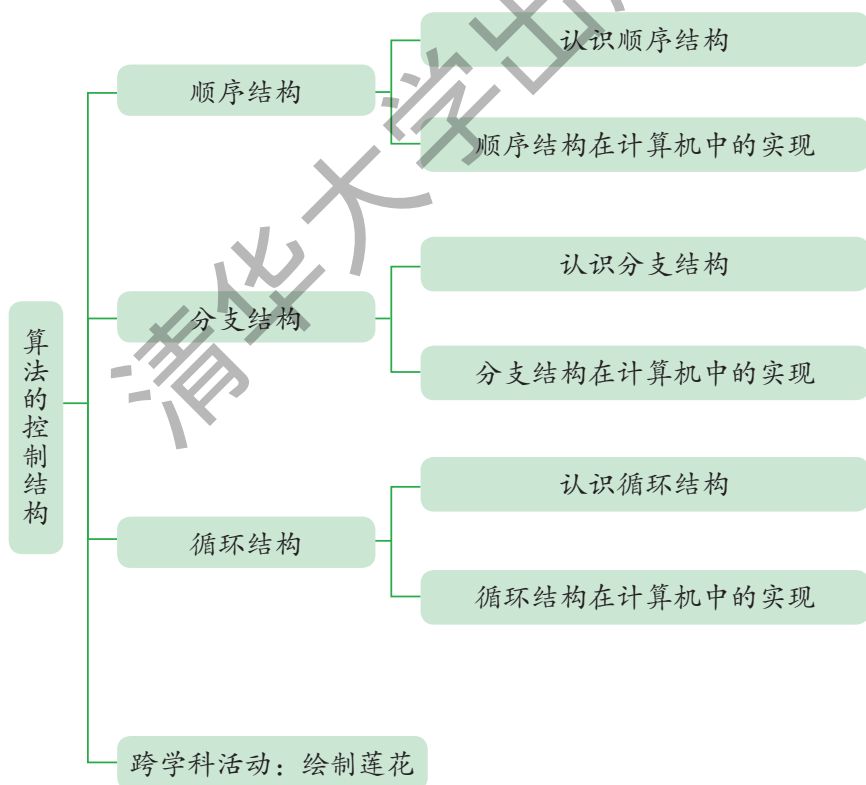
内容	遇到的问题	解决方法	感悟收获	总体评价
想一想				☆☆☆☆☆
做一做①				☆☆☆☆☆
做一做②				☆☆☆☆☆
议一议				☆☆☆☆☆

小结与评价

本单元你学习了：

- ◆ 学习算法的控制结构。
- ◆ 学会用顺序、分支、循环结构程序实现算法。
- ◆ 编写绘制莲花的程序。

知识梳理





学习检测

1. 算法的控制结构有哪些？

2. 用程序计算 10 以内整数的顺序累加和。

3. 用程序计算 100 以内偶数的循环累加和。



反思评价

在本单元学习过程中，肯定少不了与他人互动交流、参与作品制作等活动。现在请就此进行总结与反思，以更好地促进自身成长。

1. 从同伴那里学到过什么？

2. 向同伴分享过哪些观点？

3. 所运用的工具、方法是否得当？

4. 所设计的作品有哪些可以优化的地方？