

# 义务教育信息技术教学指南

## 身边的算法



五年级  
全一册



## 本模块设计说明

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。本模块以身边的算法为载体，使学生了解利用算法求解简单问题的基本方式，培养学生初步运用算法思维的习惯，并通过实践形成设计与分析简单算法的能力。

1. 本模块内容共 30 课，分为 8 个单元，每个单元安排 3 至 5 课。

2. 本模块有三个突出的特点。

（1）基于小学生的认知特点，对涉及数量的算法问题，通常先适当缩小问题规模，探讨清楚其中的算法规律后，再还原问题并进行求解，由浅入深、逐层深入地引导学生认识算法。

（2）适时安排“算法验证与实现”的相关内容，旨在帮助学生在相关知识学习之后能够操作实践。通过运行程序、分析程序、修改关键语句等方式，观察、验证算法的实现过程，真正感受算法的优势，体会编程解决问题的方法，了解信息科技课程的价值所在。

“算法验证与实现”大致可分为三个层面：一是简单的算法，学生可以运行自己编写的程序，获得成就感；二是具有简单核心概念和较多细节的编程算法，教师可以给出绝大部分的程序，让学生理解算法的概貌，填补一些要点；三是有趣但程序较复杂的算法，在教师提供程序的基础上，引导学生通过修改关键语句、设置不同参数，观察程序的运行，验证算法。

（3）关注分层教学，进行弹性的学习活动设计。在一些课中，安排了多个学习活动，教学时可以依据实际课时选择其中一至两个学习活动作为教学内容，其余的作为分层教学的参考。

3. 本模块所提供的图示、动态演示的程序等用于帮助教师直观呈现抽象内容；需要算法验证与程序实现的内容，要具备展示程序、运行程序、修改程序的条件；如果能保证每个学生都可以上机实践，教学效果会更好。

4. 教师可以依据本校学生的认知特点、信息科技硬件配置等情况，合理安排教学内容。

对于信息科技知识起点较低、课时较有限的学校，可以选取部分内容来重点教学，而把一些有难度的内容作为拓展，供学有余力的学生课后自主学习，通过课内课外、家校互通的多种实践活动方式，培养学生利用算法解决问题的能力，以及适应信息社会需要的数字素养和技能；对于学生有一定的知识基础，课时充足的学校，可以把其中一些学习内容、学习案例进一步拓展、延伸，也可以选用一些具有地方特色的案例进行教学，旨在引领学生进行更深入的实践探究。

## 数字素养与技能培养

五年级的学生能够进行一定的推理和计算，感知事物的准确性有所提高，抽象思维逐渐增强，逻辑思维快速发展，主动学习意识基本形成，具备了一定的实践探究能力。此阶段是提升数字素养与技能的关键期，通过本模块的学习，学生以下方面的信息科技核心素养会得到进一步增强。

### 信息意识

自然界和人类社会中的信息，通过一定的编码形成计算机能够处理的数据，算法描述了数据处理的基本过程。

- (1) 感受应用信息科技获取与处理信息的优势，初步认识信息处理的作用与价值。
- (2) 根据学习与生活需要，有意识地选用信息技术工具处理信息。在分析信息、数据和算法之间关系的过程中，培养信息意识。
- (3) 在问题求解过程中，有意识地寻求恰当的算法解决问题，尝试利用算法解决现实生活中的问题。
- (4) 输入同样的数据信息，利用不同的算法可能会输出不同的结果，获得不同的信息，以此了解算法的优势和局限。
- (5) 理解科学精神、原创精神，具有将创新理念融入自身学习、生活中的意识。

### 计算思维

五年级是计算思维培养的关键期，本模块从不同角度、不同层次得到体现。

- (1) 通过生活中的实例，了解算法的特征和效率，能用自然语言、流程图等方式描述算法。知道解决同一问题可能会有多种方法，认识到采用不同方法解决同一问题时可能存在时间效率上的差别。
- (2) 对于给定的任务，能将其分解为一系列的实施步骤，使用顺序、分支、循环三种基本控制结构简单描述实施过程，通过编程验证该过程。
- (3) 在问题解决过程中，能将问题分解为可处理的子问题，在逐步解决子问题后完成整个问题的解决。
- (4) 在一定的活动情境中，能够对简单的问题进行抽象、分解、建模，制订相应的解决方案。
- (5) 验证解决方案，反思问题解决的过程和方法，并对其进行优化。

### 数字化学习与创新

- (1) 通过学习身边的算法，体会算法的特征，有意识地将其应用于数字化学习过程中，适应在线学习环境。
- (2) 根据学习任务，合理选择数字设备、平台和资源，通过任务分解的方式提高学习效率。
- (3) 按照任务需求，运用信息科技获取、加工、管理学习资源，开展数字化探究和创新活动。

### 信息社会责任

- (1) 了解算法的优势及对知识产权保护的作用，认识到算法对解决生活和学习中的问题的重要性。
- (2) 认识到自主可控技术对保障网络安全和数据安全的重要性。
- (3) 在信息分享与交流活动中，能通过合理的方法辨别与抵制不良信息，尊重他人观点，理性发表个人观点。
- (4) 初步具备数据安全防护能力，能保护个人及他人隐私，自觉维护网络安全和数据安全。
- (5) 知道实际应用中的算法一般都存在某些局限，增强在信息社会中的责任担当和正确应对能力。

### 内容要求

- (1) 借助学习与生活中的实例，体验身边的算法，理解算法是通过明确的、可执行的操作步骤描述的问题求解方案，能用自然语言、流程图等方式描述算法。
- (2) 结合生活中的实例，了解算法的顺序、分支和循环三种基本控制结构，能分析简单算法的执行过程与结果。
- (3) 通过真实案例，知道算法步骤的执行次数与问题的规模有关，观察并体验采用不同算法解决同一问题时在时间效率上的差别。
- (4) 针对简单问题，尝试设计求解算法，并通过程序进行验证。
- (5) 以信息社会日常活动中蕴含的算法为例，讨论在线生活中算法的价值与局限(包括算法对知识产权保护的作用等)，及算法对生活的指导意义。

## 学业要求

1. 能用符号和编码表示问题求解所关心的对象，采用自然语言、流程图等方式，运用三种基本控制结构及其组合，正确进行问题求解的算法描述。
2. 基于给定的算法，能针对不同的输入数据规模，分别“数出”算法中某些步骤执行的次数。
3. 在此基础上，能进一步判断解决同一问题的不同算法在时间效率上的高低。
4. 能基于对算法的理解，设置和调整参数，观察相应程序的执行。
5. 基于对算法价值和局限性的认识，初步具有知识产权保护和应用安全意识。

## 单元主题与课标要求对应表

单元	单元主题	课名称	相关逻辑主线	核心内容	对应的课标内容要求
第一单元	无处不在的算法	第1课 生活处处有算法	算法、数据、信息安全	日常生活中的算法，算法的基本含义	1、5
		第2课 算法认识与体验	算法、数据、信息处理	算法的总体结构，进一步认识算法	1、2
		第3课 游戏体验寻规律	算法、信息处理	汉诺塔益智游戏的操作规律和算法	1、2
		第4课 算法应用在身边	算法、信息处理	算法在社会各领域的应用及其影响	1、5
第二单元	算法结构与描述	第5课 数学运算讲方法	算法、数据、信息处理	数学解题时的算法，算法是解决问题的方法描述，用流程图表示算法的顺序结构	1、2、5
		第6课 判断选择用分支	算法、数据、信息处理	用流程图表示的算法分支结构，生活中存在的分支结构	1、2、5
		第7课 重复操作用循环	算法、数据、人工智能	用流程图表示的算法循环结构	1、2、5
		第8课 算法验证与实现	算法、信息处理	编写简单程序验证算法，知道Python中顺序结构程序与简单分支结构程序	2、4
第三单元	用算法解决问题	第9课 互传密信有诀窍	算法、数据、信息安全	移位密码的加密算法，加密算法在保护信息安全方面的作用	1、5
		第10课 猜数游戏有捷径	算法、数据	从猜数游戏中知道二分法查找的对象、范围和判断条件，二分法用于解决实际问题	1、3、5
		第11课 闰年平年我知道（1）	算法、信息处理	算法的多分支结构，判断闰年的规则	1、2
		第12课 闰年平年我知道（2）	算法、信息处理	明确判断处理的条件以及输入、输出的内容，能够用程序验证与实现	1、2、3

续表

单元	单元主题	课名称	相关逻辑主线	核心内容	对应的课标内容要求
第四单元	发挥算法的优势	第13课 让计算机数数	算法、数据	用循环结构解决问题时的重复过程，认识循环结构和计算机工作特点的关系	1、2、3
		第14课 算法效率比一比	算法、数据	解决同一个问题可以有不同的算法，不同的算法具有不同的效率	1、2、3
		第15课 简单密码易破解	算法、数据、信息安全	估算破解密码所需的重复判断次数和时间，利用枚举法遍历数据	2、3、5
第五单元	把数据排列有序	第16课 比较交换找最值	算法、数据、信息处理	在一列数值中找出最大数，辨认其中的算法结构	1、2
		第17课 选择排序轻松做	算法、数据、信息处理	选择排序的基本步骤，执行次数与选择排序数据量的关系，算法验证与实现	1、2、3
		第18课 冒泡排序齐体验（1）	算法、数据、信息处理	冒泡排序的算法步骤	1、2
		第19课 冒泡排序齐体验（2）	算法、数据、信息处理	冒泡排序的过程，执行次数与冒泡排序数据量的关系，算法验证与实现	1、2、3
		第20课 化大为小桶排序	算法、数据、信息处理	桶排序的算法思想，桶的数量和范围在排序过程中的作用	1、2、5
第六单元	快速遍历数据	第21课 鸡兔同笼巧计算	算法	鸡兔同笼问题的不同求解方法，算法验证与实现	1、2
		第22课 兔子增长有规律（1）	算法	用列表法呈现兔子增长的数据变化，用算法表示一系列数据的递推规律	1、2
		第23课 兔子增长有规律（2）	算法	用流程图描述兔子增长的算法，算法验证与实现	1、2、3

续表

单元	单元主题	课名称	相关逻辑主线	核心内容	对应的课标内容要求
第七单元	了解更多的算法	第24课 多人过河巧安排	算法	规划算法的应用,把大问题分解成小问题解决	1、2
		第25课 有趣的七桥问题	算法、数据	抽取问题中的关键要素并进行简化来解决问题,实现一笔画的判断方法	1、2、3
		第26课 寻找最短的路径	算法、数据、人工智能	把全局问题分解成局部问题解决,寻找最小路径的算法描述	1、2、5
		第27课 网页排名有策略	算法、网络、信息安全	网页排名算法的作用,提升网页价值的意义,网络使用的规范及其存在的风险	1、2、5
第八单元	人工智能中的算法	第28课 认识决策树算法	算法、人工智能	人工智能应用中的决策树算法,决策树算法的应用场景	1、2、5
		第29课 智能工具再体验	算法、人工智能	生成式人工智能及其应用	1、4、5
		第30课 生命游戏有规则	算法、人工智能	生命游戏中的进化规则及其算法思想	1、2、4、5

# 目 录

## 第一单元 无处不在的算法

第 1 课	生活处处有算法	12
第 2 课	算法认识与体验	16
第 3 课	游戏体验寻规律	20
第 4 课	算法应用在身边	25

## 第二单元 算法结构与描述

第 5 课	数学运算讲方法	31
第 6 课	判断选择用分支	35
第 7 课	重复操作用循环	40
第 8 课	算法验证与实现	45

## 第三单元 用算法解决问题

第 9 课	互传密信有诀窍	54
第 10 课	猜数游戏有捷径	59
第 11 课	闰年平年我知道(1)	63
第 12 课	闰年平年我知道(2)	67

## 第四单元 发挥算法的优势

第 13 课	让计算机会数数	74
第 14 课	算法效率比一比	80
第 15 课	简单密码易破解	85

# 目 录

## 第五单元 把数据排列有序

第 16 课	比较交换找最值	92
第 17 课	选择排序轻松做	97
第 18 课	冒泡排序齐体验 (1)	102
第 19 课	冒泡排序齐体验 (2)	106
第 20 课	化大为小桶排序	111

## 第六单元 快速遍历数据

第 21 课	鸡兔同笼巧计算	118
第 22 课	兔子增长有规律 (1)	123
第 23 课	兔子增长有规律 (2)	127

## 第七单元 了解更多的算法

第 24 课	多人过河巧安排	133
第 25 课	有趣的七桥问题	137
第 26 课	寻找最短的路径	142
第 27 课	网页排名有策略	148

## 第八单元 人工智能中的算法

第 28 课	认识决策树算法	157
第 29 课	智能工具再体验	162
第 30 课	生命游戏有规则	166

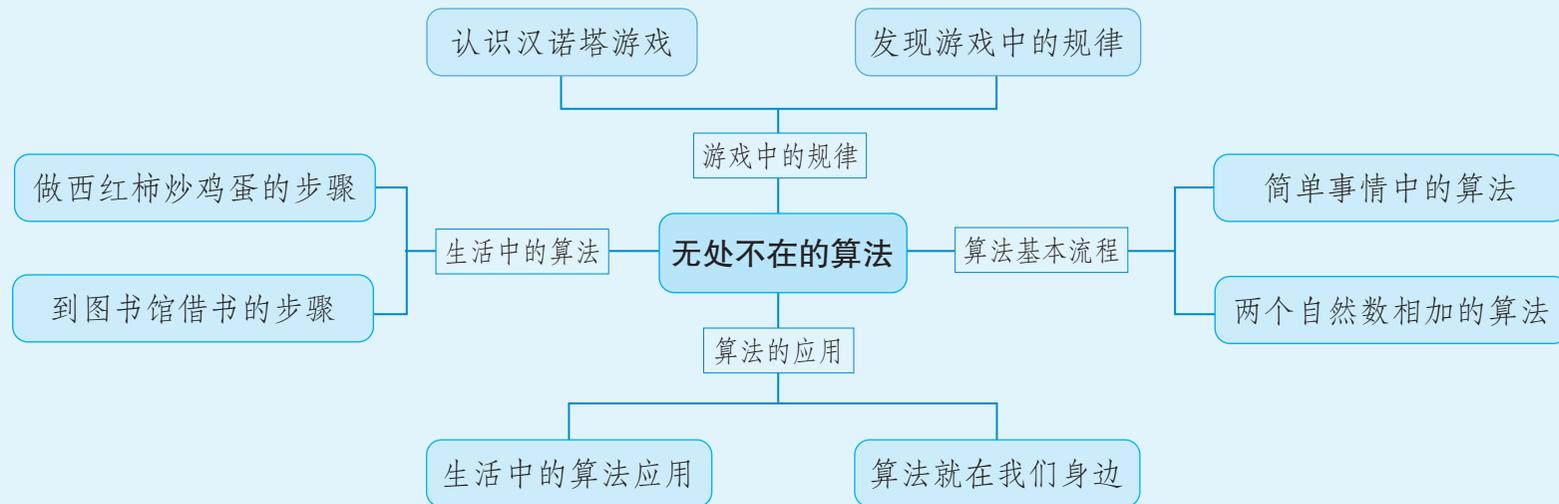
### 【引言】

本单元首先通过日常生活中的情境，引导学生认识算法的含义，再通过清晰、可执行的步骤描述问题的求解过程，进一步了解算法，明确“算法是通过明确的、可执行的操作步骤描述的问题求解方案”，知道算法是为了解决问题而设计的，因此算法与问题解决密不可分。

使用数字设备和工具完成任务或解决问题，本质上都是各种算法在发挥作用。

本单元的学习旨在让学生知道如何利用算法思想解决简单的问题，即将问题的解决过程分解为可以清晰表达出来的一系列步骤，这些步骤会让我们做事情更有条理、更有效率。从符合学生年龄和认知特点的身边问题出发，在他们所熟悉的场景和事件中，通过一个个学习活动或任务，自然而然地将他们带入到奇妙的算法世界之中。

### 【内容结构】



## 【教学实施条件】

1. 第1课、第2课、第4课可以在普通教室学习，配备教学演示与互动的设备、教师用的计算机等。
2. 第3课建议在信息科技实验室学习，计算机系统能够运行配套资源中的游戏、程序等。

## 【学习目标】

1. 通过分析日常生活中情境、实例中隐含的算法步骤，初步了解算法。
2. 体验用步骤描述的方法解决简单问题，从算法视角求解问题，知道算法是解决问题的策略，形成自主探究的意识。
3. 通过对社会生活中算法应用案例的分析，进一步体验身边的算法应用。
4. 从生活、学习等情境中发现规律，初步了解从特殊到一般的归纳过程，培养发现问题、解决问题的能力。

## 【评价建议】

下表可作为单元学习评价的参考。

评价内容	学生自评	教师评价
能够把典型的生活情境、过程描述为有序的步骤	☆☆☆☆☆	☆☆☆☆☆
知道算法的基本含义	☆☆☆☆☆	☆☆☆☆☆
认识算法的基本流程“输入—处理—输出”	☆☆☆☆☆	☆☆☆☆☆
能够从益智游戏体验中，知道其中的步骤、规律等	☆☆☆☆☆	☆☆☆☆☆
知道社会生活中各个方面都存在算法的应用	☆☆☆☆☆	☆☆☆☆☆
学习过程中，主动思考、积极探究，与同学合作交流	☆☆☆☆☆	☆☆☆☆☆

# 第 1 课

## 生活处处有算法

### 学习目标

1. 通过生活情境案例，了解算法是通过明确的、可执行的操作步骤描述的问题求解方案。
2. 通过分析日常做事的步骤，知道算法存在于日常生活之中。

算法在我们的日常生活中无处不在，从简单的日常做事到复杂的问题决策，都蕴含着算法的思想，算法改变着我们工作、学习和生活的方式。

### 学习活动 1

#### 做西红柿炒鸡蛋的步骤

有一天，小夏的爸爸妈妈因为去医院陪爷爷看病，不能回家做饭。小夏的姐姐依据菜谱做了西红柿炒鸡蛋，小夏觉得太好吃了，和妈妈做的味道几乎一样。



做西红柿炒鸡蛋的情境示意

想一想：为什么小夏姐姐做出的西红柿炒鸡蛋，能和他们的妈妈做出的味道几乎一样呢？

教学重难点：

1. 算法的基本含义；
2. 把问题解决的过程描述成有序的步骤。

首先以问题开启教学过程，例如，同学们会做西红柿炒鸡蛋这道菜吗？都有哪些做菜步骤呢？

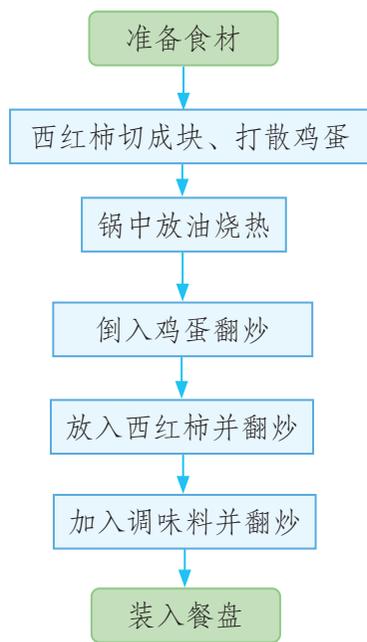
教师鼓励并耐心听取学生口头描述，可以用启发式教学，在引导学生思考发言的过程中，总结出用文字描述的步骤。

接着继续提问：这些步骤的顺序能改变吗？操作时哪些步骤需要反复观察？

然后引入到步骤及顺序的介绍。

以下是做西红柿炒鸡蛋的一份菜谱。

1. 准备食材：鸡蛋、西红柿、油、调味料。
2. 把西红柿切成块，在碗中把鸡蛋打散。
3. 在炒锅中加入适量油，开火烧热。
4. 把打散的鸡蛋倒入锅中，用铲子翻炒。
5. 把西红柿块放入锅中，继续翻炒。
6. 即将炒熟时，撒入适量调味料并翻炒。
7. 当西红柿炒熟时，装盘，准备享用。



做西红柿炒鸡蛋的步骤

从学生熟悉的依据菜谱做菜这件事，自然地引入算法思想。这里可能学生不能完全理解，但能够知道“算法是明确的、有步骤的”就可以，后续还会进一步深入学习和感悟。

这里引领学生从步骤图出发，认识用图框表示步骤的方法，先不要引入流程图的概念。

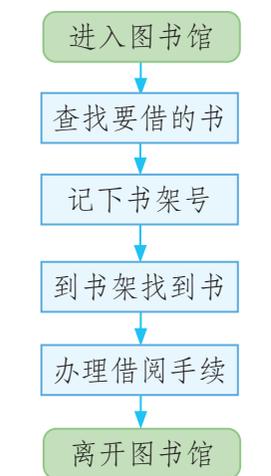
引导学生观察、分析、发现菜谱中存在的操作顺序、关键步骤。可以适当启发学生总结依据菜谱做菜有哪些好处。

## 学习活动2

### 到图书馆借书的步骤

下面是到学校图书馆借书的一般步骤。

1. 进入图书馆。
2. 在图书管理系统中查找要借的图书信息。
3. 记录下该图书所在的书架编号。
4. 根据编号到相应书架，找到并取出图书。
5. 办理借阅手续。
6. 带着图书离开图书馆。



到图书馆借书的步骤

实际上，在图书管理系统中查找图书的时候，又包含下面的步骤。

1. 登录图书管理系统。
2. 进入查询界面。
3. 输入查询的内容，可以是书名、作者姓名等信息。
4. 查看搜索结果，了解馆藏信息和图书所在位置。

那么，查找到要借阅的图书后，办理借阅手续又有哪些步骤呢？

1. 扫描借阅者的借阅卡。
2. 在管理系统中查看借阅者的借阅数量。
3. 如果没有超过借书的数量上限，就扫描图书的编码，登记借阅信息。
4. 显示借阅成功，完成借阅手续。

通过以上分析发现，简单的图书借阅也存在着相应的步骤和方法，正是因为有这些规定的步骤，才能保证图书馆的工作有条不紊，让每个人能很快借到自己想要的书。

到图书馆借书也是学生熟悉的情境，学生可能到学校图书馆借过书，也可能在家长的陪同下到过更大图书馆借书。有了这些经历，这里只需强调主要的步骤和流程，不必考虑不同规模图书馆的借阅规则。

不同图书馆的管理和借阅系统往往包含不同的步骤，但基本的流程是一致的。在梳理学生到图书馆完成借书任务的过程中，教师可以带着学生进一步分析每一步的具体操作，让学生体会身边的算法隐藏在日常生活的熟悉场景之中。

现在很多书店和图书馆都有自助查询系统，有的学生接触过。也会遇到想借的书已被借出、借阅时超过借阅册数上限、还书时提示超期等情况，这其中都蕴含着算法。可以通过提问引导学生思考，鼓励他们细化内容用语言描述出来，既能够引导学生将一个大问题拆解成多个小问题，也能将思考的过程外化出来。

例如，在完成借书任务过程中，图书管理系统在背后替我们做了哪些事情？要做到这些，需要什么样的算法？

如果学生没有以上借书经历，教师可以用图片展示、视频演示等方式，引导学生思考。例如，在借书过程中，图书管理员做了哪些处理操作？

### 学习活动3

#### 认识算法的基本含义

以上我们感受到了算法存在于日常生活与学习之中。

那么，什么是算法呢？

算法是通过明确的、可执行的操作步骤描述的问题求解方案。

换句话说，算法是将解决一个问题或做某件事情的过程描述为一些明确的、可操作的步骤。

进一步分析发现，前面依据菜谱做西红柿炒鸡蛋的过程中，菜谱对做菜步骤的描述是有顺序、可操作的，但同样的菜谱、同样的原材料，不同的人做这道菜可能会出现不同的口味。因此，操作步骤还可以进一步明确和细化。

思考与探究：如果要用算法控制机器人制作西红柿炒鸡蛋，应该进一步明确哪些步骤或操作？

提示：为了使算法更严谨、具有自动可执行性，需要细化西红柿、鸡蛋、油和调味料的量，明确操作时间等。

#### 拓展与提升

在日常生活中，还有哪些场景存在着算法？对我们的学习与生活产生了哪些作用？

教师还可以选择生活中的其他事情，帮助学生了解其中存在的算法，注意应把握明确有序的原则。

了解算法的基本含义后，为了加深认识，并让学生知道生活中处处有算法，可以适当进行总结、提炼。同时可以简要地说明后续课程要继续学习的相关内容，让学生知道理解算法需要逐步深入。

可以给出适当修改后的菜谱，引导学生继续补充修改。

在日常生活中，我们习以为常的许多事情和任务中都有步骤，通过将做事或完成的过程描述为一系列有序的步骤并遵守，可以更好更快地完成任务。建议将学生能想到的算法步骤呈现出来，选择较典型的例子进行交流分享。

## 第2课 算法认识与体验

### 学习目标

1. 通过分析并描述现实生活中的情境和案例，进一步认识算法的含义。
2. 能够用自然语言描述做事的步骤，知道算法的基本流程，初步培养按步骤解决问题的能力。

在上一课的学习中，我们知道很多事情中都包含着算法。那么，算法具有什么样的基本流程呢？

### 学习活动1

#### 简单事情中的算法基本流程

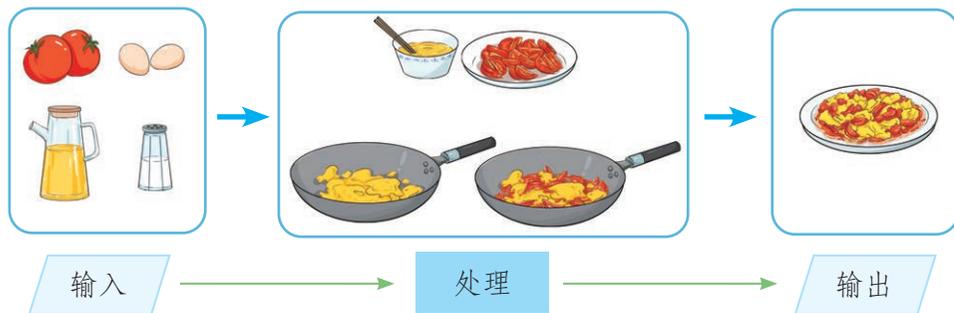
观察和分析前面做西红柿炒鸡蛋的过程，可以发现，要根据菜谱做出一道特定的菜品，需要经过三个阶段。

第一，准备原材料。这些原材料相当于“输入”。

第二，遵循菜谱炒菜。这个过程相当于“处理”。

第三，完成处理步骤后获得一盘菜肴。这个最终结果相当于“输出”。

这个过程可以总结为下面的示意图。



做西红柿炒鸡蛋的过程

教学重难点：

1. 算法的基本流程；
2. 用自然语言描述做事的步骤。

本课的教学案例进行了一定程度的分层设计，教学时可以依据课时与学生知识基础，灵活选择，从而更贴近教学实际。

回顾上一课做菜的过程，引导学生观察、发现菜谱中存在的步骤与顺序，再按照算法的基本流程分析其中的各个阶段，把步骤与算法基本流程建立关联。

在这个学习活动中，准备鸡蛋、西红柿、油和调味料等原材料是输入；切西红柿、搅拌鸡蛋、倒入锅里、翻炒、加入调味料等操作是处理；最后做好的“西红柿炒鸡蛋”这道菜就是输出。

教学时，突出算法的基本流程是本课的核心内容。

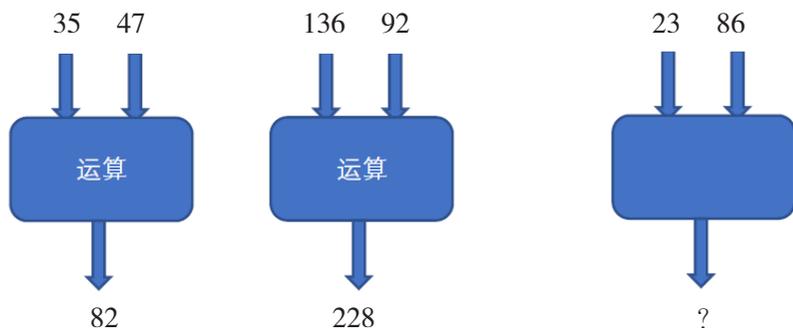
在实际算法设计中，允许零输入的存在，即没有输入部分，尤其是在自动控制的过程中，往往存在自动获取数据的情况。

通常，做不同的事情往往有不同的方法，不同人做同一件事情也会有所不同，但做事情的基本过程有很多共同之处，这些共同之处就是算法的思想所在。算法的基本流程可以表示为输入、处理和输出三个基本部分。

## 学习活动2

### 两个自然数相加的算法分析

数学课的学习中有一个闯关活动：观察下图，发现规律。



闯关活动法中的运算

下面把两个自然数相加的计算步骤描述如下。

第1步：把两个自然数按个位上下对齐排列。

第2步：从个位开始。

第3步：把数位上的两个数字相加。

(1) 如果相加的和小于10，把这个和直接写在数位的下面。

(2) 如果相加的和大于或等于10，把这个和的十位数字1进位到前一位上，个位数字写在数位下面。

第4步：向左移动一位，对后续数位进行类似第3步的操作，如果有进位要把进位数也加上，直到处理完成最左侧的数位。

第5步：计算完成后，最下面一行的数就是两个自然数的和。

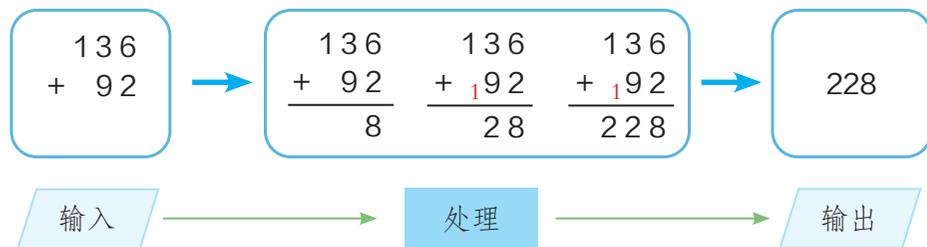
继续以学生熟悉而且简单的“两个自然数相加”为例，介绍算法的含义和基本流程，进一步加深认识。

引导学生观察并提问：闯关活动的结果是怎么得出的？你发现了什么规律？学生会发现，将两个数相加得到结果，那么无论给出什么样的数值，按照这个运算法则，都能获得结果。

关键点：中间的运算过程就是算法基本流程中的“处理”。

注意：数位上的两个数字相加时，如果该位置上有进位，相加结果要加上1。这是容易被学生忽略的地方，教学时适当进行提示。

与前面做西红柿炒鸡蛋的过程一样，两个自然数相加的过程也遵循了算法的基本流程。可以用下图表示这个过程。



两个自然数相加的基本流程

进一步探究：除了两个自然数的加法外，减法、乘法和除法等运算中的计算过程，是否也要遵循一些步骤？请将这些步骤写出来。

在信息科技领域，算法用于计算机解决问题或处理事情的时候，对算法步骤的界定和限制更加严格，描述算法的语言必须严谨、明确，并且能准确地转换为计算机可实际做到的形式。

我们解决问题时，通常遵循“输入—处理—输出”的基本流程。

在解决加法的算法问题时，由于学生已有相关的数学基础，此处不必强调计算过程中的算理，而要强调算法的基本流程，在已知算理的指导下，通过计算步骤认识基本流程。

教师还可以选择学生已经学过的公式等计算问题，重点突出解决问题的步骤。比如求平均数、不同分母分数的加法等，都能体现算法的步骤。

这些步骤的描述看起来有点复杂，但方法是学生熟悉的，只是学生没有注意到这些步骤的存在，以至于在实际做加法时感觉不到自己在遵循这些步骤进行计算。

在信息科技领域，算法指的是利用计算机或其他数字设备，根据要完成的任务或待解决问题的条件和要求所实施的一系列步骤。而算法的思想及方法并不局限于特定设备或工具，而是一种数字化的思维方式（计算思维），在完成任何任务及解决任何问题时都能发挥不可替代的作用。

“身边的算法”主要指在解决与学生学习生活相关的问题时所涉及的思想和方法。

## 拓展与提升

24 点游戏活动。

24 点游戏是一种益智游戏，要求在 1 至 10 之间任意抽取 4 个数，通过加、减、乘、除运算将 4 个数计算后得出 24。

要求：抽取的每个数必须用而且只能用一次。例如，抽取的数是 2、4、6、9，那么算式为  $(9 - 6) \times 4 \times 2 = 24$ 。

尝试玩一玩这个游戏，写出计算步骤，体会其中存在的算法。

教师可以提供几组数，引导学生以小组竞赛的形式进行计算，教学关注点是记录步骤。在调动学生积极性的同时，引导他们思考下面的问题。

1. 是否有不同的解法？
2. 同样的数，如果改变计算顺序，对计算结果有什么影响？

在这个游戏中，乘法很重要，24 是 30 以下（不含 30）公因数最多的自然数。常见的计算方法是利用  $3 \times 8$ 、 $4 \times 6$  和  $2 \times 12$ （需再分解），所以应优先考虑这三种组合。一般情况已有其中的一个数，可用其他 3 个数去凑另一个数。

## 第3课 游戏体验寻规律

### 学习目标

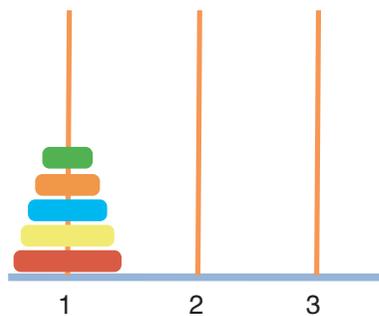
1. 通过体验汉诺塔益智类游戏，了解其中存在的操作规律。
2. 进一步认识算法是通过明确的、可执行的操作步骤描述的问题求解方案。

汉诺塔是一个益智游戏，在游戏探究过程中有助于培养逻辑思维能力。

### 学习活动1

#### 探究汉诺塔游戏规律

汉诺塔游戏有三根柱子，在其中一根柱子上有一些大小不同的圆环。要求把圆环从起始位置移动到目标位置。



汉诺塔

规则如下。

1. 每次只能移动一个圆环。
2. 在移动过程中，可以借助任意一个有圆环或者没有圆环的柱位，但要保持小圆环始终在大圆环的上面。

教学重难点：

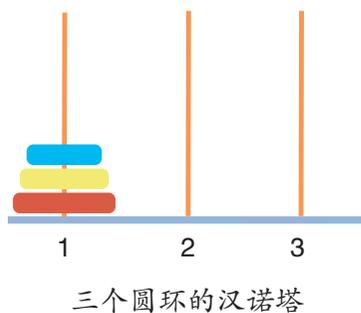
1. 在游戏过程中体会其中存在的操作规律；
2. 把问题解决的过程用图示或文字描述为有序步骤。

汉诺塔是学生见过或者玩过的益智游戏，能够激发学生兴趣，如果用实物玩具在课堂上展示，效果更直观。如果没有实物玩具，则可以通过语言描述、图片展示或者播放视频，向学生说明规则。

当然，益智类游戏有很多，教学时也可以根据学生实际情况选择使用。

培养学生的计算思维是本模块的重点内容。其中，帮助学生有条理、有层次地分析问题，是培养计算思维的基础。

活动要求：柱 1 上有由三个圆环组成的汉诺塔，请把圆环从柱 1 移动到柱 3，描述移动的过程。



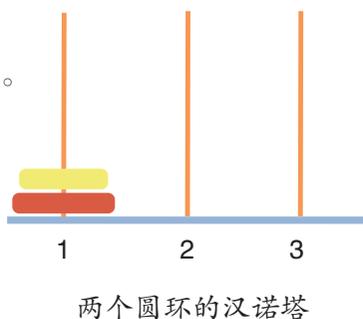
开始移动前，先思考以下关键问题。

问题 1：初始状态和完成状态分别是什么？

问题 2：解题的关键是什么？

为了简化问题，先分析只有两个圆环的移动情况，基本步骤如下。

1. 小的圆环在最上面，先把它移动到柱 2 上。
2. 然后把大的圆环移动到柱 3 上。
3. 再把小圆环移动到柱 3 上。



三步就可以完成，操作非常简单。

接下来，完成移动三个圆环的任务。

我们可以忽略最大的圆环，先将其他两个圆环移动到柱 2 上，方法与前面移动两个圆环一样，只不过第一步先移动小圆环到柱 3 上。

参照实物或图示，帮助学生认识游戏的操作规律，以便保证后续活动的顺利进行。

开始移动前，先让学生思考两个关键问题，进一步明确任务要求和活动规则。

当学生能回答“问题 1”和“问题 2”时，就理清了解决问题的思路。

接下来，可以由学生将操作过程细化，写出移动的步骤。

可以用图示、表格、文字等不同方式来描述，目的是引导学生外化自己的思维过程，将步骤清晰地表达出来。

先将问题简化，分析两个圆环的情况，降低学生认知的负担。

还原问题。

移动过程的描述，建议在图示的基础上引导学生用文字描述。

移动三个圆环的步骤描述如下。

第 1 步：把小圆环从柱 1 移动到柱 3。

第 2 步：把中圆环从柱 1 移动到柱 2。

第 3 步：把小圆环从柱 3 移动到柱 2。

第 4 步：把大圆环从柱 1 移动到柱 3。

第 5 步：把小圆环从柱 2 移动到柱 1。

第 6 步：把中圆环从柱 2 移动到柱 3。

第 7 步：把小圆环从柱 1 移动到柱 3。

此外，还可以采用表格的方式来描述移动的过程。

用表格描述移动三个圆环的过程

次数	小环	中环	大环
1	柱 1 → 柱 3		
2		柱 1 → 柱 2	
3	柱 3 → 柱 2		
4			柱 1 → 柱 3
5	柱 2 → 柱 1		
6		柱 2 → 柱 3	
7	柱 1 → 柱 3		

进一步探究：

尝试移动四个圆环或更多圆环，体会其中存在的规律。

提示学生分析：哪些步骤是相同的或类似的？哪些是需要反复操作的？这些步骤有什么规律？

可能学生不能正确回答，但通过游戏过程中的观察与思考，培养学生观察、分析问题的能力和思维习惯。

移动四个圆环有一定难度，教师提出思考方向，让学生进一步思考、探索。

提示：有四个圆环时，忽略最大的一个圆环，用移动三个圆环的方法，把它们移动到柱 2 上，然后把最大的圆环移动到柱 3 上，再把柱 2 上的三个圆环移动到柱 3 上。

规律提示如下。

圆环为奇数时，第一步将最小圆环移动到柱 3 所用步骤最少。

圆环为偶数时，第一步将最小圆环移动到柱 2 所用步骤最少。

实际上，只要学会移动两个或三个圆环，即使再增加圆环个数，操作方法都是先把最大圆环之上的所有圆环先移动到柱 2 或柱 3，然后重复操作，逐个把圆环移动到目标位置。也就是不管需要移动多少圆环，这个游戏最终都能达成目标。例如，两位学生分别用了 7 步和 11 步完成了 3 个圆环的移动，体现了不同的移动方法和效率。

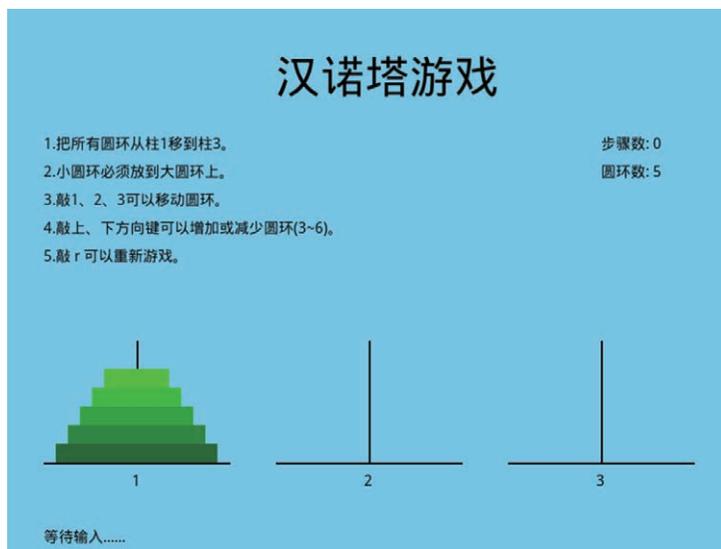
这个游戏中包含了一种算法——递归算法。教学时不必要求学生深入了解，只要能发现其中的操作规律即可。

## 学习活动2

### 试玩汉诺塔程序游戏

一位信息科技老师编写了一个汉诺塔游戏的程序，在配套资源中找到这个程序，试着玩一玩，看看谁能移动四个以上的圆环？

活动要求：选择不同数量的圆环来体验这个游戏，小组同学比一比移动圆环的数量和所用的时间。



游戏程序界面

进一步思考：

这个游戏是否也遵循了算法的基本流程？其中的输入、处理和输出分别是什么？

这个专为学生学习编写的小游戏程序，学生通常比较喜欢。可以引导学生尝试移动多个圆环，进一步体验其中存在的操作规律。运行程序时，要切换到英文输入状态。

以下是给教师的一些提示，不必要求学生了解。

圆环比较多时，分析移动细节会很困难。操作时可以发现：

如果要把  $n$  号圆环从柱 1 移动到柱 3 上，那么 1 至  $n-1$  号圆环肯定在柱 2 上；

要把  $n-1$  号圆环从柱 1 移动到柱 2 上，那么 1 至  $n-2$  号圆环肯定在柱 3 上；

要把  $n-2$  号圆环从柱 1 移动到柱 3 上，那么 1 至  $n-3$  号圆环肯定在柱 2 上……

因此，移动过程的描述可以分为三步。

第 1 步：把 1 至  $n-1$  号圆环移动到过渡柱上。

第 2 步：把  $n$  号圆环移动到目标柱上。

第 3 步：把 1 至  $n-1$  号圆环移动到目标柱上。

使用递归算法时，只需考虑“满足什么条件后停止递归调用”和“每次递归调用所需设置的条件”。例如，在这个程序中，当变量  $n$  等于 1 时，停止递归调用。

运行这个程序，先调用的函数不一定会先执行，计算机层层深入直到满足相关条件后才停止递归调用。

提示：输入是选择开始位置的圆环数，处理是移动圆环的操作，输出是移动到目标位置的圆环。

## 拓展与提升

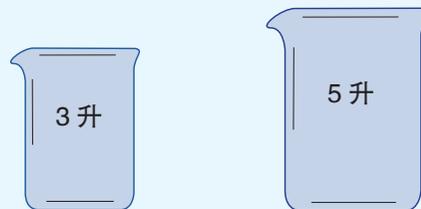
求解量水问题。

现在有两个量杯，一个容积是5升，一个容积是3升，杯壁上没有刻度线，如何准确地量出4升水？

要求：参照本课汉诺塔游戏的描述过程，补充下表呈现的操作步骤，体会用操作步骤描述问题解决的过程。可以使用第三个没有刻度的容器。

步骤	操作	5升量杯状态	3升量杯状态
1	将5升量杯装满水	满5升水	空
2		2升水	3升水
3	将3升量杯中的水倒入其他容器		
4	将5升量杯中剩下的2升水倒入3升量杯		
5		满5升水	2升水
6	将5升量杯中的水倒入3升量杯直至装满		

这个案例也是益智类学习活动，根据教学需要和学生的学习能力，这个案例可以作为教学替换案例使用，也可以作为课后学生自主拓展的案例。



3升和5升量杯示意图

步骤	操作	5升量杯状态	3升量杯状态
1	将5升量杯装满水	满5升水	空
2	将5升量杯中的水倒入3升量杯	2升水	3升水
3	将3升量杯中的水倒入其他容器	2升水	空
4	将5升量杯中剩下的2升水倒入3升量杯	空	2升水
5	将5升量杯装满水	满5升水	2升水
6	将5升量杯中的水倒入3升量杯直至装满	4升水	3升水

## 第4课

### 算法应用在身边

#### 学习目标

1. 通过对社会生活中算法应用案例的分析，了解身边的算法应用。
2. 进一步巩固对算法的认识，知道算法在社会各领域扮演着重要的角色，发挥着不可替代的作用。

在社会生活中，到处都存在算法的应用，算法应用促进了社会进步。例如，下面的应用中都有算法的身影。

- 智能设备解锁、人脸识别等有识别算法
- 短视频网站、购物网站有推荐算法
- 搜索引擎中有搜索算法、网页排名算法
- 导航软件有路径规划算法
- 相机中有图像处理算法

通过了解和认识算法的应用，可以更好地应对数字时代的挑战，为未来更好地学习、生活和工作做准备。

#### 学习活动1

##### 算法在生活中的应用

###### ► 搜索引擎

互联网是我们获取信息的重要渠道。在网上搜索信息时，搜索引擎是我们的好帮手。

教学重难点：

1. 社会生活中算法应用的主要领域；
2. 身边的算法应用实例。

本课安排了两个学习活动，使教学有更多的选择，并尽可能地提供了算法应用的多个领域、场景。实际教学时，可以依据课时与学生认知灵活选择，重点列举介绍其中的几个方面即可，不必追求面面俱到。比如，展开介绍学习活动1，而把学习活动2作为学生自主学习的内容。

当然，教学时也可以根据实际情况，选择其他更符合学生认知的场景，目的是关注算法在社会生活中所起到的作用，让学生知道算法存在于社会的各个领域。

组织教学时，教师可以针对相关领域依次进行介绍。或者将学生分组，每组负责一个应用领域，由教师设计任务，引领学生阅读、查找资料并汇报交流。学习任务举例如下。

1. 你有某领域算法应用的经历吗？与同学交流自己的想法。
2. 查找某个应用背后存在的算法，总结整理成文档。
3. 以一种算法应用为例，结合实例，在班级内分享。

介绍这些应用时，不要局限于应用本身，而是要注意强调本课的重点——算法应用。同时注重立足核心素养，培养信息意识，引导学生把所学知识与生活实际建立联系。

用搜索引擎搜索信息时，算法可以帮助我们找到相关的网页，分析网页的关键词、内容质量和链接的重要性等，把最相关的网页呈现出来，提高我们搜索信息的效率。

### ▶ 交通出行

在交通出行中，我们可以利用数字地图中的导航功能进行导航，算法帮助导航系统分析交通情况、距离和预计时间，给出最快最优的到达方案。而且算法会根据交通拥堵状况实时提醒驾驶者避开拥堵的道路，从而节省时间和降低能源消耗。我们还可以通过数字地图查询去某地的路线，除了查询目的地，还能查询步行最少、地铁优先、高速优先等信息。此外，自动驾驶汽车也使用算法来感知环境、做出决策并控制车辆运行，实现安全和高效地行驶，提升驾驶与乘坐的舒适性。

算法有助于提高交通效率和安全性，改善人们的出行体验。

### ▶ 在线购物

我们在线购物时，网站设计者利用算法帮助人们更方便地购物。这些算法会根据个人购买历史、浏览记录和评价等，进行个性化的产品推荐服务，让人们更容易找到想要的商品。算法还可以计算最优价格，帮助配置最佳的优惠方案。同时，算法还能帮助网站管理商品库存，确保购买的商品有货并能提醒商家及时进货。

学生在之前的学习中使用过搜索引擎，能够通过输入关键词来搜索、查询信息。为此，教师可以从学生的应用经验出发，以知识介绍形式帮助学生了解搜索引擎中关键词匹配、页面呈现等操作中包含的算法，知道这些算法让我们搜索信息时更快更准确。

关于交通出行中的算法应用，学生在生活中基本都有体验，教学时利用学生的这些已有经历展开介绍，从而让学生对算法就在我们身边形成一定的共识。此外，交通信号灯控制中也有算法的支持，许多城市使用智能信号灯来自动调整红绿灯的时长，以减少拥堵和提高交通效率。在共享出行服务中，常常会使用算法来根据需求量、用户位置等自动匹配车辆、司机，以提供更便捷的服务。

在线购物是学生容易理解的内容，教师可以结合学生的实际经历，介绍算法在其中所起的作用，如推荐算法、搜索算法、价格分析算法、库存管理算法等。

## 学习活动2

### 算法在一些领域的应用

#### 图像识别

在图像识别中，算法帮助人们认识和理解不同的图像。图像识别技术利用算法识别人脸、物体、图像的形状、颜色等特征，并将它们与特征库中的信息进行比较，从而获得判断结果。

图像识别技术已经广泛应用到社会各个角落。例如，可以刷脸乘坐地铁、公交、高铁等，还可以刷脸或刷指纹进入小区或办公楼，这些算法应用增强了社会的安全性。

#### 医疗诊断

在医疗诊断领域，算法发挥着越来越重要的作用。诊断系统利用算法来辅助医生们诊断疾病和进行图像分析，协助做出更准确的诊断，提供更有效的治疗方案，使人们的健康得到更好的关注和照顾。例如，医学影像诊断使用图像分析算法来检测异常和病变部位。此外，药物研发中的计算机模拟和数据分析也依赖算法，算法让药物研发和筛选更加快速有效。

#### 艺术创作

艺术创作也在借助算法的力量。例如，数字艺术相关软件使用算法来生成各种艺术效果。这些算法能够分析图像的特征，根据不同的艺术风格生成独特的效果。算法为艺术创作提供了更多可能性，大大降低了准入门槛，让每个人都能在创作艺术作品中发现乐趣和惊喜。

图像识别技术作为人工智能领域的一个重要分支，已经展现了广泛的应用价值。对学生来说，生活中使用图像识别的场景较多。教学时注意启发学生从不同场景进行观察、分析和思考。

例如，可以结合学校的刷脸进校园、停车场的车牌识别与计费、超市的商品识别与分类、通过拍照识别植物种类等应用来展开。

教学时应注意强调基于算法的信息科技在医疗诊断中的应用。例如，人工智能相关算法可以分析患者的医学数据，提供辅助医生进行疾病诊断的决策支持。医学影像是医生进行诊断的重要依据，算法能够对医学影像进行自动化分析，并根据患者的个体特征和病情数据，提供个性化的参考方案。

如果条件允许，可以引导学生尝试利用相关软件进行艺术作品的创作，然后结合软件能够实现的效果，展开介绍相关的算法。

## ▶ 语言翻译

语言翻译工具使用算法来帮助人们理解不同的语言。翻译工具利用算法先分析输入的文字或语音，再翻译成另一种语言。这些算法会不断“学习”不同语言的规则和词汇，从而更准确地进行翻译，让人们更方便地与使用不同语言的朋友交流，了解他们的想法和文化。

## ▶ 体育运动

在人们的日常运动中，智能手表、手机等移动设备都能安装运动类的应用软件，这些软件使用传感器获得运动数据，并通过算法分析这些数据，帮助人们监测运动情况，如运动时间和运动强度等，提升运动质量的同时享受其中的乐趣。

此外，运动员利用追踪设备记录运动数据，通过运动学和动力学等算法分析数据，进而监测训练进展，改善运动技能。

## 拓展与提升

1. 试着从算法应用的角度，观察自己身边存在的算法应用实例，思考算法在其中所起的作用。

2. 目前，人脸识别技术已经在很多场合广泛应用，你能否描述人脸识别过程中的关键步骤？想一想算法在这一过程中的主要作用。

教学时，可以给学生提供不同的语言翻译系统，引导他们翻译自己的作文，或者信息科技、语文、数学等学科的学习内容。

体育运动在学生生活中必不可少，其中有很多场景都能体现算法的应用。例如，计算篮球联赛场次，篮球联赛进行双循环比赛，所有参赛队伍互相轮赛两次，给出参赛队伍，利用算法安排场次。又如，运用人工智能技术，学生在跳绳、跳高、跳远、仰卧起坐等运动项目中，完成运动数据的识别与记录等。

有关算法应用的例子很多，教学时可以根据本地实际和学生的认识水平，有选择、分层次地展开学习。

1. 可以引导学生结合自己的生活经历思考身边的哪些实例应用了算法。这样既能帮助学生回顾已学知识，又能帮助他们联系生活实际、拓宽思路。

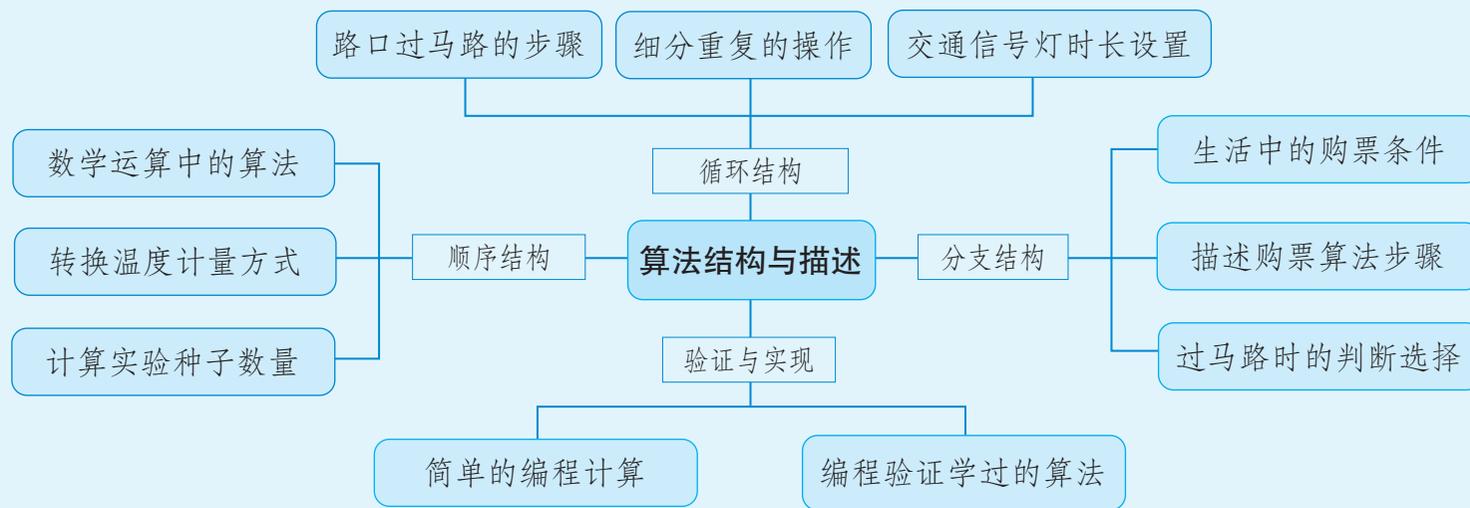
2. 人脸识别算法是人脸检测和识别系统的基础。在人脸识别过程中，算法的基本流程是：获取图像、视频中的人脸数据（输入）；计算、抽取人脸的数据模型（处理）；将获得的数据模型与特征库中的数据模型进行比较，从而识别或验证身份（输出）。

### 【引言】

在初步了解算法的基本含义及其应用之后，本单元的主要任务是认识、学习算法的三种基本控制结构：顺序结构、循环结构和分支结构。利用这三种基本结构可以组成各种不同的算法。算法是解决问题的方法和步骤，通常可以用自然语言、流程图等进行描述。自然语言灵活性比较高，描述起来比较简单，但有时会显得不够严谨。流程图用特定的图框和流程线来表达操作步骤和先后顺序，让思维过程具体化，更加直观、易于理解，贴近程序实现的方法。算法描述时，首先用自然语言对问题的核心步骤进行描述，再用流程图对算法做更严谨的描述。恰当的算法描述便于认识、分享和实现问题解决的思路。

在问题解决过程中，带领学生认识流程图中每种图框的含义，初步认识三种基本控制结构的流程图表示方式，为后续学习打好基础。

### 【内容体系】



## 【教学实施条件】

1. 本单元建议在具备运行 Python 环境的信息科技实验室里教学，能够运行配套资源中的游戏、程序等。
2. 可以制作一些矩形框、圆角矩形框、菱形框等实物，根据需要填写内容、调整位置，补充流程线，从而构成流程图。

## 【学习目标】

1. 通过生活中的实例，知道算法是为了解决某个特定问题所确定的一系列步骤，并能用自然语言、流程图描述。
2. 认识流程图中每种图框的含义和表示，知道顺序结构、分支结构和循环结构的描述方法。
3. 针对特定问题，能按照算法的基本流程（输入—处理—输出）分析问题并解决问题。
4. 能利用程序验证算法，尝试编写简单的程序实现算法。
5. 知道算法步骤之间的逻辑关系，了解算法的执行路径。

## 【评价建议】

下表可作为单元学习评价的参考。

评价内容	学生自评	教师评价
明确算法的基本流程，知道并能描述顺序结构	☆☆☆☆☆	☆☆☆☆☆
知道并能描述存在的分支结构及其判断条件	☆☆☆☆☆	☆☆☆☆☆
知道并能描述存在的循环结构及其循环的终止条件	☆☆☆☆☆	☆☆☆☆☆
能读懂算法的流程图，了解算法的执行路径	☆☆☆☆☆	☆☆☆☆☆
能利用程序验证算法，尝试编写简单的程序实现算法	☆☆☆☆☆	☆☆☆☆☆
能积极主动地和同学讨论问题，互相帮助	☆☆☆☆☆	☆☆☆☆☆

## 第 5 课

### 数学运算讲方法

#### 学习目标

1. 通过对数学运算中求解方法的分析，了解算法在数学解题中的作用，认识算法的顺序结构。
2. 从算法视角看待计算问题，知道算法是解决问题的方法描述。

上一个单元中，学习了算法的概念以及算法的应用。这一课通过分析、求解数学运算问题，了解其中包含的算法。

#### 学习活动 1

##### 温度计量方式的转换

作为校园记者站的一员，老师请你写一篇关于友好学校情况的介绍。你在网上查询到该友好学校所在地 2 月的平均气温是 50 华氏度。为了方便同学们了解情况，需要把华氏温度转换成摄氏温度。

华氏温度与摄氏温度可以通过以下公式进行计量方式转换。

$$\text{摄氏温度} = 5 \times (\text{华氏温度} - 32) \div 9$$

公式建立了摄氏温度与华氏温度之间的转换规则。即：

给定一个华氏温度的值（输入），通过公式可以计算（处理）得到相应摄氏温度的值（输出），这个转换公式就是一种比较简单的算法。

教学重难点：

1. 用自然语言描述数学运算中的算法；
2. 把简单计算题的解题过程，用明确有序的步骤表示。

从知识基础上来看，学生已经掌握了相关数学知识，学习过举例验证的方法，这为自主探究提供了基础。

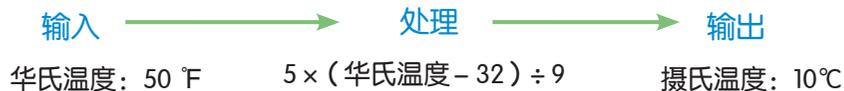
教学时可以联系实际，通过较直观的方式展示教学内容，帮助学生进行推理、验证结论，感受数学运算与算法的联系。

这里用较简单的温度转换公式来分析问题，有助于学生的接受和理解，逐步引领他们用算法思维来思考问题，进一步理解算法是解决问题的有序步骤和方法。

虽然是一个简单的问题，同样可以设计问题链，这也是为学生思考问题提供方向。学生在面对新问题时，可以模仿这样的问题分析方式，从而找到立足点，形成一定的迁移能力。

可以先引导学生分析以下问题。

1. 所求是什么？——摄氏温度。
2. 已知条件是什么？——华氏温度。
3. 怎么求解？——通过已知的公式进行转换。



温度计量方式转换的基本流程

以下是分步骤求解的过程。

第 1 步: 输入华氏温度的值。

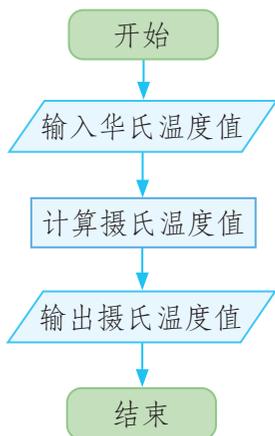
第 2 步: 利用温度计量方式的转换公式, 计算摄氏温度的值。

第 3 步: 输出摄氏温度的值。

以上过程是一个按顺序依次执行各个步骤的结构, 通常把这种控制结构称为顺序结构。顺序结构是算法设计与描述中最基本的结构。

为了更直观地描述以上步骤, 可以使用流程图描述。

流程图用一些图框来表示各种类型的操作, 在框内写出操作的内容, 然后用带箭头的线把它们连接起来, 以表示执行的先后顺序和方向。



算法的流程图

圆角矩形: 表示算法的开始或结束。



直角矩形: 表示过程的执行与处理。



平行四边形: 表示数据的输入或输出。



带箭头的连接线: 用来连接各个框图, 表示执行的方向, 也叫流程线。



教师先带着学生复习算法的含义——算法是通过明确的、可执行的操作步骤描述的问题求解方案, 然后结合前面的学习, 进行知识的复习巩固。

教学中应始终关注学生解决问题能力的培养。从带着学生分析问题开始, 提供学习支架, 到后面逐渐放手, 使学生具备相应能力。

问题解决过程也是思维细化的过程, 即把大问题分解为若干小问题, 解决这些小问题, 整个问题就会迎刃而解。

这里要向学生介绍流程图的基本含义, 解释流程图中图框的功能, 包括开始和结束的表达、表示执行方向的流程线等。

可以按照顺序呈现每个步骤的框图, 并用流程线连接起来。

在这个学习过程中, 要引导学生认识算法的顺序结构。

用流程图表示顺序结构时, 可以一目了然地看到每个步骤之间的关系和执行顺序。

教学时, 可以一边描述一边带领学生用实物图框在黑板上拼出流程图。同时学生可以用铅笔在纸上同步绘制, 进一步加深认识。

## 学习活动2

### 计算大豆种子的数量

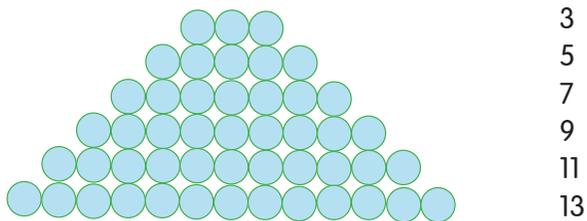
在实验室做实验时，科学课老师分发大豆种子让同学们进行发芽实验。

第1小组分3粒、第2小组分5粒、第3小组分7粒……按此规律，计算第10小组的种子是多少粒？老师一共分发了多少粒种子？

参照下图，可以用以下公式计算。

第  $n$  组的种子数  $m = n \times 2 + 1$

一共分发的种子数  $s = n \times (3 + m) \div 2$



第1小组至第6小组种子数示意

因此，第10组的种子数  $m = 10 \times 2 + 1 = 21$ （粒）

一共分发的种子数  $s = 10 \times (3 + 21) \div 2 = 120$ （粒）

在以上通过公式计算获得答案的求解过程中，计算公式也是一种简单的算法。求解的具体步骤描述如下。

第1步：输入要求解的组数  $n$ 。

第2步：用公式  $m = n \times 2 + 1$ ，计算第  $n$  小组的种子数  $m$ 。

第3步：用公式  $s = n \times (3 + m) \div 2$ ，计算一共分发的种子数  $s$ 。

第4步：输出  $m$  的值和  $s$  的值。

以上计算过程也是一个按顺序依次执行各个步骤的顺序结构。

教学时，可以直接用公式计算，也可以引导学生观察、推理获得公式，让学生体验推理的具体过程。以下是求解过程示例。

第1步：观察这一组数的规律，发现是从3开始的奇数。

第2步：从这组数中发现规律。

第1小组是3粒，表示为  $1 \times 2 + 1$ ；

第2小组是5粒，表示为  $2 \times 2 + 1$ ；

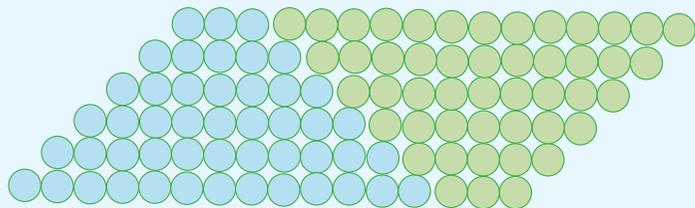
第3小组是7粒，表示为  $3 \times 2 + 1$ ；

以此类推，第  $n$  小组的种子数就应该是  $n \times 2 + 1$ 。

用公式表示为： $m = n \times 2 + 1$

第3步：观察发现，这是第一个数为3、差值为2的一组数。

第4步：要计算图中用小圆表示的种子总数，可以用正反放置的两个梯形组成平行四边形，通过求平行四边形的面积来求解。



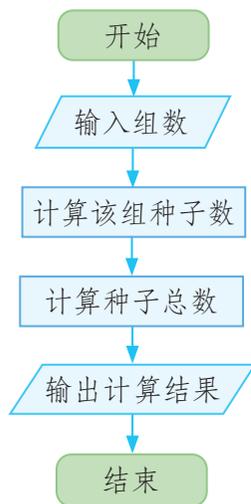
两个梯形组成平行四边形

第5步：依据上述分析，可以得到求这组数总和的公式。

$$\begin{aligned} s &= \text{组数} \times (\text{第1小组的种子数} + \text{第}n\text{小组的种子数}) \div 2 \\ &= n \times (3 + m) \div 2 \end{aligned}$$

借用求平行四边形中物体的个数来推理获得公式，对于小学生而言，能迁移数学的学科知识，把抽象的数学问题形象化。

同样地，可以用流程图直观地表示。



算法的流程图

进一步思考：

在数学、科学等学科中，你还学习了哪些公式？这些公式起到了什么作用？其中包含着什么样的算法？

## 拓展与提升

为提升学生阅读的积极性，某教育网站推出一项活动：阅读签到能够获得积分，积分可以换购图书。具体细则是：签到1次得1个积分，签到2次共得4个积分，签到3次共得7个积分，签到4次共得10个积分……

按此规律，计算签到15次的积分。

教师引导学生按照顺序绘制每个步骤的图框，并用流程线连接，形成结构清晰的流程图，进一步认识算法的顺序结构。

从算法视角描述计算过程。

1. 算法就是计算的方法，根据问题解决的要求，通过计算获得问题的答案。
2. 计算的步骤和顺序会直接影响结果和问题的求解。
3. 没有计算方法的指导，很难进行计算，也就得不到答案。

相比用数学方法，从算法实现（即用计算解决问题）的角度，引导学生思考、探究。这里从学生熟悉的数学运算出发，为后续算法的学习奠定基础，让学生知道数学运算中存在不同算法。

教学时还可以选择更适合本校学生的其他案例，通过类比帮助学生认识算法。

此处引导学生在计算过程中观察、发现其中存在的计算规律，进一步加深对算法的认识，巩固、迁移所学知识。

$$\begin{aligned} \text{参考答案： 签到 15 次的积分} &= 3 \times 15 - 2 \\ &= 43 \end{aligned}$$

## 第 6 课

### 判断选择用分支

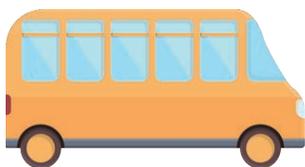
#### 学习目标

1. 通过购买公园门票的情境，认识算法的分支结构，初步学会用自然语言和流程图描述分支结构。
2. 通过分析公园购票的规则，能够确定分支结构的判断条件和执行路径，培养分析问题的能力。

在日常生活中，我们乘坐高铁、公交车，观赏电影，听音乐会，进入游乐场、公园等，大都需要购票，不同场合通常会有不同的收费标准。这些标准通常会对老人、儿童、学生、军人、残障人士等，给予相应的优惠。



高铁



公交车



电影院

结合自己的认识，思考下面的问题。

各类购票的收费标准中，存在哪些规则或条件？

教学重难点：

1. 分支结构的判断条件和执行路径；
2. 用流程图描述分支结构。

我们身边很多事情都蕴含着算法，本课借助生活实例，引领学生发现身边的算法，知道用自然语言或流程图描述算法。以公园购票为学习场景，逐步地分析算法中的输入、处理和输出，以及分支结构在算法中的意义。

学生生活中一般都有购票经验，教师可以通过图片、视频等介绍购票过程和收费标准。

教师先要引领学生明确购票场景。不同场景的购票标准会存在不同的规则或条件，用于决定购票的资格或适用的票价。依据购票标准，可以识别和确定符合条件的人员，从而提供相应的票务服务。

知道购票标准中存在判断与选择，需要根据购票人的条件选择与之相应的购票标准，其中的条件可能是年龄、身份、身高等。确定购票人的标准后，就可以按规定购票。

## 学习活动 1

### 描述公园购票的算法

周末，小智和妹妹参加在某公园举办的青少年科技主题活动，小智在售票窗口购买了门票。小智与售票员的对话如下。

小智说：“我要买两张票，我和我的妹妹。”

售票员问：“你们的身高分别是多少？”

小智回答：“我身高138厘米，我妹妹身高116厘米。”

售票员告诉小智：“你妹妹身高小于120厘米，可以购买半价票，票价是5元。你身高超过120厘米，需要购买全价票，票价是10元。一共是15元。”

基于以上情境，讨论下面的问题。

1. 在购票过程中要经历哪些步骤？
2. 售票员采用的售票规则是什么？

首先，售票员需要询问购票人的身高信息，然后根据购票规则确定购票类型并计算价格，再告知购票人。

事实上，以上公园购票的步骤中包含一个分支结构的算法。

第1步：确定输入的数据。

需要获取购票人的身高数据。

第2步：描述判断的条件。

如果身高小于120厘米，可以购买半价票，票价是5元。否则需要购买全价票，票价是10元。

本环节让学生从生活情境入手，通过情境展示，借助学生的生活经验，探讨购票标准，明确本课探究的主题就是分支结构的描述。

通过思考与讨论，让学生了解公园购票算法中的基本流程“输入—处理—输出”，知道根据不同情况选择购票方案的标准，这有助于培养他们的解决问题能力。

针对问题1，通过情境分析，结合算法的基本流程，引导学生讨论公园购票的主要步骤。

针对问题2，引导学生分析，售票规则主要依据身高进行判断。

引导学生用自然语言描述购票过程中的判断条件和不同的执行结果，从而了解分支结构中判断条件与执行路径的关系。

此处逐步引导学生从用口头语言描述向用规范的算法描述发展，这个过程看似简单，实则是一个循序渐进的过程。

第3步：用分支结构规范化描述。

把判断条件的描述转化为分支结构的表达方式。

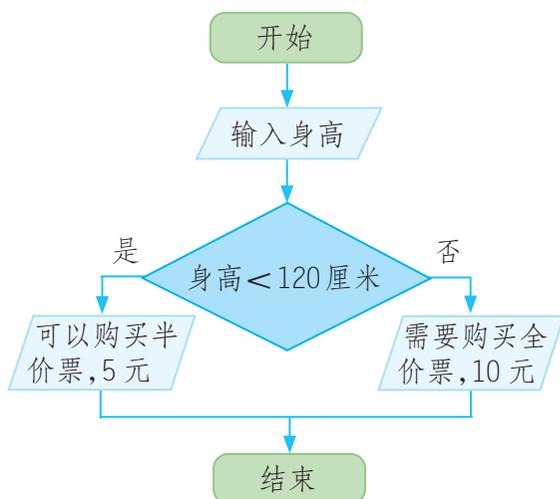
如果（身高小于120厘米）：

可以购买半价票，票价是5元

否则：

需要购买全价票，票价是10元

第4步：用流程图描述。



算法的流程图

在算法描述中，用菱形框来表示判断条件。

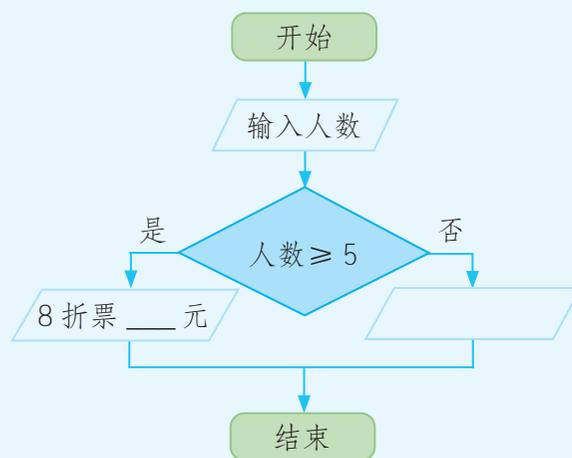
算法中的分支结构是一种选择结构，通过判断是否满足指定的条件，选择不同的执行路径。

进一步学习：在以上公园购票中，如果是学生团体（人数 $\geq 5$ 人），那么团体票在购票人原票价的基础上打8折。请尝试修改流程图。

可以给出部分流程图，让学生思考，再带领他们填写判断条件和不同的输出结果，有助于学生认识分支结构及其流程图的使用。

教学时，还可以引导学生思考，当以“身高 $\geq 120$ 厘米”作为判断条件时，如何修改流程图？

引导学生用不同的判断条件描述同一个问题，培养他们从不同角度思考同一个问题的思维能力。



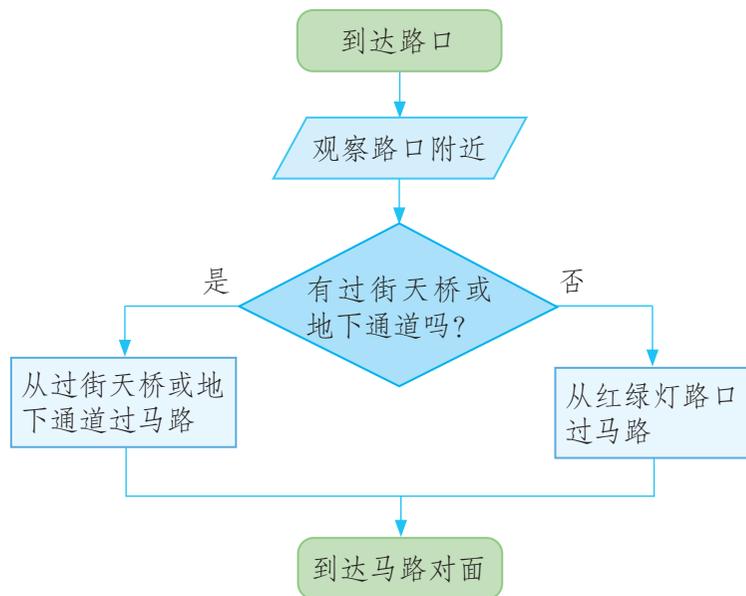
算法的流程图

## 学习活动2

### 描述过马路的分支选择

在现实生活中，算法往往不是固定的步骤，有些步骤需要根据外部条件来确定如何做。

例如，在路口过马路时，除了按照路口红绿灯的要求通过，还可以观察路口附近是否有过街天桥或地下通道。如果有，从条件框的“是”分支出发，走过街天桥或地下通道过马路；如果没有，则从条件框的“否”分支出发，仍然从红绿灯路口过马路。也就是说，要根据实际情况进行选择。



算法的流程图

进一步思考：假如要过马路去十字路口的斜对面，这时，可以根据红绿灯的情况灵活选择。你会怎么做呢？其中存在怎样的判断与选择？

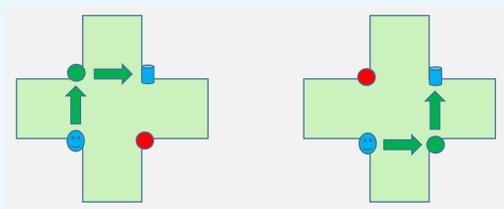
学生明确分支结构的判断与选择后，教师适时介绍分支结构的作用。分支结构是一种选择结构，根据条件是否成立而选择不同的执行方向。

这些条件判断可以使算法更加贴近实际生活，并让学生知道在不同情况下需要采取不同的行动。通过这样的训练，学生可以更好地理解算法的适应性。

要注意的是，应该根据目标和需求来确定，哪些步骤需要条件判断以及在什么情况下使用。这样能够培养学生分析问题、解决问题的能力，并且为他们之后学习算法验证与实现奠定基础。

还可以让学生了解，利用分支结构解决生活中实际问题的一般过程。生活中有很多场景都存在分支结构，认识生活中的分支结构及其判断条件，能够帮助我们解决问题时进行决策。

提示：到达路口后，可以从绿灯先亮起的方向过马路到对面，再等待另一个方向的绿灯亮，然后过马路。

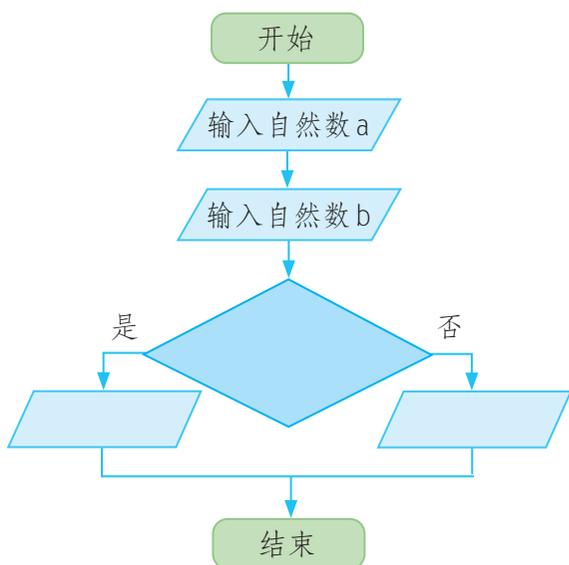


要去路口斜对面的示意图

## 拓展与提升

1. 生活中还有哪些存在分支结构的情况？请说出这些分支结构中的判断条件和执行路径。

2. 判断两个自然数的大小时，我们往往通过观察就可以比较出来。而要让计算机完成这项工作，就要用到分支结构。补充下面的算法流程图，要求输入两个不同的自然数，比较判断后输出其中较小的数。

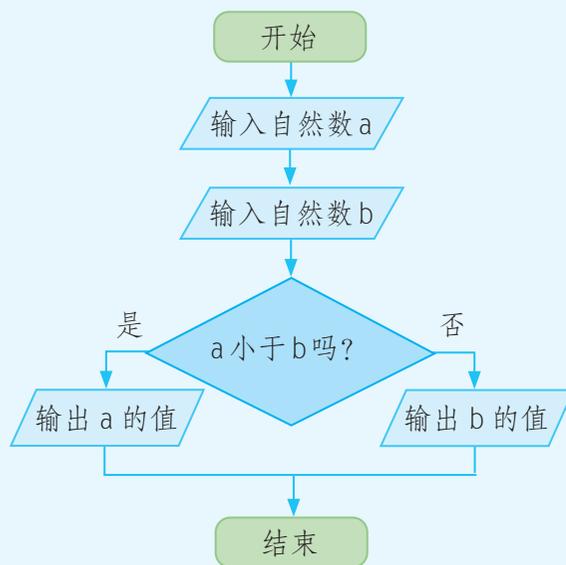


算法的流程图

1. 举例如下。

- (1) 通过年龄是否大于等于 18 周岁，判断是否是成年人。
- (2) 依据天气判断出门是否要带雨具。
- (3) 依据交通状况，选择出行方式或提前出发。
- (4) 依据放假天数，选择旅行的行程。

2. 参考提示。



算法的流程图

## 第7课

### 重复作用循环

#### 学习目标

1. 通过对路口红绿灯的观察，认识其中存在的重复规律与循环结构。
2. 能够用流程图描述循环结构，了解交通信号灯时长设置的作用和意义。

这一课学习算法中需要重复执行的操作——循环结构，并引入流程图描述算法的基本执行过程。

#### 学习活动1

##### 行人过红绿灯路口的步骤



遵循红绿灯过马路

要在红绿灯路口过马路时，通常需要遵循下面的步骤。

1. 到达有红绿灯的路口。
2. 观察路口的红绿灯，等待对面绿灯亮起。
3. 绿灯亮起时，表示可以过马路了。
4. 开始过马路，在这个过程中，要保持警觉，注意过往车辆。
5. 到达马路对面，顺利通过。

教学重难点：

1. 观察、发现生活中的循环结构；
2. 用流程图描述循环结构。

本课的重点是通过在红绿灯路口过马路的生活经历，引导学生回顾并描述其中的步骤，从而进一步认识算法的顺序结构，并引入循环结构，让学生能看懂相应的流程图。

可以通过提问开启教学过程：同学们平时在有红绿灯的路口是怎么过马路的？都有哪些步骤呢？

教师鼓励并耐心听取学生描述，可以用启发式教学，在引导学生思考发言的过程中，总结文字描述的步骤。

为了更加直观，可以用流程图来描述以上步骤。

这是一个按顺序依次执行各个步骤的顺序结构。



算法的流程图

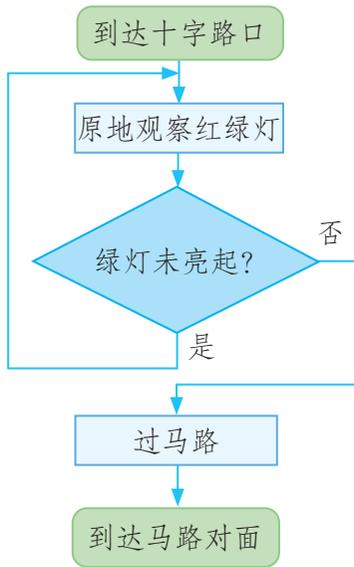
## 学习活动2

### 细分重复的操作——认识循环结构

在上面的例子中，到达路口后，仔细分析“观察并等待绿灯亮”这一步骤发现：需要不断地观察红绿灯并保持原地等待，直到红灯熄灭、绿灯亮起。也就是需要重复多次，次数不确定。

这样，描述这个算法时，可以把这个步骤改成循环结构。

在以上流程图中添加一个菱形的条件判断框，表示需要根据路口红绿灯的情况进行判断和选择。



算法的流程图

关于过马路的方法，可能有的学生还会说可以走过街天桥、地下通道等，对于这些情况，要积极鼓励学生思考并及时解答他们的问题。

要向学生说明，做同一件事，可能会有不同的算法步骤，会出现不同的顺序，但只要是适合的、有效率的、能达成目标的，都可以选择。学生只需要明白做事情时要有步骤，要遵循一定的先后顺序和基本方法。

教师先提问：哪些动作需要多次重复进行？

然后引导学生对前面顺序结构中的“观察并等待绿灯亮”这一步骤进行分析。同时指出，在算法描述中，把这种需要重复执行指定步骤的算法结构称为循环结构，也叫重复结构。

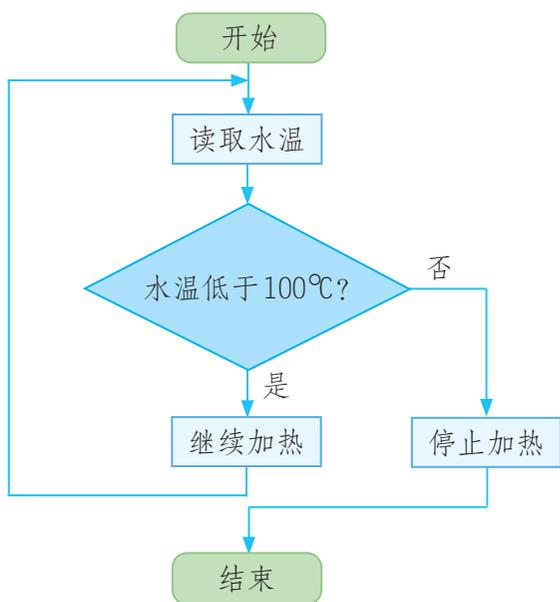
循环结构是算法的一种基本控制结构。在流程图中，条件判断框起到循环指示的作用，表示只要满足“绿灯未亮起”的条件，就要重复执行“原地观察红绿灯”的步骤，直到条件不再满足，即“绿灯亮起”时结束循环。

还可以与条件判断的流程图进行对比。循环操作可以重复执行某个步骤，条件判断可以根据不同情况选择不同的处理方式。

进一步学习：回想路口机动车信号灯的交替变换，可以发现，在红、绿、黄信号灯的交替变换中，其实也包含了循环结构。

思考这一过程中存在哪些循环结构。

在日常生活中，有很多利用循环结构进行控制或调整的例子。以热水壶水温达到  $100^{\circ}\text{C}$  自动断电为例，用流程图描述如下。



算法的流程图

学生通常对信号灯的变换情况较为熟悉，能够认识理解其中的变换规律。虽然不同路口的交替变换时间会有所不同，但一般都是一个循环往复的过程。教学时，要鼓励学生思考，把想法说出来。

热水壶能够自动控制的关键操作是不断地检测水温并与  $100^{\circ}\text{C}$  进行比较，判断出温度的高低后，让相关组件继续运行或停止，在达到  $100^{\circ}\text{C}$  后自动停止加热。

不管循环结构里的步骤如何复杂，都可以抽象为循环条件和循环体。为了减轻学生初学循环结构的认知负担，循环体的概念可以先不引入，在后续学习中，随着学生算法知识的积累再适时介绍。

可以在引导学生分析流程图的过程中，找出其中的循环判断条件和需要反复执行的步骤。同时也会发现，可以先判断是否满足条件再执行指定的步骤，也可以根据需要先执行指定步骤，再判断是否满足条件。

时间允许的情况下，可以启发学生寻找生活中哪些场景还存在循环结构。例如，扫地机器人能够避障，空调按设定的温度制冷，洗衣机到达指定时间自动停止等。

### 学习活动3

#### 交通信号灯的时长设置

现在，在一些人流量、车流量较大的路口，细心观察会发现，交通信号灯在不同时段，红绿灯的时长往往是不一样的。

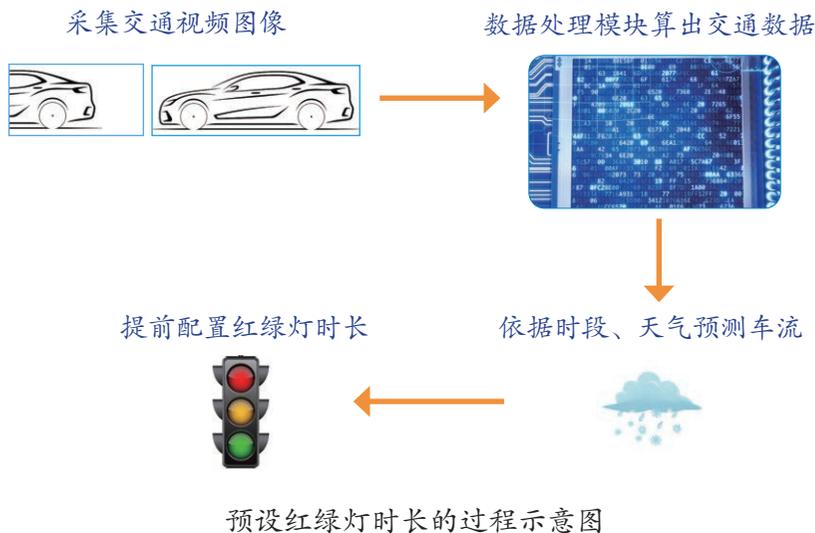
不同路口红绿灯的时长差异设置正是智慧交通的体现，这样的智能设置解决了很多拥堵问题，满足人们的便捷出行。

有了算法支持的城市交通控制系统，利用摄像装置采集图像、视频数据，并对这些数据进行计算处理，获得实时的交通流量数据，再依据这些数据适时进行调整与控制。

例如：为车流量大的方向配置较长的机动车通行时间；为人群聚集地点周边的路口配置较长的行人通行时间；预测未来某一时段或天气条件下的交通流量，提前做好信号灯的设置。

这一内容可以进行课堂讲解，也可以给学生一些指导，让学生们课下去观察、阅读和理解。

安排这部分内容的目的是提升学生的认识。从学生身边的遵循红绿灯过马路，拓展到城市的交通管理。让他们认识到：所有身边的“小”设置，到了全局就是“大”系统，让学生能够以小见大，打开视野，培养独立思考问题的能力。



## 拓展与提升

1. 假如你乘坐车辆出行，从机动车行驶的角度看，机动车信号灯的交替变换是否比人行道信号灯的变换更复杂？思考其中存在的判断条件和算法。

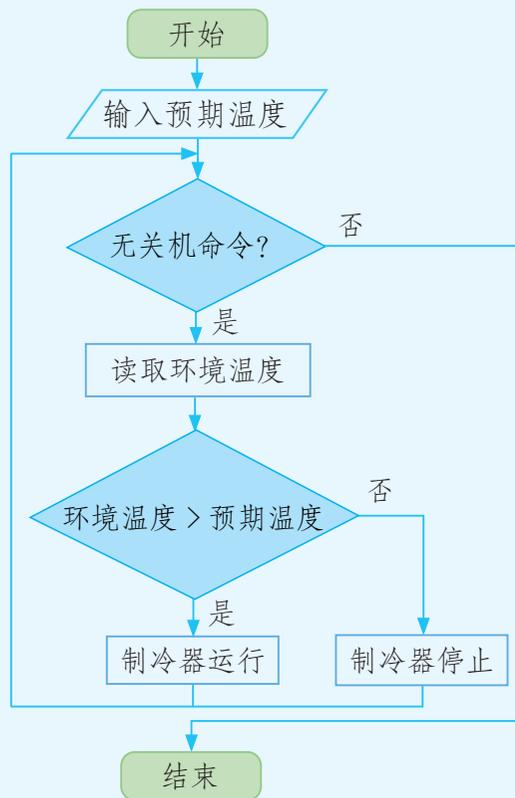
要求：基于算法的角度，从方向指引、变换顺序和时长设置等方面，把所学知识用来重新认识身边的场景、事物等。

2. 在空调制冷过程中，其中关键的一个环节是：空调通过温度传感器会实时获取当前室内温度，并与用户预设的温度值进行比较。如果当前室温高于预设的温度，空调中的制冷器就会运行并制冷，否则，制冷器就会停止运行。在这个过程中，空调会反复地获取当前室内温度并进行判断。

请尝试用流程图描述这一环节。

1. 让学生在学习了循环结构之后，通过教师的课堂引导或者课后的观察，体会算法描述的作用，了解其中的循环控制条件和循环规律。注意提示学生认真观察不同路口的信号灯变化，从算法角度发现并总结其中的规律。

2. 流程图提示。



## 第 8 课

### 算法验证与实现

#### 学习目标

1. 初步尝试编写简单的 Python 程序，对算法进行验证与实现。
2. 知道变量的含义，了解 Python 中 input()、int()、float() 和 print() 等函数的用法。

本课通过用程序验证与实现算法，体验用计算机解决问题的过程。

了解算法并不一定需要编程，可以通过日常生活中的问题分解和步骤规划来认识。然而，要让算法真正发挥效力，通常需要借助程序来实现。编程可以把我们的想法和思路转化为计算机能够“认识”的指令，并指导计算机按照设计的算法来执行任务。

#### 学习活动 1

##### 编程实现买水果计算价格

小智家里要来客人，他陪妈妈去超市买水果，每千克苹果 8.5 元，每千克香蕉 12.6 元。

请编写一个程序，帮助小智妈妈计算买水果需要的总金额。

先来讨论以下问题，了解编程让计算机执行的过程。

1. 我们设计的算法，计算机认识吗？
2. 怎样让计算机认识呢？
3. 如何编写程序，把这些步骤写成计算机能认识的呢？

教学重难点：

1. input()、int()、float() 和 print() 等函数的用法；
2. Python 中变量的含义及用法。

编程验证与实现的环境，建议使用 Python IDLE，也可以使用配套资源里的相关工具软件。

可以使用的 Python 平台很多，为了方便程序编写，Python 提供了基础的程序编辑器 IDLE。

本课活动安排建议：以学习活动 1 为主，其他学习内容作为分层教学时的选择。

从解决问题的步骤描述到编程实现算法，看似简单的过程，对于初学的学生来说是一个不容易理解的难点，是直观到抽象的跨越。所以这里通过设计问题链，把这个难点拆分成几个小问题，引导学生分析讨论。

问题 1：使用自然语言描述，计算机无法识别。

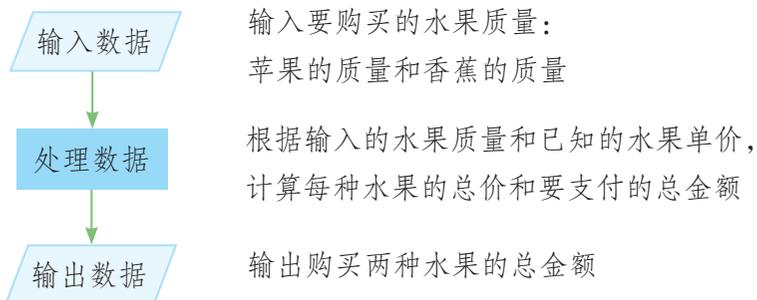
问题 2：将算法描述的过程通过程序告诉计算机。

问题 3：用计算机认识的程序语言写出每个步骤的指令。

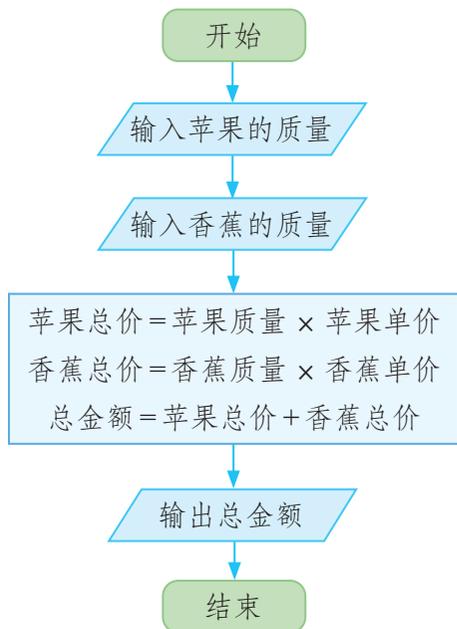
这样，在计算机的帮助下，就能利用算法解决问题。

下面分析要解决的“买水果计算价格”问题。

1. 要输入的数据有哪些？
2. 要对这些数据进行哪些处理？
3. 输出的数据是什么？



根据以上分析，可以画出下面的顺序结构流程图。



算法的流程图

先帮助学生读懂问题要求，再引导他们分析问题。找到问题的答案后，算法的流程图描述就容易了。

从学生已经熟悉的数学求解方法引入，用文字描述配合流程图的方式，有助于学生理解用算法解决计算问题的基本流程，把算法知识融入他们已有的数学知识之中。

基于算法的基本流程，引领学生分析购买水果的过程，找到数据的输入、处理和输出环节，并用流程图描述。

编程实现算法时，顺序结构是最简单的程序结构，也是最常用的程序结构。按照解决问题的顺序写出相应的程序指令，并让计算机按照自上而下的顺序依次执行。

下面分析程序实现的关键要素。

### 1. 输入数据

输入就是向计算机提供数据或指令。可以通过键盘输入文字、数值等内容，然后让计算机使用这些输入的数据完成各种处理任务。

在 Python 中，可以用 `input()` 函数作为输入指令来获取数据。

例如，执行下面的语句，可以把输入的苹果质量这个数保存到变量 `a` 中。

```
a = input('输入苹果的质量：')
```

### 2. 计算处理

执行下面的指令，观察显示的内容。

```
a = input('输入苹果的质量：')
```

```
b = a*8.5
```

第二条指令的含义是：计算  $a \times 8.5$  的值，即苹果的总价，并把这个值保存到变量 `b` 中。

执行上述指令，会出现一条 `TypeError`（类型错误）的提示信息，表示操作失败。原因是通过 `input` 输入到变量 `a` 中的数，被 Python 默认为是字符，就如同字母、符号、汉字一样，因此不能参与数值运算。

为此，要获得正确的计算表达式，需要将输入的这个数进行类型转换，也就是把这个数转换为浮点数（带有小数点的数）或整数后再进行计算。

执行下面的指令，就可以顺利完成苹果总价的计算。

```
a = input('输入苹果的质量：')
```

```
b = float(a) * 8.5
```

其中，`float()` 用于把输入的数据转换为浮点数。

### 3. 输出数据

在 Python 中，可以用 `print()` 函数作为输出指令来显示数据。

学习使用 `input()` 函数获取输入的数据时，先向学生介绍这个函数的作用，并示范如何用来获取输入的数据。

语句中的“=”是赋值号，而不是等号。这是学生理解上的一个难点，要着重讲解清楚。这里表示把语句右侧获得的数据或计算结果“赋值”给左侧的变量。

在 Python 中，算术运算符的用法与数学相同，只是表示方法略有不同。算术运算符的优先级别是：先乘除后加减，如果要改变优先级别，需要加上小括号。

四个常用算术运算符的用法

运算符	描述	举例	运算结果
+	加号，把两个数相加	12+3	15
-	减号，一个数减去另一个数	12-3	9
*	乘号，两个数相乘	12*3	36
/	除号，一个数除以另一个数	12/3	4.0

数据类型需要转换后才能正确计算，是学生理解程序指令时的又一个难点，教学时要向学生解释清楚原因，并通过示范演示引导他们认识。

例如，继续执行下面的指令，窗口中就会显示计算得到的苹果总价。

```
print(b)
```

即利用 print(b) 指令可以输出并显示变量 b 中保存的数值。

下面编程实现买水果计算价格并输出。

第 1 步：启动 Python，新建一个程序文件，逐行输入下面的程序代码，然后保存程序文件，完成“买水果计算价格”的编程任务。

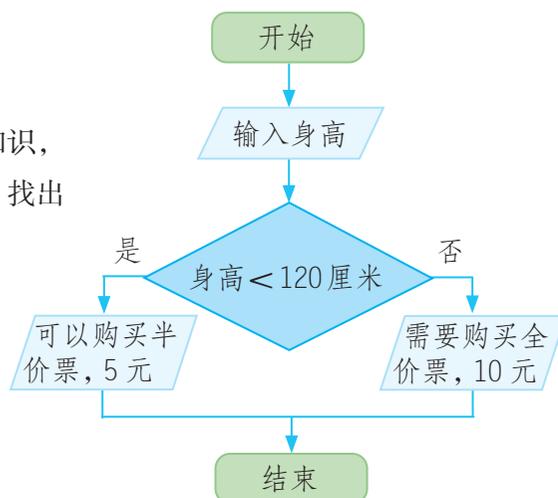
```
a = input(' 输入苹果的质量: ')  
b = input(' 输入香蕉的质量: ')  
c = float(a)*8.5  
d = float(b)*12.6  
e = c+d  
print(' 总金额: ', e)
```

第 2 步：运行程序，分别输入苹果和香蕉的质量，就可以看到总金额。

## 学习活动 2

### 编程验证公园购票的算法

第 1 步：回顾第 6 课所学知识，分析公园购票的算法基本流程，找出其中的输入、处理和输出。



算法的流程图

学习使用 print() 函数输出变量中的数据，同样需要边示范边讲解。例如，输入 5，程序就会把 5 保存到变量 a 中，再通过 print() 函数输出显示出来。

执行“File（文件）”菜单中的“New File（新建文件）”命令，打开编辑窗口界面，输入程序代码，然后执行“File（文件）”菜单中的“Save（保存）”命令，可以把程序保存起来。

程序代码编写完成后，执行“Run（运行）”菜单中的“Run Module（运行模块）”命令，或者敲 F5 快捷键，就可以运行程序。

引导学生了解算法实现的过程与方法，体验编程实现与验证算法过程中的基本过程：编写—运行—调试—改进。

先组织学生回顾第 6 课所学知识。学生经过前面几课的学习，对算法的分支结构有了初步感知。此处通过回顾公园购票算法的流程图，了解执行过程，再用 Python 程序验证算法。

第 2 步：依据算法流程图，补全用红色文字描述的程序代码。

# 输入身高数据

```
height = int(input('请输入身高 (厘米): '))
```

# 判断购票的类型和价格并输出

if 判断条件符合：

```
    print('身高为', 身高变量, '厘米, 可以购买半价票, 5 元')
```

else:

```
    print('身高为', 身高变量, '厘米, 需要购买全价票, 10 元')
```

第 3 步：启动 Python，打开配套资源中的这个程序文件。

第 4 步：运行程序，输入不同的身高数据，观察执行情况。

在 Python 程序中，通常用“#”符号表示注释，用来标注语句的主要内容和功能。

在以上程序中，int() 是取整函数，用于把身高数据转换为整数，从而进行比较与判断。用“if (如果)……else (否则)……”语句实现了分支结构的判断。

在 Python 中，通常用 if 语句构成单分支结构。

基本格式表示如下。

if 判断条件：

语句组

if 语句首先判断条件是否满足。如果满足条件，就执行“语句组”里的语句；如果不满足条件，则直接进入下一步的操作。

教师适时给出完整的代码，并进行分析介绍。带领学生初步了解语句与算法的对应关系，认识语句的功能，不必要求语句的用法。

# 输入身高数据

```
height = int(input('请输入身高 (厘米): '))
```

# 判断购票的类型和价格并输出

if height < 120:

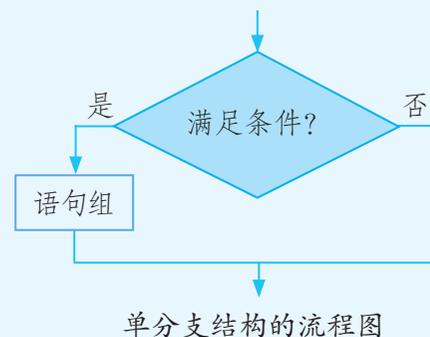
```
    print('身高为', height, '厘米, 可以购买半价票, 5 元')
```

else:

```
    print('身高为', height, '厘米, 需要购买全价票, 10 元')
```

执行“File(文件)”菜单中的“Open(打开)”命令，可以打开程序文件。输入程序验证算法时，可以不输入注释内容。

为了降低操作难度、节省课时，这里不建议让学生输入代码，而是直接打开配套资源中的程序并运行。



通常用 if……else……语句构成双分支结构。

基本格式表示如下。

```
if 条件:  
    语句组 1  
else:  
    语句组 2
```

if 语句首先判断条件是否满足。如果满足条件，就执行“语句组 1”里的语句；如果不满足条件，则执行“语句组 2”里的语句。

进一步学习：调试转换温度计量方式的程序。

根据下面“华氏—摄氏”温度计量方式转换的公式，有同学编写了一个程序，实现温度计量值的转换。

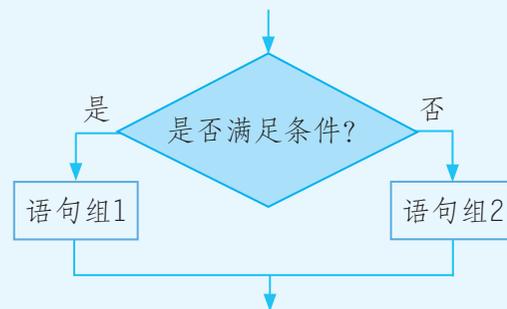
摄氏温度 =  $5 \times (\text{华氏温度} - 32) \div 9$

```
华氏温度 = input('请输入一个华氏温度的值：')
```

```
摄氏温度 = 5 * (华氏温度 - 32) / 9
```

```
print('对应的摄氏温度为：', 摄氏温度)
```

运行这个程序，观察出现的情况，适当调试修改。



双分支结构的流程图

运行程序时，同样会出现类型错误的提示信息。

要获得正确的计算表达式，同样要将输入的值进行类型转换。

即需要把输入的值转换为浮点数，再利用转换公式进行计算。

```
摄氏温度 = 5 * (float(华氏温度) - 32) / 9
```

这个程序也可以在获取输入值时就转换成浮点数，然后再用转换公式进行计算。

```
华氏温度 = float(input('请输入一个华氏温度值：'))
```

```
摄氏温度 = 5 * (华氏温度 - 32) / 9
```

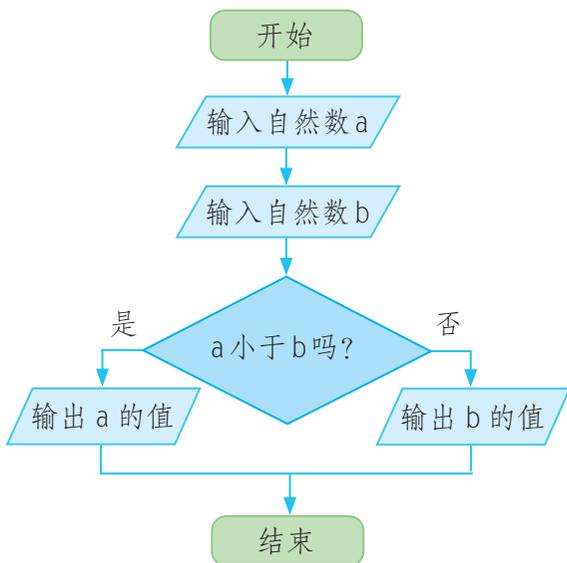
总结强调：编程不仅是一种技能，更是一种思维方法。先把复杂问题分解为基本步骤，再通过编程把这些步骤转化为计算机能够识别的指令，并让计算机执行这些指令，从而完成各种任务。

## 拓展与提升

1. 小智陪妈妈去超市买水果时，遇到超市有活动优惠，购买的单种水果超过4千克就可以打9折。请修改前面编写的程序，重新计算买水果需要支付的总金额。

2. 尝试用程序验证与实现第6课中的以下算法。

输入两个不同的自然数，比较判断后输出其中较小的自然数。



算法的流程图

1. 参考程序如下。

```
a=float(input(' 输入苹果的质量: '))
b=float(input(' 输入香蕉的质量: '))
if a>4:
    c=a*8.5*0.9
else:
    c=a*8.5
if b>4:
    d=b*12.6*0.9
else:
    d=b*12.6
e=c+d
print(' 总金额是: ', e)
```

2. 通过简单比较两个整数大小并输出较小数的判断操作，引导学生进一步了解程序实现，认识算法与程序的关系。

```
a = int(input(' 输入自然数a: '))
b = int(input(' 输入自然数b: '))
if a<b:
    print(' 较小数是: ', a)
else:
    print(' 较小数是: ', b)
```

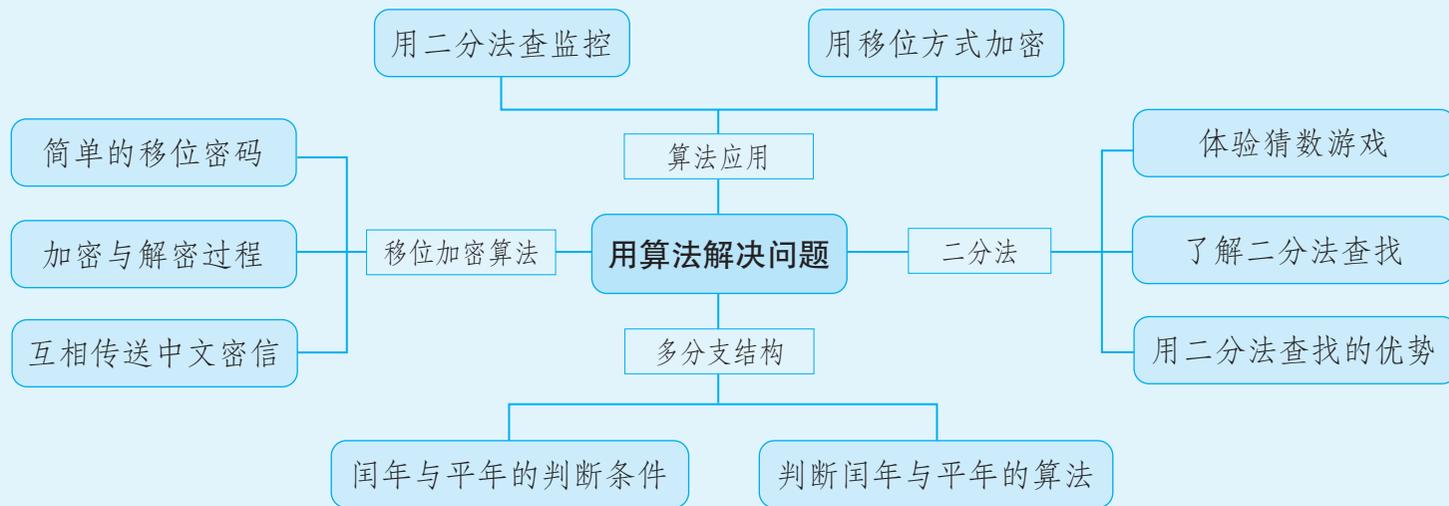
### 【引言】

在上一单元的学习中，通过生活实例，认识了算法的三种基本控制结构及其描述的基本方法。在本单元的学习中，进一步结合典型问题，运用算法的基本控制结构来解决实际问题。用算法解决问题需要提取问题的关键因素，确定解决问题的相关因素，忽略非必要因素，清晰地认识问题、准确地描述问题。可以结合数学学习中的问题分解方法，通过分析问题的已知信息、求解目标、条件关系来解决问题。

本单元的移位密码、用二分法猜数、判断闰年与平年等案例都涉及基本的数学知识和方法。这些知识和方法可以作为学生设计、运用算法解决问题的基础，有利于学生借助原有知识经验用计算思维解决熟悉的问题，学习新的知识与技能。

本单元的学习强调实践和实际应用。学生通过对所求解问题的算法描述与设计，观察执行步骤，讨论执行路径，判断算法是否解决了指定问题，了解算法与程序之间的关系，知道编写程序的目的是实现、验证算法。

### 【内容结构】



## 【教学实施条件】

1. 第 9 课可以在普通教室实施教学，配备教师用的计算机，能够运行配套资源中的游戏、程序等。
2. 第 10 课、第 11 课、第 12 课在信息科技实验室实施教学，具备运行 Python 的环境。

## 【学习目标】

1. 了解移位密码及其算法，知道明文与密文的关系，了解加密算法在保护信息安全方面的意义。
2. 了解二分法的基本思想，结合二分法了解算法的效率与数据规模之间的关系。
3. 进一步认识算法的分支结构，学会从问题情境中提取关键因素、确定判断条件，知道判断条件对算法执行结果的影响。
4. 认识 Python 中分支结构的判断条件与程序执行路径的关系，体会用计算机解决问题的基本方法。

## 【评价建议】

下表可作为单元学习评价的参考。

评价内容	学生自评	教师评价
知道移位密码的规则，认识移位加密算法	☆☆☆☆☆	☆☆☆☆☆
知道明文与密文的关系，了解加密算法在保护信息安全方面的意义	☆☆☆☆☆	☆☆☆☆☆
了解二分法的基本思想，知道用二分法查找的基本步骤	☆☆☆☆☆	☆☆☆☆☆
在用二分法查找的过程中，明确算法执行次数与数据规模之间的关系	☆☆☆☆☆	☆☆☆☆☆
结合具体生活实例，明确地描述多分支结构，能识别其中的判断条件	☆☆☆☆☆	☆☆☆☆☆
能用程序验证、实现相关的算法	☆☆☆☆☆	☆☆☆☆☆
体会使用分支结构解决实际问题的方法	☆☆☆☆☆	☆☆☆☆☆
在小组合作的任务中，积极与同学交流，主动思考与实践	☆☆☆☆☆	☆☆☆☆☆

## 第9课

### 互传密信有诀窍

#### 学习目标

1. 初步了解移位密码的基本规则，看懂用流程图描述的移位加密算法，知道明文与密文的关系。
2. 了解加密算法在保护信息安全方面的意义，增强信息安全意识。

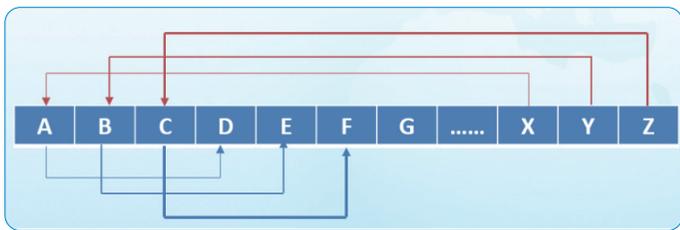
这一课通过简单且广为人知的凯撒密码，初步了解移位加密算法。

#### 学习活动1

##### 认识简单的移位密码

凯撒密码采用的是一种简单的移位加密算法。规则是把明文中的每个字母按照一定的规则移动，从而生成密文。

明文中的所有字母在字母表中向右移动指定位后被替换成密文。例如，当字母表中的字母向右移动三位时，A替换成D，B替换成E，以此类推。如果移动后的字母超过了字母表最后的字母Z，则需要回到字母表的开头，继续向右移动。



字母向右移动三位示意图

教学重难点：

1. 信息加密传输的过程，明文与密文的关系；
2. 加密算法在保护信息安全方面的意义。

可以先以讲故事的方式，简单介绍凯撒密码，让学生知道加密的重要性、了解移位密码的历史意义。

凯撒密码是历史上真实存在的一个经典加密方法。在罗马共和时期的战争中，军事指挥官为了预防情报所表达的信息在传递过程中被敌军截获，往往会依据指定的规则对情报进行加密，只有知道规则的人才能解密。据说当时发明的这套加密方法，保护了重要军情信息。

到了现代，凯撒密码已经无法完全保证人们的信息安全，很少被直接应用，但常常作为一个部分被用于其他的加密方法之中。

明文 A B C D E ..... T U V W X Y Z  
 密文 D E F G H ..... W X Y Z A B C

明文 A B C D E F G H I J K L M N O P Q R S T U V W X Y Z  
 密文 D E F G H I J K L M N O P Q R S T U V W X Y Z A B C

移动三位时明文与密文的对应关系

解密是加密的反向操作。

例如，将加密时的向右移动三位改为向左移动三位，就可以解密。如果移动后的字母超过了字母表的最前面一个字母 A，则需要回到字母表的最后，继续向左移动。

密文 A B C D E F G H I J K L M N O P Q R S T U V W X Y Z  
 明文 X Y Z A B C D E F G H I J K L M N O P Q R S T U V W

密文与明文的对应关系

上述移动三位的加密算法，可以用自然语言描述如下。

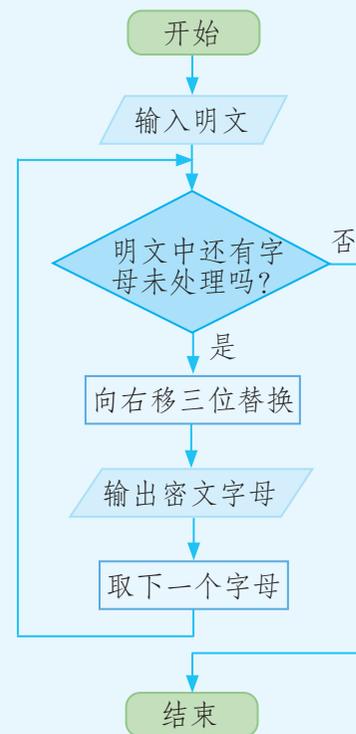
第 1 步：输入要传送的明文。

第 2 步：查看第一个字母，将字母向右移动三位并输出对应的字母。

第 3 步：重复第 2 步操作，分别判断并输出第二个字母、第三个字母、第四个字母……直到处理完成原文中的所有字母。

可以看出，以上构成了一个循环结构的算法。

可以结合流程图进行讲解，分析每个步骤或环节，让学生知道自然语言描述与流程图的对应关系，进一步加深用流程图描述算法的意识。



算法的流程图

教学时，可以带领学生做移位密码游戏。教师出示密文，让学生解密。例如，出示“jrrg prurlqj”，答案是“good morning”。

用这种方式多猜几组，让学生熟悉移位密码的规则。

## 学习活动 2

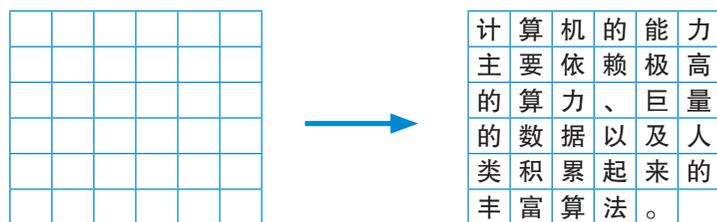
### 与同学互传中文密信

要求：两个同学一组，利用移位加密的规则，给对方写一封简单的中文密信，然后解密并相互验证。

具体的操作过程如下。

第 1 步：依据自己所写密信的字数，确定一个多行多列的方格，把明文依次放入方格中。设定的行数、列数成为密钥的第一部分。

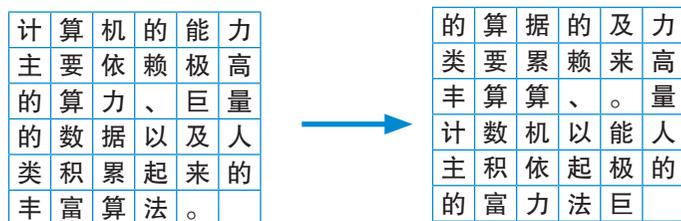
例如，把原文“计算机的能力主要依赖极高的算力、巨量的数据以及人类积累起来的丰富算法。”放入 6 行 6 列的方格中。



把原文放入方格中

第 2 步：选定要移位的列和移位的方式，如奇数列、下移 3 行。这是密钥的第二部分。

第 3 步：把奇数列的文字依次下移 3 行，找到对应的方格填入，下移超过最后一行的文字，回到第一行继续下移替换，构成密文。



把奇数列的文字下移、替换

接下来，组织学生两人互相传送中文密信。这个学习活动是在学生知道移位密码算法之后的进一步巩固，从移位算法角度进行的知识迁移。目的是让学生们在加密与解密的游戏里体会加密算法的规则和过程。

学习活动 2 背后隐藏的算法也需要使用循环结构，逐个对指定的文字进行移位替换操作，从而达到加密的目的。对文字加密时，如果移位后超出了最后一行，就要回到第一行继续移位替换，所以需要设置判断条件。

密钥由方格数和移位规则两部分组成。在相同移位规则下，不同行数和列数的方格，加密得到的密文是不同的。因此解密时，不仅要推算移位规则，还要了解方格的行数和列数。

需要提醒学生的是，方格数可以依据要传送的文字数量确定。把密文传给同学后，要给对方关于密钥的提示，必要时向对方讲解自己加密的方法。

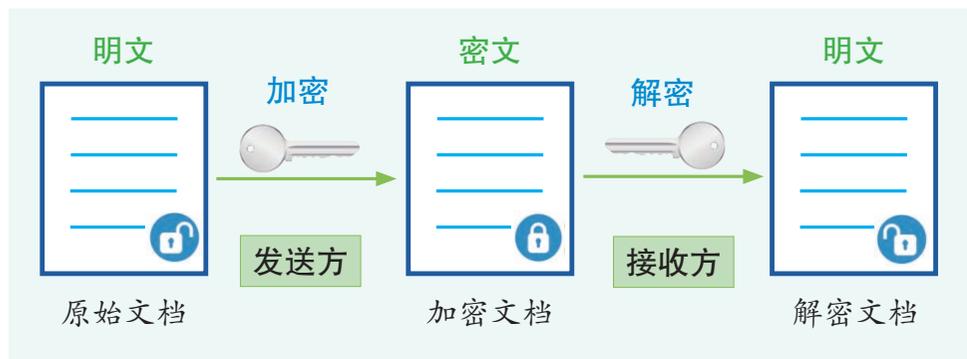
第4步：两位同学互传加密后的密信，并尝试解密。

解密时，按照加密的规则反向操作。例如，将加密时向下移动改为向上移动。可以给对方一些提示，如自己所用的方格数，移位的行数等，也可以双方共享密钥。

### 学习活动3

#### 讨论交流：信息加密与信息安全

在信息传输过程中，为确保信息安全，发送方往往需要把原文信息进行加密后再传输，接收方收到信息后通过解密，获得原文信息。



信息传输过程的加密与解密

以小组讨论方式进行交流。

1. 为信息加密时，在字母表中改变移动的位数，如果不是移动三位，而是移动五位、七位等，是不是更安全？
2. 如果不知道移动的位数，如何破解移位密码？
3. 加密时，把英文字母替换成指定中文段落中的文字，或者把中文替换成字母表中指定的字母，是不是更安全？

教学时注意关注算法的基本流程。

加密时，明文是输入，移位操作就是处理，密文是输出。解密时则反之。

实际应用中，只能发送方和接收方知道密钥，不能被其他人知道或猜出。

如果不知道移动位数，或者移动位数较多，通常不容易解密。但如果发送方和接收方事先约定好，接收方就很容易解密。

此外，可以利用循环操作遍历所有移位次数进行“暴力破解”。比如，在26个英文字母中，把所有可能的情况全部尝试一遍。这里可以引导学生数一数不同的循环次数，了解利用计算机的快速运算能力，将“笨办法”变得简单实用。

现实应用中，还可以利用频度分析，找到密信中出现次数最多的字母，然后推测出移位的次数。

发送方和接收方事先约定好所有文章段落和移动方式（向前或向后、移动几位等），在双方保守秘密的情况下，这种方式的安全性高于只用字母或只用中文的方式。

进一步思考：学习了移位加密算法后，有同学说，以后给自己或家长设置密码时，也要运用所学的算法规则。先想一个自己很容易记住的密码，然后利用移位、替换等方式，变成不容易被别人猜到的密码。

你认为这个做法好吗？与小组同学交流看法。

## 拓展与提升

1. 尝试用所学加密算法，设计一个自己的密码本，并用这个密码本的加密方式，加密一则信息并传送给同学或家长，看看他们能否顺利解读出来。
2. 尝试通过网络搜索，查询还有哪些常用的加密算法以及这些算法的关键因素。

这里引导学生进一步思考，把想法大胆说出来，有助于学生巩固所学知识并迁移应用，教学时要鼓励学生发散性地去独立思考、积极表达观点。

1. 要适当给出加密的方法提示，以便对方解读消息内容，降低解密的难度。
2. 教师可以给学生一些搜索的关键词，或者提供一些课外阅读资料，引导学生明确地去查询和阅读。

## 第 10 课

### 猜数游戏有捷径

#### 学习目标

1. 通过猜数游戏认识二分法，从具体情境中确定二分法查找的对象、范围和判断条件。
2. 了解二分法在生活中的应用，知道用二分法解决实际问题，体会用算法解决问题的优势。

这一课以猜数游戏为例，在体验二分法猜数的过程中，学习用二分法查找数据的初步知识。

#### 学习活动 1

##### 体验猜数游戏

两个同学一组，依据以下规则，玩一玩猜数游戏。

1. 有一个 1 到 100 之间的自然数，通过猜测的方式来确定这个数。
2. 每次猜测并说出一个数后会得到一个提示：说出的数相对于要猜的数是大了还是小了，或者猜对了。
3. 通过不断缩小数值范围来逼近要猜测的数，直到猜对这个数。

活动过程如下。

1. 一位同学默想一个数，让另一位同学猜。
2. 两位同学各猜一轮，记录猜测的次数和结果，填写记录表。
3. 在猜数过程中，观察、思考猜数范围的变化规律。

教学重难点：

1. 从具体情境中确定二分法查找的对象、范围和判断条件。
2. 用二分法解决实际问题。

猜数游戏是典型的二分法应用，可以帮助学生直观地认识二分法查找的规律和执行过程。学生在参与游戏活动中，观察数值范围的变化、判断条件的运用，从而更好地掌握二分法查找的规律，加深他们对算法的理解。

教学时，可以围绕某个主题，创设猜数的情境来展开学习活动。

还可以这样安排：教师在纸上写一个 1 至 100 之间的数，先让一位同学来猜一次，然后根据提示调整猜测的范围，接着让另一位同学猜一次，同样地，根据提示调整猜测范围，继续轮流让其他同学来猜，直到猜到这个数。

游戏本身对于学生来说并不复杂，在学生体验后，要引导他们对规律进行分析和总结。例如，可以分析以下问题。

1. 在这个游戏中，已知信息是什么？

已知要猜的数在 1 至 100 之间。

2. 猜数的判断条件是什么？

判断条件是说出的数与被猜的数比较之后，所了解到的数是“大了”“小了”还是“对了”。

通过游戏发现：如果猜出的数不正确，猜数的范围就可以根据所猜数的位置逐渐缩小，直到找到符合的数。

同时还发现：猜数范围越小，猜出指定数的可能性就越大。

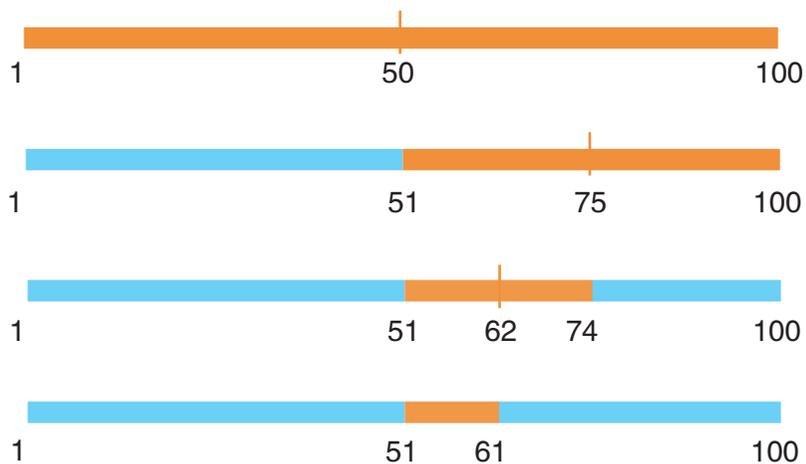
那么，是否有快速缩小猜数范围的算法呢？

那就是二分法。

二分法是一种高效的搜索算法，适用于在有序数据中查找目标数。

二分法通过将查找范围不断分成两半，并与目标值进行比较，选取目标值所在的那一半，从而迅速缩小查找范围。

例如，在 1 至 100 的自然数中，要猜测的数是 58 时，用二分法来猜，第 1 次猜 50，然后继续细分，最多只需猜 4 次，就可以把范围缩小为 51 至 61，再进一步细分，很快就会获得答案。



用二分法猜数过程示意图

学生在活动中，确定了判断条件，认识到猜数范围随着猜数过程在逐渐缩小，鼓励学生识别猜数游戏中的问题关键，揭示猜数范围与被猜数的关系，即猜数范围越小猜出数的概率就越大。从而引出进一步探索的问题：是否有快速缩小猜数范围的算法，让猜数的效率更高。

这里可以引导学生观察用不同颜色线段表示的数值范围变化。

例如，小组讨论以下问题。

1. 蓝色线段表示什么，橙色线段表示什么？
2. 线段上的数表示什么？
3. 每次猜测后，猜数的范围缩小了多少？

在学生有了初步认识后，结合图示观察二分法查找的特点，每次都应将范围缩小到原来的一半。

## 学习活动2

### 二分法查找的算法描述

扩大猜数范围，利用二分法继续猜数，数的范围是1至1 000。

两人一组，一位同学默想一个数，另一位同学猜，直到猜对为止。记录猜测的次数和范围，两位同学各猜一轮。

当数值范围扩大到1至1 000时，用二分法来猜最多需要10次。如果把数值范围扩大到1至10 000、1至100万，最多猜测次数分别是14次和20次。

相比按顺序依次猜测，猜测次数大大减少了。

下面描述二分法查找数据的算法。

第1步：在要查找的一组有序数中，取中间位置的数与要查找的数比较，如果两者相等，则查找成功。

第2步：如果不相等，就以中间位置的这个数为分界，把这组数分成两部分。中间位置的数大于查找数，则进一步查找比这个数小的那部分；否则进一步查找比这个数大的那部分。

第3步：重复以上两个步骤，直到找到满足条件的数。

## 学习活动3

### 用二分法解决问题

寻找拿错快递的人。

在早上8时到晚上8时之间，快递员放在小区货架上的一箱货物不知道被谁错拿了，但有监控摄像头一直可以拍摄到这个货架。现在需要查看监控视频，找到拿错快递的人。

请想办法提高查看监控视频的效率，帮助快递员尽快找到货物。

扩大猜数的范围，继续让学生参与更大范围的猜数游戏，探索数据规模的变化对算法执行次数的影响。本部分内容比较容易理解，可以让学生通过自主探究展开学习活动。

必要时，可以继续拓展数据范围：数值范围是1至1亿时，所需次数大约是27次。假设猜1次需要1秒，那么用顺序逐个猜的方式，就需要约1 157天（约为3年），用二分法只需不到27秒。

这些分析让学生体会到二分法的优势，即使数据范围很大，也能快速缩小数值范围并找到正确答案，从而感受利用算法提高问题解决效率的优势，激发学生进一步学习算法的兴趣。

二分法也要用到条件判断，不过这不是本课的重点。本课主要是让学生逐渐理解算法的重要性，知道算法对解决问题的效率有直接影响。用不同算法解决问题时效率也会不同，不同数据规模会影响算法效率。

此处是把二分法用于解决生活中的实际问题。利用二分法查看监控视频，即首先查看视频的中间处（下午2时），看丢失的货物是否还在。如果还在，就继续查看后半段视频的中间处（下午5时）；如果不在，就查看前半段视频的中间处（上午11时）。以此类推，不断缩小查看的范围。

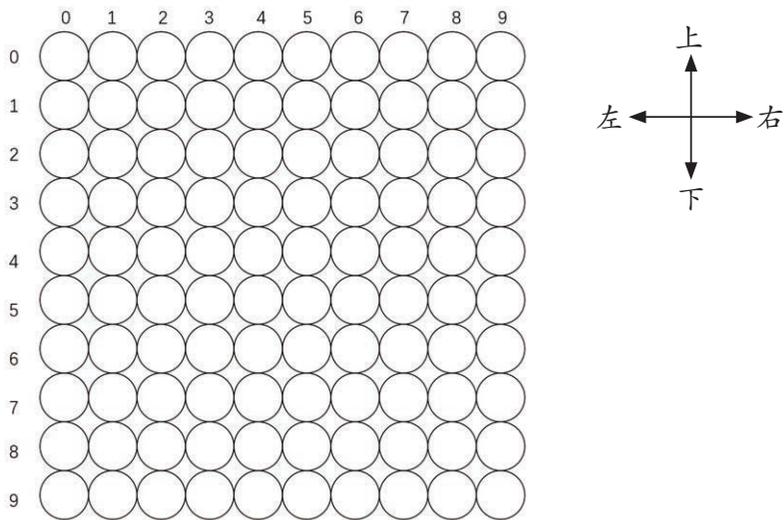
同样可以让学生明确查找的范围和问题的关键因素，进一步认识二分法在生活中的实际应用，体验学以致用成就感。

## 拓展与提升

1. 用程序玩猜数游戏。配套资源中有一个猜数游戏程序，打开并运行这个程序，猜一猜程序设定的目标数。

继续探究：一位同学尝试修改程序第一行中的目标数，再运行程序，让另一位同学来猜，进一步体会二分法的应用。

2. “猜一猜我在哪里”游戏。下图中的一个圆圈表示一个位置，一位同学想定一个位置并写下来，让另一位同学猜这个位置，每猜一次提示对方偏向于哪个方向，直到猜对为止。



猜位置游戏示意图

1. 供教师了解的程序提示。

游戏程序第一行的语句是设定要猜的数。在这个程序中，多分支结构位于循环结构的循环体中，用于判断是否继续执行。由于循环结构中包含 break 语句，使得程序可以在满足条件时跳出循环。

2. 这个游戏也是对二分法的实际应用，相当于在二维平面上的猜数游戏。游戏过程中，需要提示四个方向所发生的方向偏移。

每猜一次给一次提示，离目标位置会更进一步。

## 第 11 课

### 闰年平年我知道 (1)

#### 学习目标

1. 进一步学习分支结构, 体验多分支结构的应用, 感受用算法解决问题的过程。
2. 通过描述判断闰年与平年的算法, 知道算法的输入、判断处理中的条件以及输出。

这一课以判断闰年与平年为例, 学习在算法中使用多个条件进行判断。

#### 学习活动 1

##### 认识判断闰年的规则

有一个小朋友今年已经 11 岁了, 可是他只过了 2 次生日, 你知道这是为什么吗?

分析年历发现, 2 月 29 日出现的机会相对较少, 因为每四年才会有一次, 只有这一天出生的人, 生日次数才会较少, 也就是说这位小朋友是闰年的 2 月 29 日出生的。

2024年2月							^	v
一	二	三	四	五	六	日		
29 十九	30 二十	31 廿一	1 廿二	2 廿三	3 廿四	4 立春		
5 廿六	6 廿七	7 廿八	8 廿九	9 除夕	10 春节	11 初二		
12 初三	13 初四	14 初五	15 初六	16 初七	17 初八	18 初九		
19 雨水	20 十一	21 十二	22 十三	23 十四	24 元宵节	25 十六		
26 十七	27 十八	28 十九	29 二十	1 廿一	2 廿二	3 廿三		
4 廿四	5 惊蛰	6 廿六	7 廿七	8 妇女节	9 廿九	10 二月		

2024 年 2 月的日历

教学重难点:

1. 进一步认识多分支结构;
2. 描述判断闰年与平年的算法。

本课通过判断闰年与平年问题作为相关算法学习的切入点。

教学时可以通过提问引导学生思考。

问题 1: 哪一天出生, 生日天数会少呢? 为什么?

问题 2: 如何判断某年是闰年还是平年?

学生在数学学习中, 已经知道闰年与平年的天数区别。针对问题 1, 他们会回答是 2 月 29 日出生这一天生日会少。通过追问问题 2, 把学生引导到判断闰年与平年的问题上来。

大部分同学知道每四年会有一个闰年, 部分同学还知道“四年一闰, 百年不闰, 四百年又闰”的俗语, 但是具体的判断方法, 多数学生并不能明确表述。

闰年是有 366 天的特殊年份。闰年的二月有 29 天，而平年的 2 月只有 28 天。

那么，如何判断一个年份是闰年还是平年呢？

## 学习活动 2

### 描述判断闰年与平年的算法

下面用自然语言描述判断闰年与平年的算法。

1. 如果一个年份能被 4 整除，但不能被 100 整除，那么这一年就是闰年。
2. 如果一个年份能被 400 整除，那么这一年也是闰年。
3. 如果不满足以上条件，这一年就是平年。

把判断的条件组合在一起，进一步明确算法的描述。

如果“一个年份能被 4 整除”而且“不能被 100 整除”或者“一个年份能被 400 整除”，则结论为“是闰年”，否则结论为“是平年”。

这里，“而且”的优先级高于“或者”。

在算法描述中，为了明确优先顺序，还可以加上括号。

如果（“年份数值能被 4 整除”而且“年份数值不能被 100 整除”）或者“年份数值能被 400 整除”，则结论为“是闰年”，否则结论为“是平年”。

教师可以提供更多的年份，让学生判断哪些能被 4 整除，哪些能被 100 整除，哪些能被 400 整除。

通过对年份的判断，引导学生总结归纳判断条件。

可以分别对以下两个年份进行分析。

2020 年，只能被 4 整除，不能被 100 整除，这样的年份就是闰年，由于 2020 年不能被 100 整除，肯定就不能被 400 整除，所以判断条件确定为“如果一个年份能够被 4 整除，但不能被 100 整除，那么这一年就是闰年”。

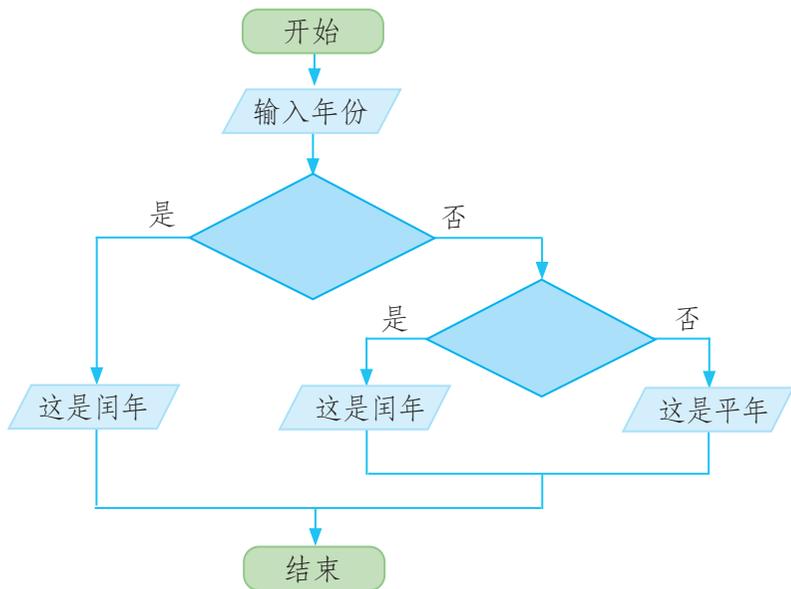
2000 年，能被 400 整除，事实上，这样的年份肯定也能被 4、100 整除，所以确定判断条件为：“如果一个年份能够被 400 整除，那么这一年也是闰年。”

引导学生观察和总结判断闰年与平年的条件，有了判断条件，判断过程就变得很简单：用年份的数值分别验证是否符合这两个条件，符合就是闰年，不符合就是平年。

加括号的描述，主要目的是让学生了解算法中存在逻辑关系。

下面用流程图描述判断闰年与平年的算法。

1. 补充流程图空白处的判断条件，梳理算法的执行过程。
2. 如果输入的年份分别是 1900、2008、2033、2400，讨论交流每个年份执行的判断路径和结果。
3. 为了实现判断闰年还是平年，流程图使用了（ ）结构。



算法的流程图

认真观察可以发现：

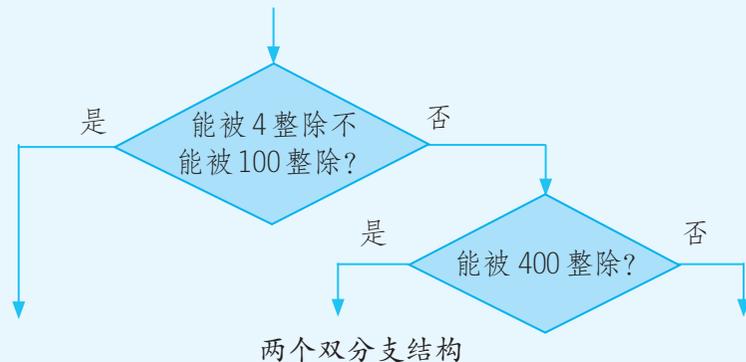
判断条件形成了两个双分支的结构。

结合补全流程图的方式确定算法中的关键因素，引导学生明确算法的执行路径。

这里要引导学生把语言分析与流程图相结合，理解判断条件后能够运用，提升学生逻辑思维和知识迁移的能力。

教学中，用流程图描述算法的方式更便于学生认识算法的执行过程，明确输入、处理和输出的基本流程。

当学生补全流程图后，可以让他们说一说：两个双分支结构和已经学习过的一个双分支结构或一个单分支结构的不同之处。



教学时，学生在理解判断条件的关系时可能会存在一定困难。例如，对“而且”“或者”的理解，对两个判断条件优先级的理解等。教学时要着重分析和讲解，引导学生交流讨论、互帮互学，最后教师归纳总结，给足学生思考和交流的时间，帮助学生循序渐进地认识判断条件之间的逻辑关系。

## 拓展与提升

### 1. 分支结构的实际应用。

小夏是五年级学生，已经满 11 岁，假期要与妈妈以及 8 岁的妹妹、5 岁的弟弟一起乘坐火车去看望爷爷奶奶。小夏和爸爸一起查看了火车购票的规定，发现针对儿童购票的情况，有下面的规定。

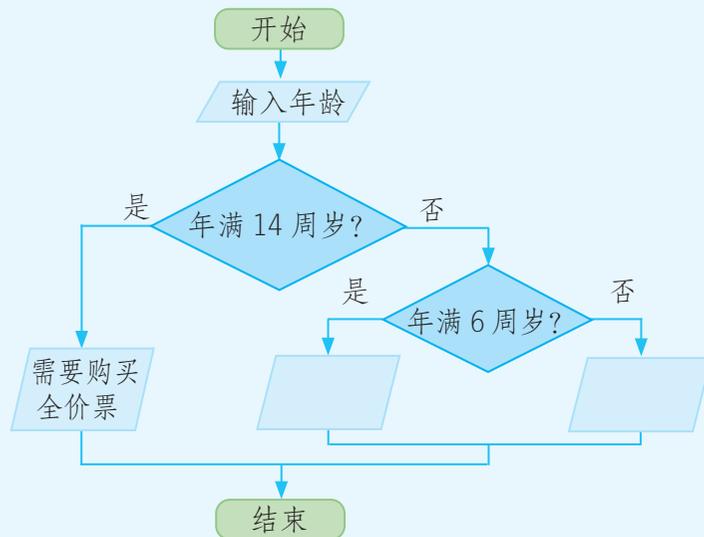
随同成年人乘车的儿童，年满 6 周岁且未满 14 周岁的应当购买儿童优惠票；年满 14 周岁，应当购买全价票。每一名持票成年人旅客可免费携带一名未满 6 周岁且不单独占用席位的儿童乘车。

请分析小夏一家的购票方案是什么？其中包含了哪些条件判断？

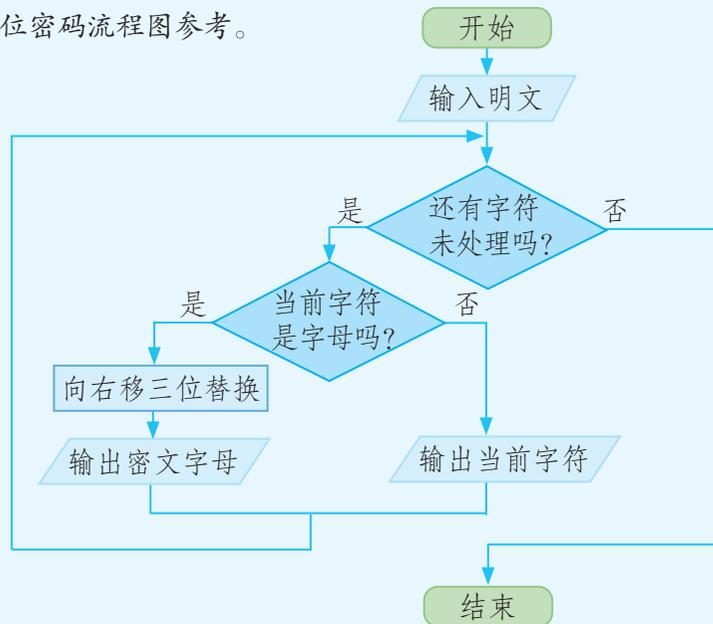
2. 在第 9 课的移位密码算法中，添加判断条件：“如果明文中出现非字母符号，则保持原样输出。”

请补充流程图，使算法在移位输出字母的同时可以原样输出其他符号。

### 1. 购买火车票流程图参考。



### 2. 移位密码流程图参考。



## 第 12 课

### 闰年平年我知道 (2)

#### 学习目标

1. 认识 Python 中的多分支结构语句，能够补全多分支结构的程序并运行，验证算法。
2. 初步了解利用多分支结构解决问题的过程，逐步培养解决问题的思维能力。

前面一课中，我们学习了判断闰年与平年的算法，你是不是很想用程序实现与验证这个算法呢？下面就一起来探究吧。

#### 学习活动 1

##### 认识多分支结构程序

在 Python 中，依据多个条件判断得到不同结果的程序结构就是多分支结构。例如，具有两个判断条件、两个双分支的嵌套结构就是多分支结构。通常用 `if …… elif …… else` 语句实现。下面是两个双分支结构的基本格式。

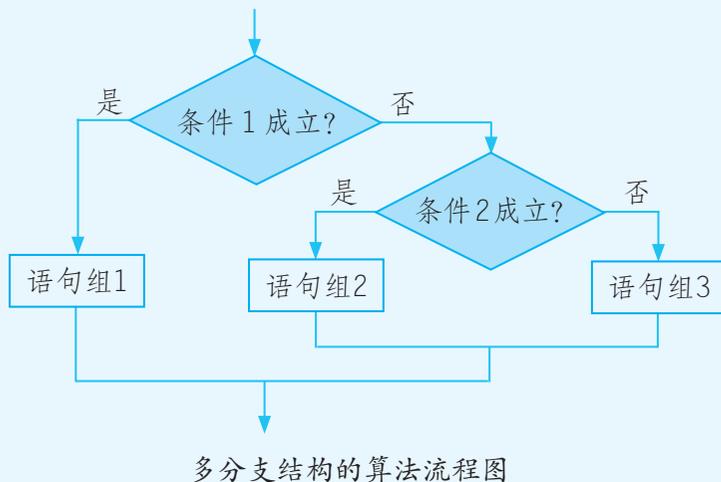
```
if 条件 1:  
    语句组 1  
elif 条件 2:  
    语句组 2  
else:  
    语句组 3
```

执行程序时，先判断条件 1 是否成立。如果成立，就转向其中的“是”分支并执行语句组 1；如果不成立，则转向“否”分支。继续判断条件 2，如果条件 2 成立，就转向其中的“是”分支并执行语句组 2；否则，转向其中的“否”分支并执行语句组 3。

教学重难点：

1. 认识实现多分支结构的程序语句；
2. 明确分支结构中的判断条件。

本课通过观察、运行程序，引导学生认识多分支结构及其用法，并能够修改简单的多分支结构程序。同时，可以引导学生思考其他类似的多分支结构问题，培养他们的逻辑思维能力和用程序实现与验证算法的意识。



在 Python 中，`elif` 可理解为 `else if` 的缩写，用于在不满足一个 `if` 条件的前提下检查另一个条件。

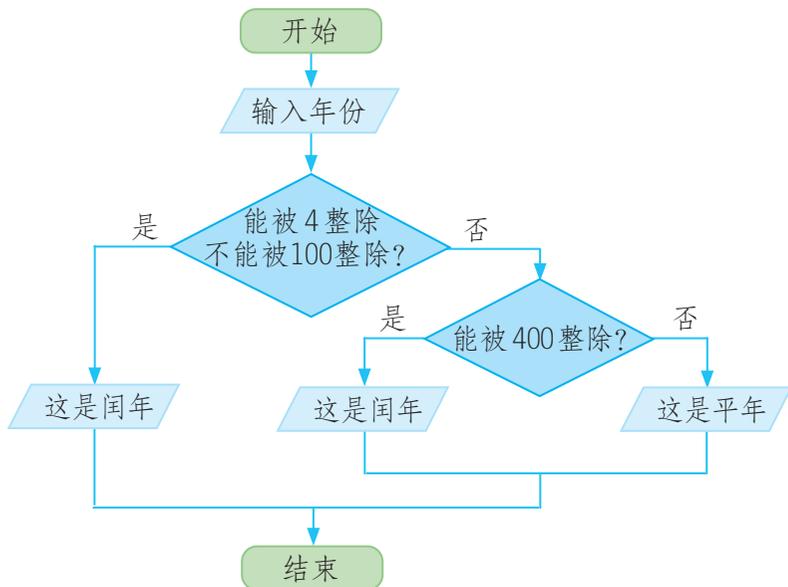
下面以“根据气温给出适当穿衣建议”为例，编写多分支结构程序。

```
temperature = float(input('请输入气温: '))
if temperature < 15:
    print('今天有点冷, 要穿厚一些的衣服')
elif 15 <= temperature <= 30:
    print('今天气候宜人, 适合穿春秋装')
else:
    print('今天有点热, 要穿薄一些的衣服并适当防晒')
```

## 学习活动2

### 验证判断闰年与平年的算法

第1步：回顾前面所学知识，进一步认识下面的流程图。



算法的流程图

利用多分支结构，可以根据不同的情况进行更多的判断，使程序层次清晰。

引导学生观察与分析，根据不同温度范围给出了不同的建议。

如果温度低于  $15^{\circ}\text{C}$ ，那么输出提示“天有点冷，要穿厚一些的衣服”。

如果温度在  $15^{\circ}\text{C}$  至  $30^{\circ}\text{C}$  之间，则输出提示“今天气候宜人，适合穿春秋装”。

如果温度大于等于  $30^{\circ}\text{C}$ ，则输出提示“今天有点热，要穿薄一些的衣服并适当防晒”。

教学时，先引导学生进一步认识流程图，再让学生依据流程图补充相关的程序代码，了解用程序语句实现算法的方法。

此外，可以引导学生探讨与年份相关的其他问题，如特殊年份的计算、历法的演变等，以拓展他们的知识。还可以进一步启发学生把所学知识和生活实际相联系，巩固对分支结构的理解，并应用到实际问题之中。

第 2 步：利用多分支结构程序语句，补全用红色文字描述的程序代码。

```
# 获取输入的年份
year = int(input(' 请输入一个年份: '))
# 判断是闰年还是平年
如果 year%4 == 0 and year%100 != 0:
    # 年份除以 4 取余等于 0 并且年份除以 100 取余不等于 0
    print (year, ' 这是闰年 ')
又如果 年份除以 400 取余等于 0
    print (year, ' 这是闰年 ')
否则
    print (year, ' 这是平年 ')
```

第 3 步：修改完成后，保存程序文件。

第 4 步：运行程序，输入不同的年份，观察执行结果。

```
year = int(input(' 请输入一个年份: '))
if year%4 == 0 and year%100!=0:
    print(year, ' 这是闰年 ')
elif year%400 == 0:
    print(year, ' 这是闰年 ')
else:
    print(year, ' 这是平年 ')
```

比较运算符说明表

比较运算符	说 明	举 例
<	小于	1<2, True; 2<1, False
<=	小于等于	1<=2, True; 2<=1, False
>	大于	2>1, True; 1>2, False
>=	大于等于	3>=2, True; 2>=3, False
==	等于	2==2, True; 'A'=='B', False
!=	不等于	2!=2, False; 2!=3, True

year%4 表示 year 除以 4 的余数，% 是取余运算符。

and 是“与”运算符，表示要同时满足两个条件，即年份能被 4 整除、不能被 100 整除。

## 拓展与提升

多分支结构的现实应用。

阅读下面有关购买火车票的多分支结构程序，补充红色文字的语句，保存并运行程序，输入不同的年龄，观察执行结果。

```
nl = int(input('请输入乘车人的年龄：'))
```

```
如果 nl >= 14:                # 判断是否年满 14 周岁
```

```
    print(nl, '岁，请购买全价票')
```

```
elif 年龄大于等于 6 且小于 14: # 判断是否年满 6 周岁且未满 14 周岁
```

```
    print(nl, '岁，请购买儿童优惠票')
```

```
否则：
```

```
    print(nl, '岁，由持票成年人免费携带')
```

参考程序如下。

```
nl = int(input('请输入乘车人的年龄：'))
```

```
if nl >= 14:
```

```
    print(nl, '岁，请购买全价票')
```

```
elif 6 <= nl < 14:
```

```
    print(nl, '岁，请购买儿童优惠票')
```

```
else:
```

```
    print(nl, '岁，由持票成年人免费携带')
```

教学时，可以进一步引导学生思考其他与购票有关的条件判断问题，如公交、地铁的分段计价，特殊身份的优惠条件等。

### 【引言】

算法是培养计算思维的核心要素之一，主要表现在三个方面：一是能够清晰地理解问题解决的规则，认识问题的起点、边界和限定范围；二是将问题进行分解，即能够将一个复杂的问题分解成若干子问题，缩小问题的范围；三是能清楚地分析问题解决方法的优势，设计和构建操作步骤少、效率高的算法。

在信息科技中，算法注重问题求解的有效性。首先要能给出结果而且是正确的结果；其次要关心效率，能在一定时间内快速完成任务，越快越好，且占用的存储资源越少越好等。

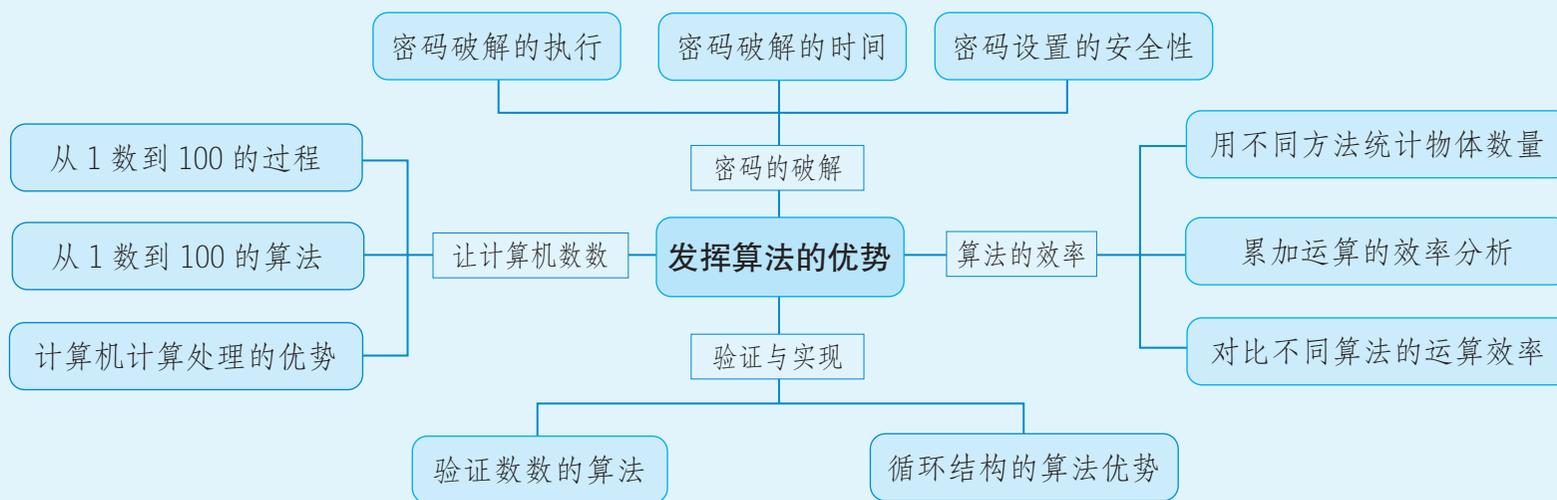
通过学习算法和程序实现，学生可以体验解决问题的过程，培养计算思维能力，这是本模块的主要目标。算法是信息科技课程的一个核心概念，也是计算机科学的核心内容。在信息社会中，对身边算法的认识与了解可以更好地提高学习和生活质量。

在前面的学习中，学生认识了算法的基本流程（输入—处理—输出），能利用自然语言、流程图等方式描述求解简单问题的算法，知道算法的三种基本控制结构是顺序结构、分支结构和循环结构。本单元将进一步引领学生认识、理解计算机解决问题的基本方法，利用算法中的循环结构来解决实际问题，在体验编程让计算机执行的过程中，感受算法实现的优势和意义。本模块强调用计算思维解决问题的独特价值，通过本单元的学习，学生能够进一步明确算法执行的正确性与效率要求，并能够进行讨论与辨析。

循环结构的基本思想是反复执行，即利用计算机的高速运算和逻辑控制的特性，重复执行指定的步骤，从而完成相应的处理任务。当然，这种反复执行不是简单机械的重复，而是每次执行都有新的内容，会产生新的结果。

在算法设计中，循环结构可以用于解决各种重复计算或重复处理的问题，如排序、搜索、迭代等。通过合理设计循环结构和选择合适的循环终止条件，可以使算法效率更高。同时，循环结构也是算法设计与优化的重要方法之一，通过优化循环结构可以提高算法的稳定性。

## 【内容结构】



## 【教学实施条件】

1. 本单元建议在具备运行 Python 环境的信息科技实验室里教学，能够运行配套资源中的游戏、程序等。
2. 配备教学演示与互动的设备，准备用于教学的图片、流程图、程序、学习任务、实物等。

## 【学习目标】

1. 认识循环结构和计算机工作特点之间的联系，体会利用计算机的快速运算能力来解决问题的方法。
2. 进一步认识 Python 中的 while 循环结构，并能运行已有程序，验证指定的算法。
3. 通过对不同的算法进行分析，帮助学生认识算法效率，知道用不同算法解决同一问题时存在不同的效率。
4. 了解简单密码破解的次数，知道执行次数与破解时间的关系。
5. 体验利用循环结构解决实际问题的过程与方法。

## 【评价建议】

下表可作为单元学习评价的参考。

评价内容	学生自评	教师评价
结合计算机解决问题的方法初步分析问题	☆☆☆☆☆	☆☆☆☆☆
知道循环结构与计算机工作特点的关系	☆☆☆☆☆	☆☆☆☆☆
了解利用计算机的快速运算能力解决问题的方法	☆☆☆☆☆	☆☆☆☆☆
知道计算机解决问题的时间效率，并能估算相应的时间	☆☆☆☆☆	☆☆☆☆☆
知道用不同算法解决同一问题时存在不同的效率	☆☆☆☆☆	☆☆☆☆☆
了解密码设定的原则，并具有一定的安全性辨别能力	☆☆☆☆☆	☆☆☆☆☆
了解简单密码破解的次数，知道执行次数与所需时间的关系	☆☆☆☆☆	☆☆☆☆☆
主动思考，与同学协作共进，积极实践探索	☆☆☆☆☆	☆☆☆☆☆

## 第 13 课 让计算机数数

### 学习目标

1. 通过描述让计算机从 1 数到 100 的算法，了解用循环结构解决问题时的重复过程。
2. 了解循环结构与计算机工作特点的关系，体会利用计算机的快速运算能力来解决问题的方法。

这一课以让计算机“从 1 数到 100”为例，学习较简单的循环结构。



从 1 数到 100

### 学习活动 1

#### 描述“从 1 数到 100”的算法

##### 1. 缩小问题规模

为了便于理解，先把问题规模缩小，分析从 1 数到 10 的过程。

思考以下问题。

- (1) 我们数数的过程是什么样的？
- (2) 数完一个数后，怎么数下一个数？
- (3) 如何让计算机数数？

人数数的过程很简单，一个数一个数地数下去。如果这个任务交给计算机来完成，也需要进行同样的过程。那么，让计算机直接数数不好体现，可以选择用输出这些数来表示。

教学重难点：

1. 循环结构中重复执行的操作和数值变化的过程；
2. 描述让计算机从 1 数到 100 的算法。

学生在前面的学习中，接触过循环结构在生活情境中的体现，通过简单的文字描述，知道循环的过程就是不断重复执行指定操作的过程。本课在这个认知基础上，通过用循环结构解决实际问题来分析算法执行的次数。

数数的过程很简单，可以直接用流程图表示。

引导学生观察并说出想法。

1. 除了输出的数不一样，其他都是一样的，是重复的操作。
2. 从中发现的规律是每个数依次加 1。
3. 可以把重复的操作通过循环结构来描述。

如何描述是一个难点。建议进行细致拆分，逐步引导。一是把重复的操作提取出来，学生能说出每次输出的操作相同，只是数值不同；二是根据什么条件来决定是否继续输出数值。

可以由学生先用自然语言描述这个过程，教师进行修正补充。



用流程图描述

## 2. 回顾学习过的循环结构

循环结构是一种基本的控制结构，用于重复执行循环体中的语句，直到满足循环条件为止。循环结构可以帮助我们处理那些需要重复执行的任务。

循环体指循环结构中需要重复执行的语句，可以包含一个或多个语句。

执行过程可以描述如下。

- (1) 如果循环条件成立，则做一遍循环体指定的操作；
- (2) 再检查循环条件，如果成立，则再做一遍循环体指定的操作；
- (3) 以此类推，重复执行，直到循环条件不再成立。

## 3. 还原问题，描述算法

要让计算机“从1数到100”，如果是“第一次输出1、第二次输出2、第三次输出3……”那么每次操作都不一样。如果画出流程图，就有100个操作框，在一张纸上根本画不下，事实上也没必要画出来。

要描述“从1数到100”的操作，每次操作都是相同的，只是数值不同，这样，可以用循环结构来描述。

初始时：这个数为1

结束时：这个数为101

第1次：输出这个数，然后把这个数加1。

(判断这个数是否小于等于100，此时条件满足，继续循环)

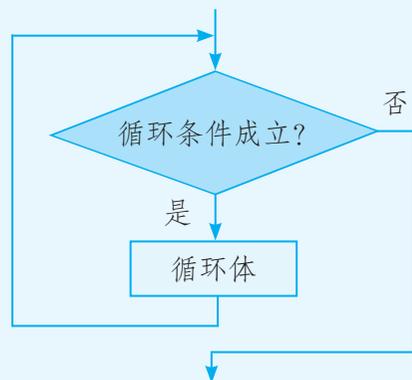
第2次：输出这个数，然后把这个数加1。

(判断这个数是否小于等于100，此时条件仍然满足，继续循环)

……

第100次：输出这个数，然后把这个数加1。

(判断这个数是否小于等于100，此时条件不满足，结束循环)

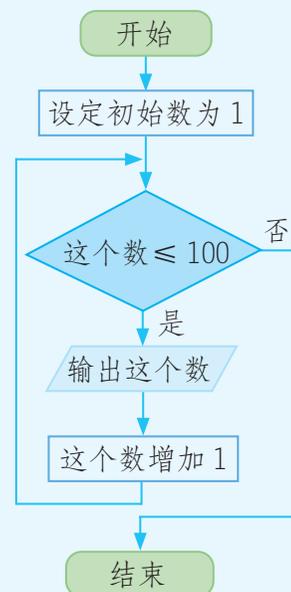


循环结构的算法流程图

在算法结构中，循环结构是最能够体现计算机解决问题高效性的内容。为此，要引导学生在学习中学会分析问题，从具体问题中抽象出问题的本质，并利用循环结构解决问题。

算法思路基本清晰之后，可以用流程图描述算法。

这里不必强调变量的概念，重点放在数值加1后变为新的数值。其实有了分支结构作为基础，学生认识循环结构并不难。只要把需要重复执行的语句（循环体）放入分支结构中，并在分支判断之后决定是否执行这些语句，再继续返回判断条件进行判断，直到条件满足。



算法的流程图

进一步探究：描述让计算机“从 1 数到 100”，只数其中的奇数。

只需稍微修改前面的算法就可以实现。

起点：这个数为 1

结束：这个数为 101

第 1 次：输出这个数，然后把这个数加 2。

（判断这个数是否小于 100，此时条件满足，继续循环）

第 2 次：输出这个数，然后把这个数加 2。

（判断这个数是否小于 100，此时条件仍然满足，就继续循环）

……

第 50 次：输出这个数，然后把这个数加 2。

（判断这个数是否小于 100，此时条件不满足，结束循环）

## 学习活动 2

### 验证“从 1 数到 100”的算法

第 1 步：在 Python 中打开并运行以下参考程序，查看运行结果，了解程序执行的过程，观察 while 循环结构。

#### # 设置计数变量的初值

```
c = 1
```

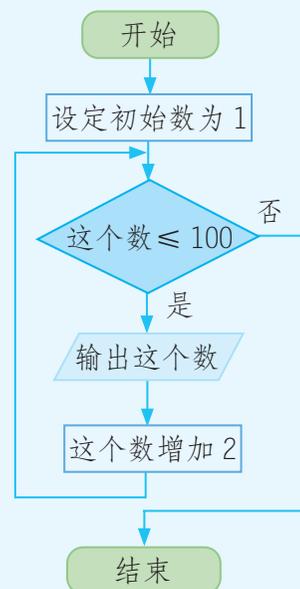
#### # 用循环控制从 1 数到 100

```
while c <= 100:
```

```
    print(c)    # 输出当前的数
```

```
    c = c + 1   # 计数变量加 1
```

教学时，可以依据实际教学时间，对从 1 数到 100 这个问题进行变化。例如，从逐一数数，到只数奇数或偶数，或是某个数的倍数等。考查学生是否知道哪些条件发生了变化，是怎么发生变化的，再修改流程图。



算法的流程图

通过观察、运行、修改程序，引导学生了解循环结构的用法。

循环的三要素：初始化、判断条件和循环体。

需要提醒学生的是：while 表达式后面的冒号不能省略，循环体中语句前面的缩进不能省略。

如果学生理解 while 循环结构存在困难，可以缩小数值范围。例如，可用下面的程序，列表分析数据的变化过程。

```
c = 1
while c <= 5:
    print(c)
    c = c + 1
print(' 输出结束 ')
```

在 Python 中，循环结构可以用 while 语句来实现。

语句的基本格式如下。

```
while 循环条件:  
    循环体
```

当循环条件成立时，执行循环体的操作；当循环条件不成立时，退出循环。这个结构不需要统计要运行多少次循环，只需要使用判断条件来确定什么时候停止循环。

程序中的语句  $c=c+1$ ，是一条赋值语句，将变量  $c$  的数值加 1，然后把计算结果再存储到  $c$  中，从而实现  $c$  值的变化，达成计数器的功能。

正是因为有这条语句，才使得  $c$  的数值不断增大，直到循环条件不成立，就退出循环。

第 2 步：修改程序。

输入一个自然数  $n$ ，输出 1 至  $n$  之间的所有偶数。

```
n = int(input('请输入自然数 n: '))  
c = 2  
while c <= n:  
    print(c)  
    c = c + 2
```

进一步探究：利用取余运算，输出 1 到  $n$  之间所有能被 5 整除的数。

c 的值	判断条件 $c \leq 5$	print(c)	c 值的变化 $c=c+1$
c=1	1<=5 成立	1	2
c=2	2<=5 成立	2	3
c=3	3<=5 成立	3	4
c=4	4<=5 成立	4	5
c=5	5<=5 成立	5	6
c=6	6<=5 不成立		

程序的难点在于赋值语句： $c=c+1$ 。

学生在此处容易产生困惑，认为左右两侧并不相等。一定要强调这不是等式，而是赋值语句，需要优先计算右侧的算式，再把计算结果赋值给左侧的变量。

利用变量能够多次被赋值的特点，实现计数器的功能。建议将 Python 中的赋值号“=”和等号“==”进行对比讲解。

```
n = int(input('请输入自然数 n: '))  
c = 1  
while c <= n:  
    if c % 5 == 0:  
        print(c)  
    c = c + 1
```

### 学习活动3

#### 感受计算机运算处理的优势

用计算机处理问题时的最大优势，就是其快速的运算能力。

第1步：启动 Python，打开它的窗口。

第2步：打开配套资源中的“数数.py”程序，观察并运行，体会编程进行计算的速度。

1秒大约进行\_\_\_\_\_次数数（加法运算）。

第3步：打开配套资源中的“数数并输出.py”程序，观察并运行。

1秒大约进行\_\_\_\_\_次数数（加法运算）和\_\_\_\_\_次输出。

通过简单的对比，可以发现，编程让计算机完成加法运算时，可以充分发挥算法的优势，提高解决问题的效率。但相对于加法运算来说，输出操作也要占用不少的时间。

利用程序来让计算机完成数数时，即使是一台普通的个人计算机，1秒也可以枚举几百万个数。这个数量可能与计算机的硬件配置和软件相关，但同一类型的计算机，数量级基本是一致的。

而人类进行翻书页、数物品个数、排队报数等操作时，每秒最多也就能翻几页书、数几个物体、报几个数。

利用循环结构从1数到100，从本质上体现了计算机解决问题的方法，即数出一定范围内的数值再对这些数值进行处理。用计算机解决这个问题时，需要确定的就是要枚举的数值范围。

对于人来说，从1数到100是一件比较无趣的事情，很多人已经不愿意做这样简单而且没有创造性的任务。但是，对于计算机来说，这类需要重复进行的计算工作，不仅可以快速地进行计算，而且还能不厌其烦地稳定工作。

在这个对比活动中，可以用科学探究的方式，鼓励学生通过编程对问题进行实践探索，体会用计算机解决问题的方法和优势。

## 拓展与提升

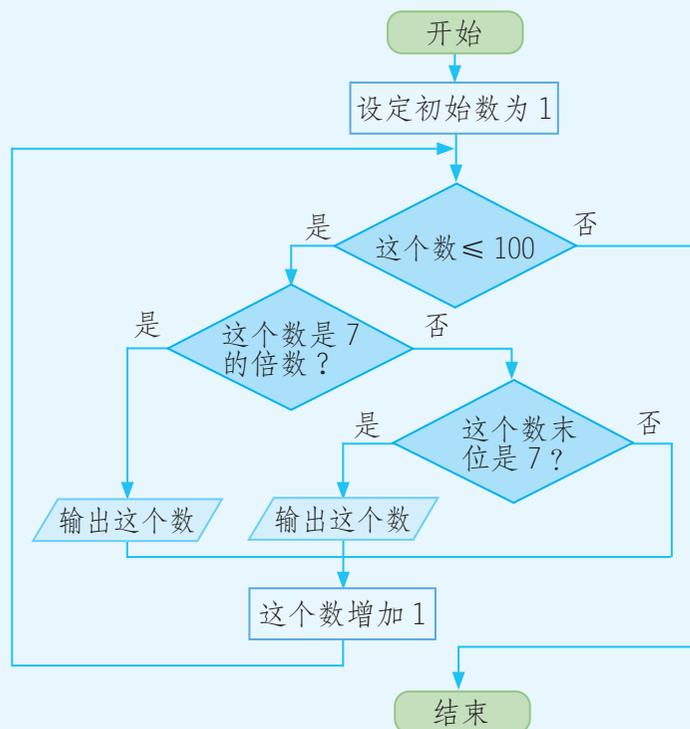
1. 如果知道起始数  $b$  和终止数  $e$ ，编程找出  $b$  到  $e$  之间的所有奇数。
2. 有一个“逢七必过”的游戏，游戏规则如下。
  - (1) 游戏参与者按顺序排好队。
  - (2) 从 1 依次报数到 100，如果是 7 的倍数或末位数是 7，就报“过”。
  - (3) 如果违规了就要被“罚”。

尝试玩一玩这个游戏并思考：游戏中的判断条件是什么？如何用流程图描述算法？

1. 解决这个问题的关键是判断  $b$  是奇数还是偶数。如果起始值  $b$  是偶数，那么初始值先加 1，然后每次循环时加 2；如果起始值  $b$  是奇数，那么把  $b$  每次加 2。直到等于或大于终止值  $e$  时结束循环。

2. 这个游戏实际上是判断一个数是否为 7 的倍数和末位数是否为 7，引导学生了解分支结构与循环结构的组合应用。

在遍历 1 至 100 之间报数的过程中，利用算法让计算机执行时，可以只把符合条件的数输出显示出来。



算法的流程图

## 第 14 课 算法效率比一比

### 学习目标

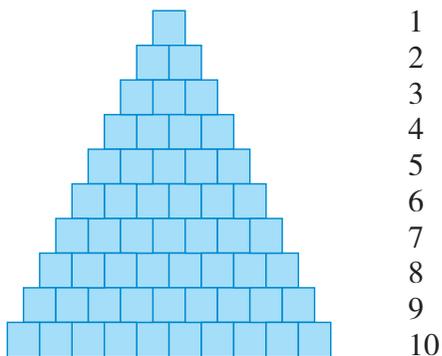
1. 知道解决同一个问题可以有不同的算法，不同的算法具有不同的效率。
2. 通过实例比较和算法分析，了解算法执行的关键步骤和执行次数，体会算法存在的效率差异。

这一课以简单的累加运算为例，了解用不同算法解决同一问题的过程，认识其中存在的效率差异。

### 学习活动 1

#### 用不同方法统计物体数量

一堆物体摆放如下图所示，要统计有多少个，你能想到哪些方法？



一堆摆放好的物体

第一种算法：把物体逐层进行累加。

教学重难点：

1. 解决同一个问题可以有不同的算法，不同的算法具有不同的效率；
2. 算法效率的重要性，理解更少的步骤和更快的解决速度通常意味着更高的效率。

引导学生观察发现：

通过数一数每层的物体个数，发现其中的变化规律。

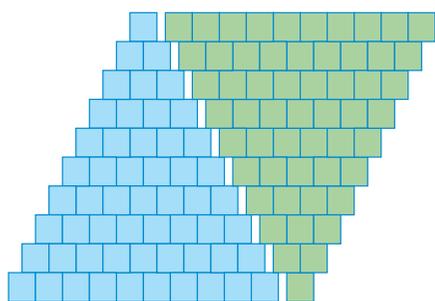
物体共 10 层，从上到下，每层分别是 1 至 10 个。

第一种算法：

$$1 + 2 + 3 + 4 + 5 + 6 + 7 + 8 + 9 + 10 = 55$$

第二种算法：观察图形，发现可以用前面学习过的方法，即利用正反放置的两个梯形组成平行四边形，通过求平行四边形中物体的个数来计算。

$$\begin{aligned} s &= (1+n) * n / 2 \\ &= (1+10) * 10 / 2 \\ &= 55 \end{aligned}$$



两个梯形组成的平行四边形

通过比较发现：算法 1 简单直观，易于理解，算法 2 所用的步数较少，计算起来更快。

从前面的分析可知，通过求“ $1+2+3+\dots+10$ ”的两种不同算法，说明解决同一个问题时，不同的算法会有不同的步骤，也就可能存在不同的效率。

## 学习活动 2

### 累加运算的效率分析

通常，用计算机解决问题时会用以下两种方法来比较算法的效率。

一是比较算法运行所需要的时间。

二是比较算法运行时所需的步数或者占用的资源。

下面主要从时间上来进行分析。

大家听过数学家高斯小时候计算“ $1+2+3+\dots+100$ ”的故事吧？高斯使用第二种算法很快给出了答案，比其他孩子的速度都快。

为什么会这样呢？

第二种算法：

$$\begin{aligned} \text{平行四边形中物体的个数} &= \text{每层个数} \times \text{层数} \\ &= (1+10) \times 10 = 110 \text{ 个} \end{aligned}$$

$$\begin{aligned} \text{因此，梯形中物体个数} &= \text{平行四边形中物体的个数} \div 2 \\ &= 110 \div 2 = 55 \text{ 个} \end{aligned}$$

进一步引导学生归纳：依据上述计算方法，就可以得到求解一组连续自然数累加之和的公式。

$$\text{累加之和} = (\text{第一个数} + \text{最后一个数}) \times \text{数的总个数} \div 2$$

例如，自然数从 1 到  $n$  的累加之和可以表示为：

$$s = (1+n) * n / 2$$

衡量计算机在运行程序时的效率，没有统一的标准，通常选择只比较其中的一个方面。

实际教学时，从时间上来进行分析、比较更易于学生理解。

我们先来做一个“合理假设”：如果做1次加法用时1秒、做1次乘法用时10秒、做1次除法用时15秒。

用第一种算法计算：

需要计算约99次加法，这样即使每次加法只用1秒，而且每次中间相加的结果都正确，最终也需要大约99秒的时间才能计算出结果。

用第二种算法来计算：

只需要1次加法（即 $100 + 1$ ）、1次乘法（即 $101 \times 100$ ）和1次除法（即除以2），需要约 $1+10+15 = 26$ 秒。

因此，单从计算步骤和时间上看，第二种算法似乎比第一种更高效。

这就是在“合理假设”的前提下，高斯比其他同学算得快的一种解释。

但是，问题并没有那么简单。

因为做乘法和除法时，通常比做加法需要更长时间。

因此，如果以上假设并不成立，比如，如果做1次乘法或1次除法都需要50秒，那么用第二种算法所需的时间就会变成 $1 + 50 + 50 = 101$ 秒。

通过上述分析可知，从用算法解决问题的角度看，要准确地比较不同算法的效率，往往比我们预想的要难很多。通常需要从数据量、步骤多少、所需时间等方面综合考虑。

在设计算法用计算机解决问题时也是如此，通常需要经过多次的比较、实验与探索来获得结论。

进一步比较分析，让学生知道，要客观地认识事物规律、获得更准确的结论，往往需要多角度、多层面地分析解决问题的方法。

以下供教师教学拓展，进一步理解编程让计算机求解累加运算的效率问题。

求和算法应该适用于任何一个自然数，因此可以直接用累加的方式，定义一个更通用的Python函数sum(n)。

sum1表示使用第一种算法求和，即逐个进行累加。sum2表示使用第二种算法求和，即用求累加和的公式。

# 用逐个累加求解  $1 + 2 + \dots + n$

```
def sum1(n):  
    final_sum = 0  
    for x in range(1, n+1):  
        final_sum += x  
    return final_sum
```

# 用公式求解  $1 + 2 + \dots + n$

```
def sum2(n):  
    return (n*(n+1))/2
```

### 学习活动3

#### 感受不同算法的运算效率

解决同一个问题通常可以用不同的算法，选择不同算法并编程实现后，程序一般会在运算速度、计算精度等方面有不同的表现。

下面通过用程序验证上述累加运算的两种算法，体会算法的效率差异以及不同程序实现引起的差异。

“累加 1.py”程序是用算式直接累加与用公式累加的对比。

“累加 2.py”程序是用循环结构实现累加与用公式累加的对比。

操作步骤如下。

第 1 步：打开配套资源中的“累加 1.py”程序，运行这个程序。

第 2 步：输入要重复执行的次数，观察运行结果。例如，分别输入 500、1 000、10 000、100 000 等，对比两种算法所用的时间。

第 3 步：打开配套资源中的“累加 2.py”程序，运行这个程序。

第 4 步：输入要重复执行的次数，观察运行结果。例如，同样分别输入 500、1 000、10 000、100 000 等。

第 5 步：尝试用更多更大的数进行反复实验。这样经由多次数值实验得出的结论会更加趋于稳定，也更加可靠。

第 6 步：依据运行结果，对算法与程序实现的效率进行总结。

在“累加 1.py”程序中，即使执行 10 000 次的累加运算，所需时间仍然非常少，甚至人已经感觉不到所需时间，主要是因为现在的计算机处理速度非常快，只有当数据量或执行次数达到一定规模时，程序才能捕捉到运算时间，操作者才能感受到细微的差异。

然而，不同算法的微小时间差，在数据量足够大时就会产生更大的时间差。因此，在大规模数据处理时，算法的选择与应用非常关键，如芯片制造、航空航天轨道控制等。

在“累加 2.py”程序中，与直接用算式累加相比，用循环结构进行累加要占用一些时间。也就是说，编程解决问题时，计算机所进行的任何操作都需要时间，在分析算法效率时要考虑这些因素。

注意引导学生用更多更大的数反复进行实验，体会经由多次数值实验得出结论的过程。

此外，运行程序获得的结果与所使用的计算机密切相关，甚至同一台计算机每次运行也会有差异，但结论反转的可能性很小。

## 拓展与提升

圆周率作为数学中的一个重要常数，其更多位数的精确值求解一直是数学家们所追求的。我国南北朝时期的数学家祖冲之，经过刻苦钻研和反复演算，推算出圆周率  $\pi$  的值在 3.1415926 和 3.1415927 之间，这一结果在当时已经非常精确。祖冲之通过其卓越的数学才能和不懈的努力，为数学的发展作出了重要贡献。

配套资源中有两个计算圆周率的程序，打开这两个程序并运行，对比计算圆周率的效率。

提示：两个程序分别采用了两种不同的算法。

算法 1：17 世纪，有学者找到了一种计算圆周率的方法。根据这个方法，只要计算足够多的数据项，就可以获得圆周率的近似值。

算法 2：很多年后，又有学者提出了另一种计算圆周率的方法。同样，只要计算足够多的数据项，也可以得到圆周率的近似值。

圆周率指圆的周长与直径之比，圆周率是精确计算圆的周长、面积和球的体积等的关键值。

例如，据古巴比伦石匾上的记载，圆周率被近似为 3.125。后来，随着数学的发展，人们逐渐认识到圆周率是一个无理数，即一个无限不循环的小数。

$$\text{算法 1 的公式} \quad \pi = 4 \times \left( \frac{1}{1} - \frac{1}{3} + \frac{1}{5} - \frac{1}{7} + \dots \right)$$

$$\text{算法 2 的公式} \quad \pi = 16 \times \left( \frac{1}{1 \times 5^1} - \frac{1}{3 \times 5^3} + \frac{1}{5 \times 5^5} \dots \right) - 4 \times \left( \frac{1}{1 \times 239^1} - \frac{1}{3 \times 239^3} + \frac{1}{5 \times 239^5} \dots \right)$$

以上公式不要求学生理解，只需知道是两种不同算法即可。

根据算法编写程序，程序的运行效率跟算法直接相关，如果一个问题能找到适合计算机使用的高效算法，那么问题就很容易用计算机来解决。

运行第一个程序时需要的时间稍长，计算出来的圆周率精确到了 3.1415 左右。运行第二个程序时几乎可以立刻得到计算结果，计算出来的圆周率的精确位数也更多。因此，依据算法 2 编写的程序，所需计算时间更短、计算结果更精确，效率更高。实际上，第二个程序中的公式每多循环一次，通常可以多得到一位精确数字。

教学时，不同计算机的运算效率也会不同。教师要提前运行相关程序，必要时根据所用计算机的运算能力，适当修改循环次数。

## 第 15 课 简单密码易破解

### 学习目标

1. 通过分析密码设定规则，计算破解密码所需的重复判断次数，估算破解密码的大致时间。
2. 进一步认识循环结构，体会利用遍历法解决问题的基本思路，了解密码设置的安全性。

在日常生活中，有许多地方都需要设置密码，密码保护着我们的个人隐私和信息安全。

问题思考：

1. 你知道哪些情况需要设置密码？
2. 你设定的密码是否安全？是否容易被别人猜到？

- 应用程序的登录密码
- 计算机的开机密码
- 手机锁屏密码
- 小区门禁密码
- 行李箱密码
- 家门的智能锁密码

日常生活中的各种密码，位数要求不同，组合形式也不一样。

教学重难点：

1. 计算破解密码所需的重复判断次数；
2. 在一定规则下，估算破解指定密码的时间。

这一课重在算出破解密码所需的循环次数，不涉及具体的破解算法和破解方式（需要多重循环，对于学生来说有一定难度）。

密码长度：四位、六位。

密码类型：数字、数字 + 字母、数字 + 字母 + 符号。

在日常生活中，学生或多或少接触或使用过密码。教学时，可以依据学生对“问题思考”的具体回答，引导学生进一步讨论：密码有哪些应用场合？教师事先预测学生可能的回答，并列举出有代表性的答案。

需要注意的是，教学过程中要不断引导学生树立安全意识，同时认识到：破解密码只限于理论学习和技术探讨，而不可以成为实际应用，不能侵犯他人的隐私信息。

## 学习活动 1

### 探究密码的破解次数

#### 任务 1：四位数字密码的破解次数

小夏家里有一个四位的数字密码锁，家里人忘记了密码，请你想办法帮忙找到密码，思考最多需要尝试多少次？



一个四位的数字密码锁

第 1 步：确定四位数字密码的组合方式。

四位数字密码的组合方式，可以从 0（即 0000）至 9999。

第 2 步：计算需要尝试的次数。

每一位数字都是 0 至 9，每一位有 10 种可能性。四位数字的组合就是：

$10 \times 10 \times 10 \times 10 = 10\,000$  种。

如果用循环结构来枚举，每种情况都需要尝试，最多需要进行 10 000 次，也就是要循环 10 000 次才能找到正确密码。

任务 2：六位“数字 + 字母”密码的破解次数

第 1 步：确定六位“数字 + 字母”密码的组合方式。

每位包括：数字 0 至 9，共 10 种；大写字母 A 至 Z，共 26 种；

小写字母 a 至 z，共 26 种。

第 2 步：计算一共有多少种组合，即所需的破解次数。

每位有  $10 + 26 + 26 = 62$  种，需要的次数是六个 62 相乘：

$62 \times 62 \times 62 \times 62 \times 62 \times 62 = 56\,800\,235\,584$ （次）（也就是约 568 亿次）

破解密码是一个较复杂的问题。组成密码的字符可以包括多种类型。教学时可以将问题分解，先从只有数字的密码开始，把解决问题的思路厘清，再加入字母或符号的组合。从简单到复杂，体现逐步深入的学习过程。

这里先引导学生讨论如何尝试。学生一般能说出一个一个地尝试，直到找出正确的密码。那么逐个尝试的过程就是重复的，再进一步启发，学生立刻就会想到用循环结构来解决。

对于本任务，可以设计以下问题让学生思考。

问题 1：四位数字密码的范围是什么？

问题 2：一共有多少种组合，如何计算？

枚举法的基本思想是逐一列举指定范围内所有可能的情况，根据条件进行检验从而找到可能的答案。枚举法是常见的算法之一，如用列表写出所有可能，学生对它的基本思想并不陌生，这里主要是要引导学生感受利用计算机程序实现算法的学科优势。

学生在四年级时就学过超过一亿的大数，理解大数应该没有难度，教学时不必占用更多课堂时间进行介绍。

进一步探究：如果是“数字+字母+符号”的六位密码，计算所需的破解次数。

提示：字母包含大小写字母，计算机标准键盘上常见符号约有 32 种。

## 学习活动 2

### 对比不同密码的破解时间

运行名为“破解时间估算.py”的参考程序，记录并对比不同密码组合所需的破解时间。

破解六位数字密码大约需要的时间是（ ）。

破解“数字+字母”的六位密码大约需要的时间是（ ）。

破解“数字+字母+符号”的六位密码大约需要的时间是（ ）。

## 学习活动 3

### 密码设置的安全性

讨论交流：日常生活中，我们设置密码时，为防止被别人猜到或被计算机程序破解，设置密码时要遵循一些基本的原则。

小组同学交流设置密码应遵循的原则，分享想法。

以下是设置密码的一些原则。

1. 设置密码需要一定的长度和复杂度。例如，六位以上数字、字母或字符的组合。

2. 密码最好包含大小写字母、数字和符号。

3. 密码不要与姓名、生日、电话号码等相同。

总之，密码是重要的个人隐私信息，要注意保护好，不要告诉他人。

在前面分析的基础上，增加符号种类数。每位密码增加 32 种可能性，即  $10+26\times 2+32=94$ 。因此需要的次数是六个 94 相乘。

$94\times 94\times 94\times 94\times 94\times 94=689\ 869\ 781\ 056$  次

将近 7 000 亿次。在这个基础上，密码长度每增加一位，破解所需的最大循环次数就会增加 94 倍。

通过运行程序，填写破解密码大致所需的时间，学生能充分感受到不同密码组合所需破解时间的差异。

这里的破解密码程序主要是利用计算机的运算能力，通过枚举法遍历每一种可能的情况，从而达成破解的目标。

可以看出，即使计算机具有强大的运算能力，在大量数据面前，也需要一定时间才能完成任务。此外，密码位数越多，组成方式越复杂，所需的破解时间就越长。

教学时，要带领学生讨论现实生活中设置密码的安全性要求，包括位数长度和符号类型等。从前面的数据分析，学生已经知道为什么位数越长的密码越安全、符号类型越多的密码越安全。

除此之外，过于常见的密码往往会被轻易破解。

现实生活中，有许多防止密码被破解的方法。

例如，人们在消费支付时，系统通常会设定密码的输错次数，超过指定的次数后就不能继续操作。又如，登录各类应用软件时，除了要求输入密码，通常还要求输入验证码，而且验证码具有时效性，需要在较短时间内输入，否则也不能继续操作。

## 拓展与提升

1. 有一个包括三位数字的密码锁，现在只记得中间一位数字是3。运行配套资源中的参考程序，分析程序破解这个密码的方法。

2. 如果一个三位数，它每个数位上数字的立方之和等于它本身，那么这个数被称为水仙花数，如  $153 = 1^3 + 5^3 + 3^3$ 。

尝试运行配套资源中的参考程序，找到所有水仙花数。

在学生对密码设置要求有了认识之后，可以进一步延伸，引导学生探讨防止密码被破解的方法。

密码在日常生活中很常用。通过了解密码，能够更好地感受身边处处有算法的现实意义，也能树立网络安全意识。

总的说来，随着信息科技的发展，我们一方面要利用技术来解决各种问题，提高解决问题的效率。另一方面，也要尽量避免技术应用带来的安全问题。

1. 知道中间位的数字为3后，密码破解还剩第一位和第三位，那么就可以用两个循环变量*i*和*j*，通过双重循环结构遍历所有可能的数字从而破解密码。也就是：第一位每枚举一个数字，第三位要枚举0至9个数字。

2. 算法与程序实现提示如下。

(1) 定义变量*i*、*j*和*k*，代表各数位上的数字。

(2) *i*从1开始递增到9，*j*和*k*从0开始递增到9。

(3) 计算*i*、*j*和*k*的立方之和。

(4) 把*i*、*j*和*k*组合成一个三位数。

(5) 如果立方之和与三位数相等，则找到一个水仙花数。

(6) 继续寻找其他水仙花数，直到所有三位数都检验完成。

### 【引言】

排序算法是计算机解决问题的基础算法之一，可以将无序的数据按照特定顺序排列。在排序过程中，需要对数据进行比较，并根据比较结果来决定是否交换位置，从而实现排序。最终的排列顺序可以是升序（从小到大）或降序（从大到小）。

通过学习排序算法，可以认识排序的基本思路和方法，了解不同排序算法之间的差异。学习排序算法有助于培养抽象逻辑思维和问题解决能力。

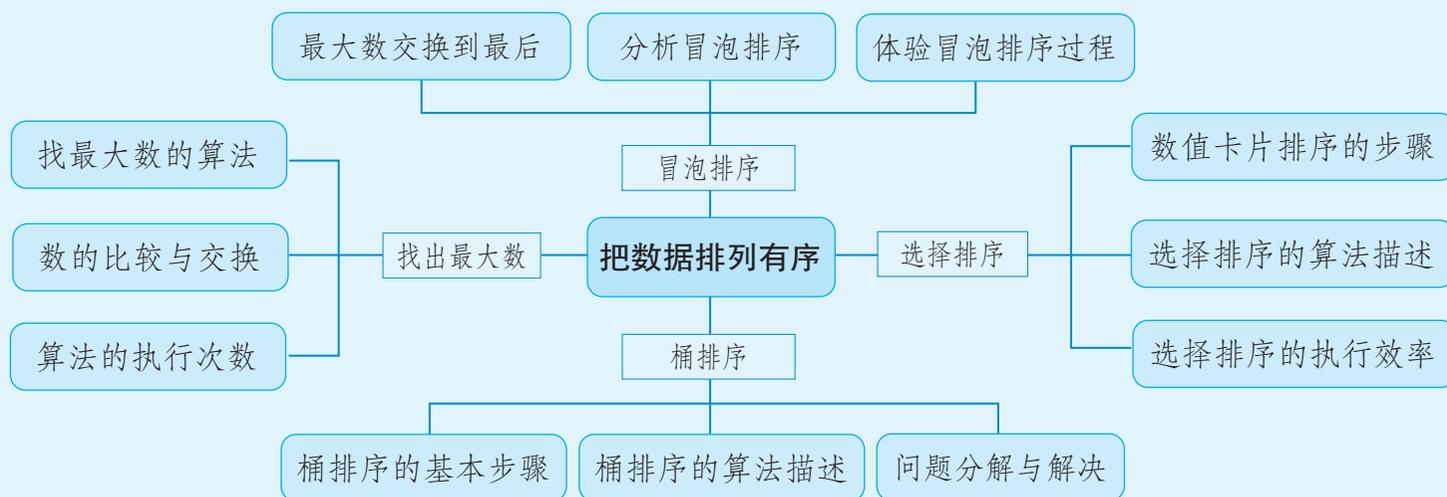
通过本单元的学习，让学生了解排序算法中搜索、比较与交换的基本过程，体会用计算机解决问题中蕴含的算法思想。通过自然语言、流程图等方式描述排序算法，在操作实践中利用程序验证与实现排序算法，了解算法与程序之间的关系。

排序算法有很多种，每一种都有自己的优势和适用范围。学习过程中可以通过对比不同的排序算法，了解算法的效率，让学生意识到面对不同数据规模的问题时，要选择合适的排序算法。这有助于培养学生优化算法的意识，提高用计算机解决问题的效率。

教学时需要将抽象的算法内容与现实的生活场景或实物联系起来，以帮助学生在具体的情境中领悟有关的知识。可以引导学生通过游戏活动、情境创设、角色扮演等方式，在运用排序算法解决具体问题的过程中，逐步感悟其中的思想和方法。结合实物、图片、视频等直观呈现的方式，帮助学生观察排序算法执行的过程、计算执行的次数等。同时，使用适当的学习资源和方法，如编程实现算法、用程序验证算法，让学生在操作实践中体会算法的应用效果和价值。

本单元的学习先从具体问题出发，逐步引导学生认识排序算法的基本思想，了解排序的一般规律，帮助学生在“认识—理解—辨析—应用”的过程中逐步认识排序算法，体会算法的应用。通过这样的教学设计，学生不仅能够做中学，更能做中思，并逐步达成创中思、创中学，培养综合实践能力和创造思维能力。

## 【内容结构】



## 【教学实施条件】

1. 本单元建议在具备运行Python环境的信息科技实验室里教学，能够运行配套资源中的游戏、程序等。
2. 配备教学演示与互动的设备，准备用于教学的图片、流程图、程序、学习任务单、实物等。

## 【学习目标】

1. 在分析排序算法过程中，理解排序算法的基本含义，包括比较与交换的方法、排序规则对数据位置的影响。
2. 能够用自然语言或流程图描述选择排序、冒泡排序和桶排序的算法。
3. 通过分析排序的具体执行过程，认识不同排序算法的特点。
4. 在解决排序问题时，通过分析问题，知道选择适合的排序算法来解决问题。

## 【评价建议】

下表可作为单元学习评价的参考。

评价内容	学生自评	教师评价
用自然语言或流程图描述寻找最大值的算法	☆☆☆☆☆	☆☆☆☆☆
用自然语言或流程图描述选择排序的算法	☆☆☆☆☆	☆☆☆☆☆
用自然语言或流程图描述冒泡排序的算法	☆☆☆☆☆	☆☆☆☆☆
用自然语言描述桶排序的主要步骤	☆☆☆☆☆	☆☆☆☆☆
结合算法描述，知道用适当的排序方法进行排序，解决将一组无序数排序的问题	☆☆☆☆☆	☆☆☆☆☆
了解不同算法的特点，结合具体的应用场景，说出不同排序算法的主要区别	☆☆☆☆☆	☆☆☆☆☆
面对不同观点，大胆说出自己的想法	☆☆☆☆☆	☆☆☆☆☆
在小组合作的任务中与同学积极交流，认真完成任务	☆☆☆☆☆	☆☆☆☆☆

## 第 16 课 比较交换找最值

### 学习目标

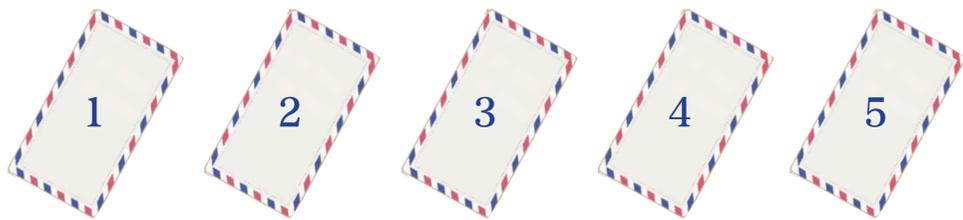
1. 了解在一组无序数中找出最大数的基本步骤，并能辨认其中的循环操作。
2. 分析、描述找出最大数的算法，体验用程序验证算法的一般过程。

日常生活中，我们常常遇到找出最大数的问题，如找出身高最高的人、找出直径最大的水果、找出某月气温最高的一天等。本课通过逐个比较数值大小的方法来找出最大数，认识其中包含的算法。

### 学习活动 1

#### 描述找出最大数的算法

桌上有 5 个信封，编号分别为 1、2、3、4、5，每个信封里装有一张卡片，卡片上写着一个大于 0 的数，请找出其中的最大数。



编号为 1、2、3、4、5 的信封

活动规则：每次只能打开一个信封，查看卡片上的数后把卡片放回原信封内，才能继续打开另外一个信封，查看卡片上的数。

活动要求：以小组合作探究的方式，找出解决问题的算法，并描述算法的基本步骤。

教学重难点：

1. 用流程图描述找出最大数的算法；
2. 用程序验证找出最大数的算法。

本课以学习活动 1 为主介绍，学习活动 2 作为进一步学习拓展时的选择。

通过情境引入，让学生感受到找出最大数的算法在日常生活中的重要性和应用价值。这样的教学设计将使更加主动地学习和探究算法问题，激发他们对算法学习的兴趣。

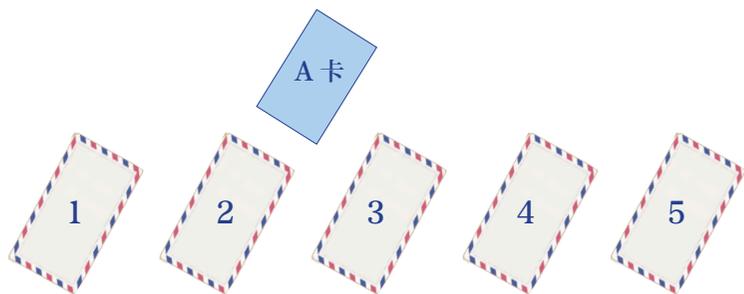
找出最大数的算法涉及搜索、比较、交换、循环等基本操作，可以结合这些操作组织教学活动。

在一组数中找出最大数的问题，在日常生活中非常普遍。例如，在一堆书中找出最厚的一本，在一排树木中找出最高的一棵等。

本课以从 5 个信封中找出最大数作为问题情境，探究其中存在的算法问题。问题本身并不难，但其中包含的步骤、方法却是学生之前很少关注的。此处通过用实物进行操作，让学生体验比较与交换的过程，认识重复操作的次数。

活动步骤描述如下。

第1步：准备一张新卡片，称为A卡，用来记录查找过程中的最大数。



A卡与5个信封

第2步：打开第一个信封，查看卡片上的数并记录在A卡上。

第3步：打开第二个信封，查看卡片上的数并与A卡上的数比较，如果这个数比A卡上的数大，那么把A卡上的数更新成这个数，否则不更新。

第4步：用同样的方法，依次打开第三个、第四个、第五个信封，查看卡片上的数，并与A卡上的数比较，决定是否更新A卡上的数。

第5步：最后，A卡上的数就是最大数。

观察发现，在以上步骤中，从第3步开始就是重复操作。如果首先在A卡上记录0，第2步就可以是重复操作。可以构建循环结构来描述算法。

第1步：在A卡上记录0。

第2步：按顺序打开一个信封，查看卡片上的数，把这个数与A卡上的数比较。如果比A卡上记录的数大，那么把A卡上的数更新成这个数，否则不更新。

第3步：对每个信封做第2步的操作。

第4步：最后，A卡上的数就是最大数。

首先让学生描述要求解的问题和规则，明确要学习的主要内容。

问题1：要解决的问题是什么？

问题2：关键规则是什么？

针对问题1：学生能够用简洁的语言描述问题，即在5个数中找出最大数。明确需要将5个数进行比较，记录最大的数。专门用一张卡片记录当前最大数。

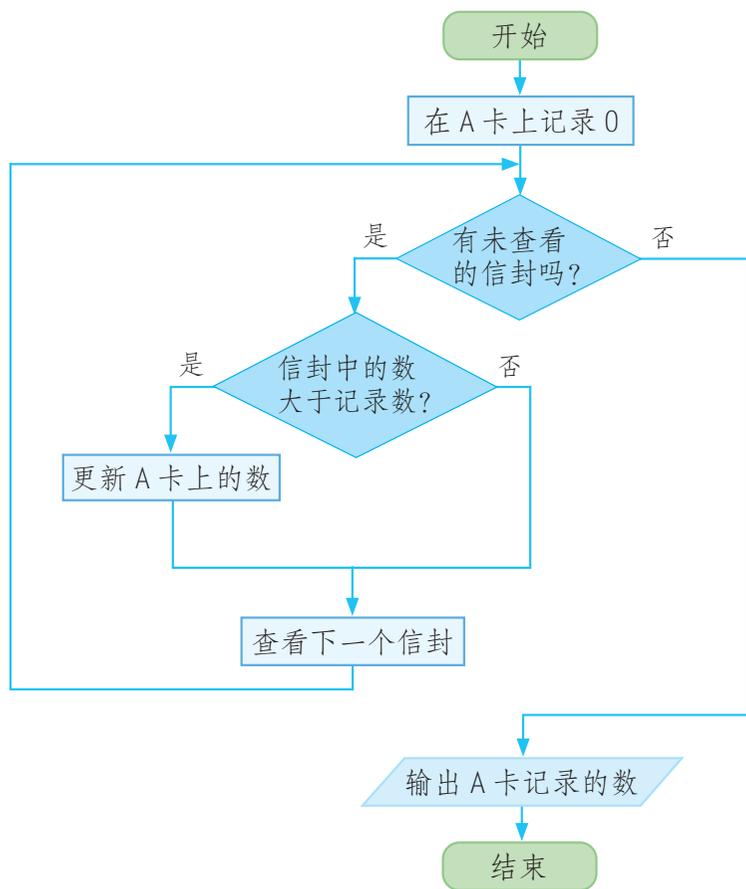
针对问题2：每次只能打开一个信封，打开查看后需要放回原位置。

可以组织学生以小组的形式边演示边描述步骤。

提示学生从1至5号信封或从5至1号信封逐一比较，有规律地查看信封中的数，以避免遗漏。

教学时引导学生关注活动步骤描述与算法步骤描述的不同之处。了解用算法解决问题时，往往借助循环结构解决重复操作的问题，并尽可能地把操作步骤放入循环结构中，简化执行过程。

用流程图表示上述过程，看起来更加简洁直观。



算法的流程图

进一步探究：找出最大数并记录所在的信封。

解决方法提示：增加一张新的卡片，称为 B 卡，用来记录信封的编号。

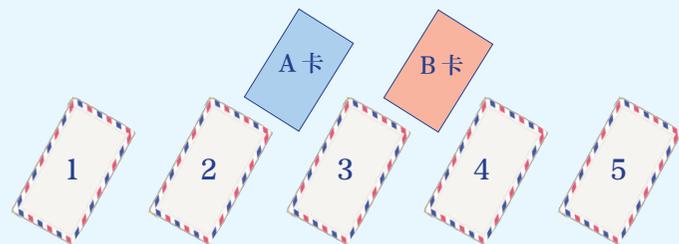
思考：在寻找最大数的过程中，一共要循环多少次？也就是说，要做多少次“比较两个数大小”的操作？如果是 5 000 个数呢？

引导学生观察流程图中存在的循环结构。可以看出，从“比较第一个信封中的数”开始就进入循环结构。

比较和交换作为循环体中的重要内容，也是寻找最大数的关键操作。学生要明确只有信封中的数大于 A 卡上记录的数才需更新 A 卡上的数。

继续观察流程图，提出问题：开始时 A 卡上的数为什么要写 0，写 100 可行吗？

A 卡上要写 0，不能写 100。因为事先知道信封里的数都大于 0，所以 A 卡上的数要比后面查看到的数都要小。如果写 100，后面查看的数不一定大于 100，这样就不能保证找到信封中的最大数。



A 卡、B 卡与 5 个信封

针对具体问题进行分析，让学生理解算法在执行过程中可以不断进行优化，对原有算法进行改进，再添加一张卡片，并在循环体中增加相应的处理操作。

以上例子中一共有 5 个数，如果先在 A 卡上写第一个数，需要循环 4 次。如果先在 A 卡上写 0，则需要循环 5 次，也就是要多一次“比较两个数大小”的操作。因此，如果是 5 000 个数，用以上两种方式，分别需要循环 4999 次和 5 000 次。

## 学习活动2

### 验证找出最大数的算法

下面通过查找列表中的最大数，利用循环结构实现查找最大数的算法。

#### 1. 认识Python中的列表

在Python中，列表是一种常见的数据序列，用于存储一系列数据。

列表使用方括号[]来把数据放入其中，每个数据之间用英文逗号“,”分隔。例如，下面的语句可以创建一个包含5个数的列表。

#### # 创建一个包含5个数的列表

```
box = [24, 91, 42, 108, 35]
```

列表数据的表示和引用从0开始，而不是从1开始。例如，在以上的box列表中，box[0]表示列表的第1个数24，box[1]表示第二个数91。

利用len()函数可以获取列表中的数据个数。例如，利用下面的赋值语句，可以用变量cc保存box列表中的数据个数，即5。

```
cc = len(box)
```

#### 2. 回顾while循环

在Python中，用while循环可以实现算法的循环结构。

当循环的次数不确定时，可以根据条件进行判断。当条件成立时，执行条件后的语句；当条件不成立时，结束循环。

while循环的基本格式如下。

```
while 判断条件:  
    语句组
```

列表在Python程序实现中很常用，很多程序编写都基于列表。

教学时通过演示讲解，带领学生学习Python中列表的相关知识，让学生初步了解列表的用法，认识列表存储数据的特点。

这里用简单的5个数，引导学生认识列表及其基本用法。此处不必要求学生完全掌握，后续还将进一步学习。

列表是可变的，可以添加、删除或修改其中的数据。

使用append()可以把数据添加到列表的末尾。

使用insert()可以在列表的指定位置插入数据。

使用del语句可以在列表中删除指定的数据。

前面的学习中，学生已经知道循环结构和while循环，此处带领他们回顾已知，引入新知。

### 3. 程序实现与验证

打开下面找出列表中最大数的程序并运行，分析程序与算法的关系。

```
box = [24, 91, 42, 108, 35]    # 创建一个包含 5 个数的列表
cc = len(box)                 # 用 cc 保存 box 列表中的数据个数
a = box[0]                    # 用 a 保存最大数，初始值设为第一个数
i = 1                          # 循环变量初始值设为 1
while i < cc:                  # 用循环结构逐个比较列表中的数
    if box[i] > a:              # 如果当前数大于 a 的值
        a = box[i]             # 更新 a 的值
    i = i + 1                  # 循环控制变量增加 1，继续循环
print('最大数为: ', a)        # 输出找到的最大数
```

在 while 循环中，使用条件语句 if 来比较，当前数 box[i] 的值是否大于已经找到的最大数 a。如果是，就将 a 的值更新为当前数 box[i] 的值。这样，通过循环操作逐个查找，找出列表中的最大数并更新到 a 中。

控制变量 i，从 1 变化到等于 cc 的值（数据个数）时，结束循环。循环结束后，输出变量 a 的值，就是这个列表中的最大数。

### 拓展与提升

随意列举一些无序的数，从中找出最小数，应该如何操作？尝试用程序验证。

可以先引导学生运行程序，观察运行结果，再与学生一起分析程序，初步了解语句的功能，不必要求理解语句的用法。

理解循环体的部分，教师可以先通过演示，让学生认识列表中每个数与当前最大数进行比较的过程。然后引领学生分析，总结比较的规律：从第 0 个数到第 4 个数。

列表中的数与当前数比较的过程

第一次比较	第二次比较	第三次比较	第四次比较
if box[1] > a: a = box[1]	if box[2] > a: a = box[2]	if box[3] > a: a = box[3]	if box[4] > a: a = box[4]

分析程序后，可以进一步引导学生尝试修改程序代码，并再次运行程序。

可以先在记录卡上写上第一个数，然后依次与后面的数进行比较，当被比较的数小于记录数时，更新记录数，直到所有数比较完成。程序验证环节建议引导有兴趣的学生学习，可以通过改写以上找出最大数的程序来完成。

## 第 17 课 选择排序轻松做

### 学习目标

1. 能够梳理出选择排序的基本步骤，通过补全流程图，了解用流程图描述的选择排序算法。
2. 了解排序数据量与算法执行次数的关系，能够用程序验证选择排序算法。

排序算法有很多种，这一课学习比较简单的选择排序。下面通过对5个数进行排序的探究活动来学习这一内容。

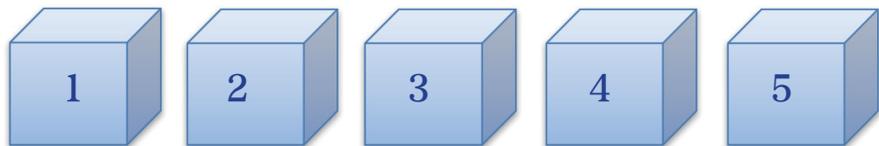
### 学习活动 1

#### 描述选择排序的算法

桌上有5个盒子，分别以1、2、3、4、5编号，每个盒子里装有一张卡片，卡片上写着一个大于0的数。

请依据这些卡片上的数，把它们从大到小排序。

要求：把写着最大数的卡片放在1号盒子，写着第二大数的卡片放在2号盒子，以此类推。



编号为1、2、3、4、5的盒子

教学重难点：

1. 选择排序的基本过程；
2. 用流程图描述选择排序算法。

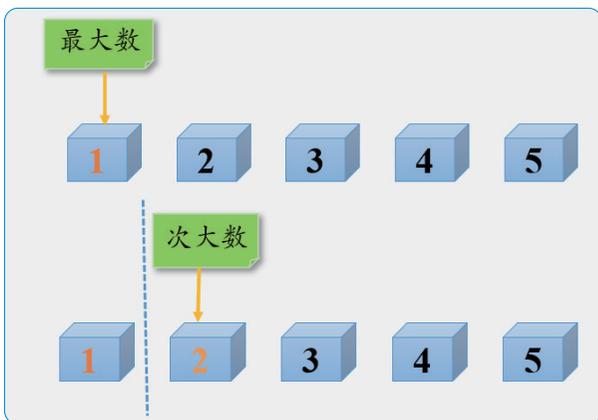
这一课主要围绕选择排序算法及其执行过程、执行次数来展开学习。以学习活动1和学习活动2为主介绍，学习活动3作为分层教学时的选择。

首先借助实物（5个盒子、5张卡片），通过查看、比较、记录盒子里卡片上的数，体验排序的步骤和基本过程。

由于只有5个数，操作起来比较容易。

这里可以是教师演示讲解，也可以组织学生进行小组学习。后者更能调动学生的积极性，在引导学生共同探讨选择排序执行步骤的过程中，促进学生交流和协作。

为了让抽象的思维过程易于学生理解，教师还可以结合视频进行演示讲解，引导学生观察选择排序的动态过程后，再分析总结排序过程的基本步骤。



最大数与次大数的位置

选择排序的基本步骤如下。

第1步：用前一课所学方法，找出最大数的卡片所在盒子。

第2步：把写着最大数的卡片与1号盒子里的卡片进行交换。

第3步：以此类推，在2号到5号盒子里，用“找出最大数”的方法，继续查找其中最大数的卡片，找到后与对应盒子里的卡片进行交换。

以上操作，其实也形成了一个具有循环结构的算法。

第1步：把卡片放入盒子，已排序个数设为0。

第2步：在未排序数中找到最大数，与未排序的第一个数交换位置。

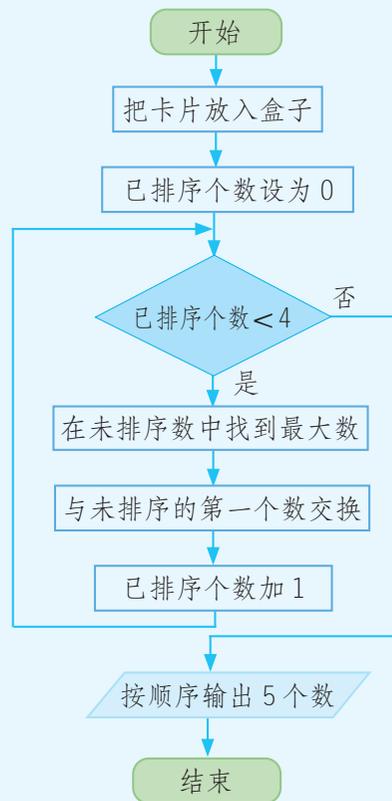
第3步：重复第2步，依次查看所有数并交换到相应的位置。

第4步：按顺序呈现5个数。

选择排序算法借助在特定范围内寻找最大值或最小值，并交换位置的方式进行排序。通过探究选择排序与寻找最大值或最小值的关系，感知排序算法与循环结构的联系。

排序算法解决的关键问题：搜索、交换位置、改变搜索范围。

教学时，不必要求学生画出完整流程图，可以引导他们对照算法的步骤，通过补全的方式来认识流程图。



算法的流程图

## 学习活动2

### 分析选择排序的过程

5张卡片上有5个数，如108、24、12、91、35。运用前面所学选择排序算法，完成从小到大的排序，验证算法执行的步骤，记录卡片交换的次数，填写任务单。

初始状态:	108	24	12	91	35
第1轮:	108				
第2轮:	108	91			
第3轮:	108	91			
第4轮:	108	91			
最终结果:	108	91			

□ 原始数据    ☆ 未排序中最大值    ■ 已排序的数据

任务单示例

进一步思考：学习选择排序算法后，有一位同学说：“我观察一下就能把这5个数排好顺序，不必反复操作，计算机有点笨呀！”

你认同这个说法吗？与小组同学交流你的想法。

学习活动2通过直观的数值排序活动，让学生进一步巩固选择排序执行次数与数据顺序的关系，了解选择排序算法的执行效率。

让学生将算法的执行过程填写到任务单中，进行展示和交流，可以检验他们对排序算法执行过程的了解程度。

学生可能会采用直接观察的方法将卡片进行交换，这样做不利于感知算法的执行过程，教学时要注意提示和引导学生用选择排序算法完成任务。

教学时也可以采用其他学习活动，如小组同学一天步数的排序。全班学生按小组分为4至6人一排，每人手持一张写有不同步数的纸，用选择排序算法完成小组同学步数的降序或升序排列。

提示：只有5个数时，人通过看一看、比一比，就能完成排序。但如果是50个数、500个数、5000个数……，就不能很快完成了。然而，通过算法编写程序让计算机执行，不管是5个数，还是50个、500个、5000个……计算机都能够瞬间完成。

### 学习活动3

#### 验证选择排序的算法

在上一课的学习中，当找到列表中的最大数后，还可以进行多次比较和交换，实现列表中的数按从大到小排序。

打开以下参考程序并运行，利用程序实现选择排序算法，把数据由大到小排序。

```
data = [64, 25, 12, 22, 11, 90, 88, 77, 55, 33] # 要排序的数据列表
n = len(data) # 获取列表数据的个数
i = 0 # 设置外层循环控制变量的初始值
while i < n - 1: # 外层循环用于控制排序的轮数
    max = i # 最大值变量设为当前轮数的起始值
    j = i + 1 # 设置内层循环控制变量的值
    while j < n: # 内层循环用于找到未排序数中的最大数
        if data[j] > data[max]: # 如果找到的值更大就更新最大数
            max = j
        j = j + 1
    if max != i: # 如果最大数不在指定位置就交换位置
        data[i], data[max] = data[max], data[i]
    i = i + 1 # 进入下一个数的排序
print(' 排序后的数 :', data) # 输出排序后的列表
```

学生们经过前面的学习，对排序算法有了初步的感知。教师可以根据学生学习情况，引导他们依据算法流程图，分析程序语句的功能。

通过利用程序对数据排序的验证，让学生进一步了解选择排序算法。可以根据实际情况，引导学生课堂学习或课后拓展探究。

这里不要求学生读懂所有程序语句，只需引导他们从功能实现上认识程序与算法的对应关系，不必要求理解语句的用法。

但对于学有余力的学生，可以适当提示，让学生对程序进行调整，尝试输入不同的数据量进一步探究排序规则。

## 拓展与提升

在 Python 中，可以使用 `sorted()` 函数来进行排序，编程验证与实现算法时更加简单便捷。打开配套资源中的参考程序，观察并运行，体会用程序实现算法的多样性，感受多途径解决问题的过程。

程序实现离不开算法的描述，算法为程序的编写提供了解决问题的思路。人们为了更方便地调用程序，通常会把相关功能封装为函数，便于编程时直接使用。

在 Python 中，一般把 `sorted()` 称为排序函数，它把排序的算法封装在函数之中，可以直接使用，学生学习起来会更加简单。

参考程序如下。

```
# 创建一个包含 10 个数的列表
box = [24, 99, 56, 25, 91, 77, 42, 83, 108, 35]
print(' 排序前:', box)
# 使用 sorted() 函数进行排序
box=sorted(box)
print(' 排序后:', box)
```

## 第 18 课

### 冒泡排序齐体验 (1)

#### 学习目标

1. 通过“排队”游戏，体验冒泡排序算法的思想，培养用算法解决问题的能力。
2. 初步认识冒泡排序算法，能够用自然语言描述冒泡排序算法的执行步骤。

在上一课中，我们学习了选择排序的基本方法，知道比较与交换在选择排序时有很重要的作用。这一课继续学习排序算法中一种较简单直观的排序方法——冒泡排序。

以把一组数从小到大排序为例，冒泡排序就是在一组无序的数中，从前面开始，比较相邻两个数的大小，再根据比较结果交换两个数的位置，如此循环，直到把整组数从小到大排列完成。

#### 学习活动 1

##### 体验冒泡排序的过程

情境游戏：请5至10位同学站成一排，每人随机抽取一张卡片，每张卡片上写有不同的数，然后按照每人手持卡片上的数，由小到大排队。

同时指定一名同学填写活动任务单。



写有不同数的卡片

教学重难点：

1. 冒泡排序的基本过程；
2. 描述冒泡排序算法的执行步骤。

本课通过情境体验、图示等方法来引导学生学习，进一步熟悉排序算法中的比较与交换规律。

如果课时有限，学习活动1也可以把卡片摆放在桌面上，让学生轮流完成不同轮次的寻找最大数，从而完成排序。

在冒泡排序过程中找最大值或最小值时，学生往往很容易忽略规则，教学时注意关注和引导，要求他们在体验过程中观察与思考，同时填写任务单。

学生活动过程中，教师注意巡视引导，保证排序过程中比较与交换的准确性。活动完成后，可以结合任务单进行展示交流，总结相关过程与方法。

## 1. 游戏规则

(1) 依次对相邻两位同学卡片上的数进行比较。如果左侧同学卡片上的数比右侧同学卡片上的数大，两位同学就交换位置，否则保持原位置不变。依次比较完成后，最右侧同学手持卡片上的数就是最大数。

(2) 除了最后一位同学，对其他所有同学，依据卡片上的数重复以上操作。

(3) 持续对越来越少需要排序的卡片进行比较与位置交换，直到没有任何两位同学需要交换位置。

## 2. 准备材料

(1) 数值卡片，每张卡片上写有一个不同的数。

(2) 任务单，用于记录每一轮比较的过程。



任务单示例

活动步骤提示如下。

第1步：参与游戏的同学站成一排，每人抽取一张数值卡片。

第2步：通过第一轮比较，把手持最大数卡片的同学排列到最右侧。

从左侧第一位同学开始，把自己卡片上的数与右侧同学卡片上的数进行比较，如果自己卡片上的数大就相互交换位置；如果不大，保持原位置不变。类似地，当前第二位同学与第三位比较、第三位同学与第四位比较……最后两位同学比较。

第3步：用第一轮的方法，除去已经排序好的同学，其余同学进行第二轮比较、第三轮比较……直至所有同学按照手持卡片上的数，按从小到大排列。

把每轮排序的结果填写到任务单中。

实际体验时，通过第一轮把最大数“冒泡”到最右侧的活动，学生通常就能知道冒泡排序中的比较与交换方式，感受找到最大值的过程以及其中的交换规则。

学生有了第一轮的经验，后续四轮就变得很容易了。

完成排队任务后思考下面的问题。

1. 冒泡排序每一轮分别固定了什么数，有什么特点？
2. 从第二轮开始，排序时可以简化哪一步？要比较几次？

下面用自然语言描述冒泡排序算法。

第1步：比较相邻的两个数，如果第一个比第二个大，就交换位置。对每一对相邻数进行同样的操作，从开始两个数到最后两个数。操作后，排在最后面的数就是最大数。

第2步：除已排序的数，重复第1步的操作，对其余数进行比较与交换，直到没有任何一对数需要交换位置。

问题1：引导学生观察每一轮“冒泡”后，都会固定当前未排序数中最大数的位置。

问题2：引导学生发现第一轮已经将最大数固定在最右侧，比较的次数可以减少一次，即已经排好位置的最大数，不必参与后续的比较。这既是对每轮找到最大数意义的回应，又引出了后几轮需要处理的关键问题。

通过体验探究，可以促使学生思考冒泡排序的内在逻辑，巩固他们对算法的认识，培养他们对排序问题的敏感性和解决问题的能力。这种引导方式有助于学生更深层次地理解冒泡排序的算法思想和优化方法。

冒泡排序算法的描述是本课教学的关键点，也是难点。“排队”体验活动能够逐步化解这个难点。活动完成后，要适时引导学生总结活动步骤，感悟算法规律。

实际教学时，教师可以组织更多同学参与排序活动。

此外，也可以组织其他形式的小组排序体验活动，如按学生的身高排序、按学号排序等。每个小组安排一名同学查看，保证按照冒泡排序的规则展开活动。

## 学习活动2

### 算法与用计算机解决问题

在用计算机解决问题时，算法起到了重要作用。

1. 算法为计算机解决问题提供了明确的求解步骤

依据算法，我们可以将复杂的问题分解为一系列简单、可执行的步骤，从而确保计算机能够按照预定的要求和顺序进行处理。

2. 算法是实现自动化和智能化的基础

通过算法控制，计算机可以自动地执行任务、处理大量的数据、依据判断条件进行决策等，提高了解决问题的智能化水平。

3. 算法可以提高问题求解的质量和效率

针对同一问题，不同的算法可能产生不同的解决方案。通过比较和分析不同算法的性能和结果，可以选择最优的算法来解决问题。此外，通过选择适当的算法，能够优化计算机存储与处理的步骤、时间等，减少计算量、降低存储需求，从而提高算法的执行效率。

总之，算法是程序设计的主要依据，也是解决实际问题的策略。用计算机解决问题时要充分利用算法来体现问题求解过程的自动化和智能化。

## 拓展与提升

跳绳数据比一比。

按照小组同学的学号或座位顺序，把写有小组同学跳绳数据的卡片排成一排，然后依据冒泡排序算法，按照跳绳数据从大到小的顺序排序。两人一组，一人摆放，一人记录每轮数据卡片的位置。

可以结合学生前面学习中遇到的问题来展开。例如，有同学认为只对少量数排序时，反复比较、交换，是一件很麻烦的事情，而且这种方法显得很笨。初学算法与程序实现时，很多同学都会有类似的想法。

可以通过对比计算机解决问题的方法与人解决问题的方法，来解开学生的疑惑。让他们知道，对问题进行抽象与问题求解过程的自动化是计算机解决问题的基本方法，其中需要把待解决的问题抽象为能够利用循环结构自动执行的操作。

此外，可以依据教学实际和学生的认知基础，灵活掌握这一内容学习的范围和难度。

通过跳绳数据比一比的活动，进一步巩固对排序算法的认识。在活动过程中，明确每一轮“冒泡”后被归位的数。通过填写、展示任务单，梳理和总结有关过程，深化对每轮关键操作的认识。例如，执行次数与未排序数的顺序有关，不同的初始顺序可能导致不同的比较和交换次数等。

## 第 19 课 冒泡排序齐体验 (2)

### 学习目标

1. 进一步认识冒泡排序算法，能够对一组数据进行排序过程分析。
2. 能够用程序验证排序算法，观察数据的排序过程，培养用算法解决问题的意识。

上一课我们通过“排队”活动，体验了冒泡排序的算法步骤，下面通过对数据进行排序，分析冒泡排序的过程，并利用程序进行验证与实现。

### 学习活动 1

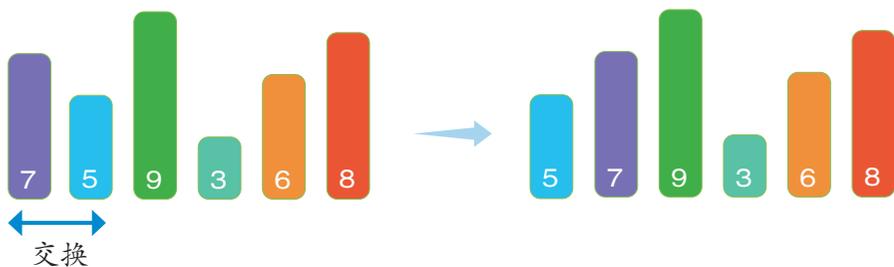
#### 冒泡排序的过程分析

任务 1：把最大数交换到最后。



6 个不同的数

利用冒泡排序算法，把最大数排列到最后（最右侧），操作描述如下。



第 1 步：比较第一个数和第二个数， $7 > 5$ ，交换位置

教学重难点：

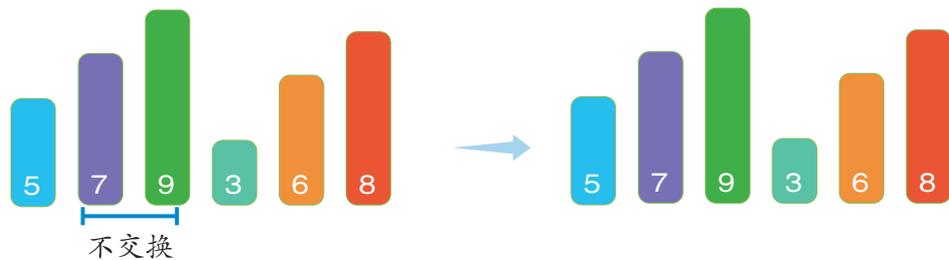
1. 分析冒泡排序的基本过程；
2. 用流程图描述冒泡排序算法。

在初步了解冒泡排序的过程之后，为了加深认识，巩固所学，可以通过图片、视频等方式，进一步帮助学生理解冒泡排序算法的步骤，通过分析对数据进行冒泡排序的过程，了解冒泡排序的思想和方法。

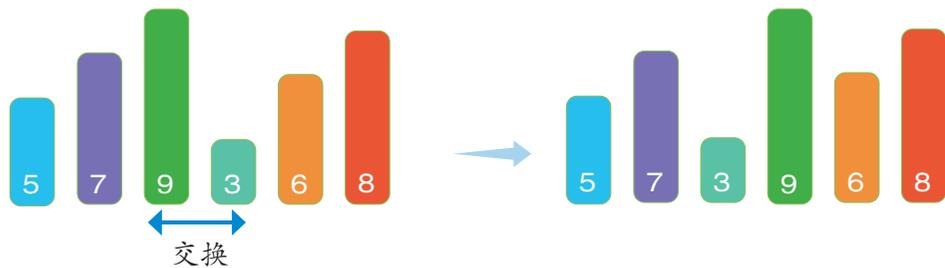
冒泡排序是稳定的原地排序算法，即相同的数据在排序后保持其原有的顺序，编程实现也相对简单。但是，对于较大数量的数据，冒泡排序会很耗费时间，对于已经部分排序或接近完全排序的数据，效率也不高。

在实际应用中，冒泡排序通常不是最高效的排序方法，但在算法教学和理论探讨上有其独特的价值，学习这一内容有助于学生对排序算法基础概念和规律的认识。

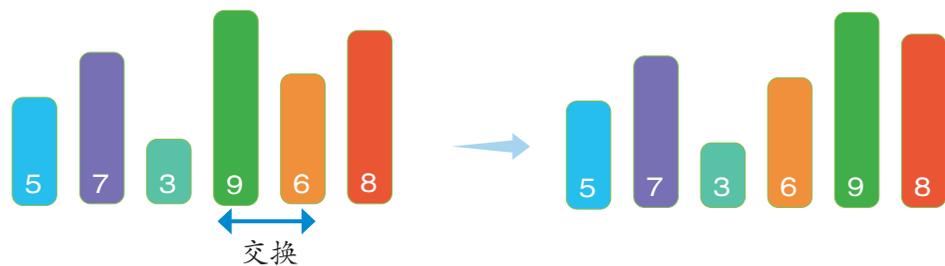
在引导学生分析冒泡排序过程时，先通过第一轮数值比较寻找最大数，认识交换的基本规则，然后通过多轮类似操作完成数值的排序。这样，通过两个单重循环来展开学习内容，可以大大降低学生认知的难度。



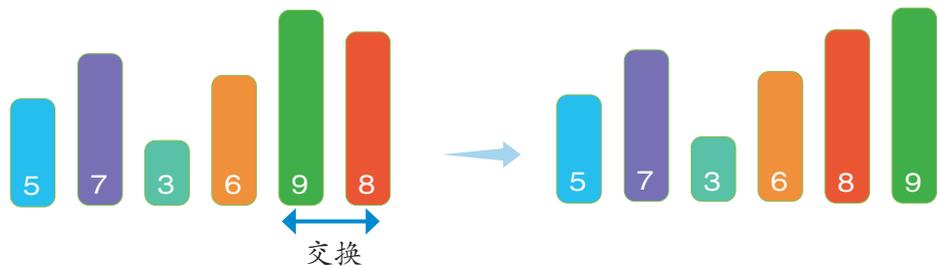
第2步：比较第二个数和第三个数， $7 < 9$ ，不交换位置



第3步：比较第三个数和第四个数， $9 > 3$ ，交换位置

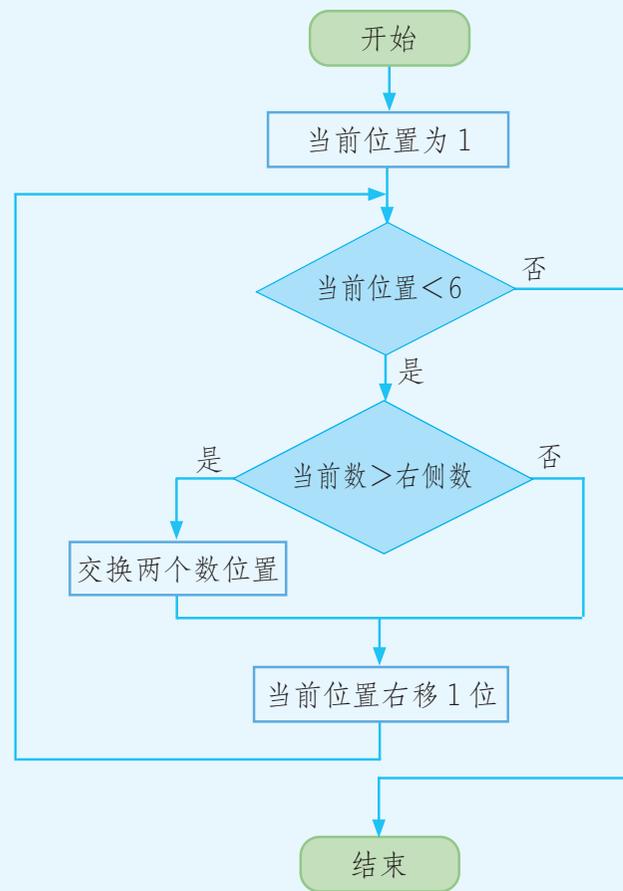


第4步：比较第四个数和第五个数， $9 > 6$ ，交换位置



第5步：比较第五个数和第六个数， $9 > 8$ ，交换位置

结合流程图来进行教学，更便于学生理解。以下是本案例冒泡排序中寻找最大数的参考流程图。

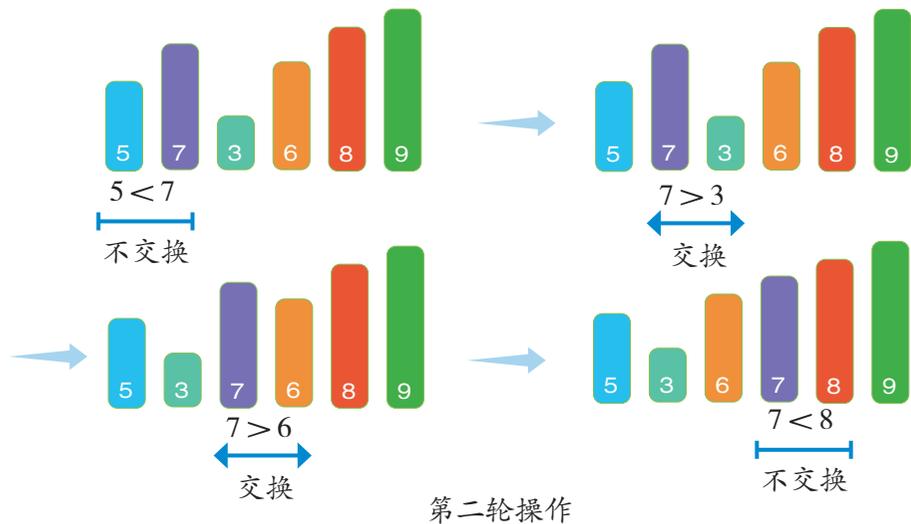


算法的流程图

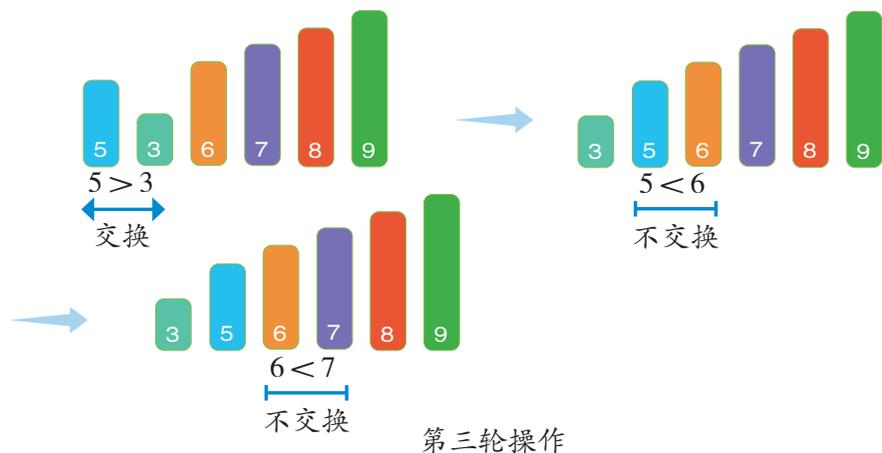
任务2：按数值从小到大排序。

第一轮：把最大数交换到最后，前面已经完成。

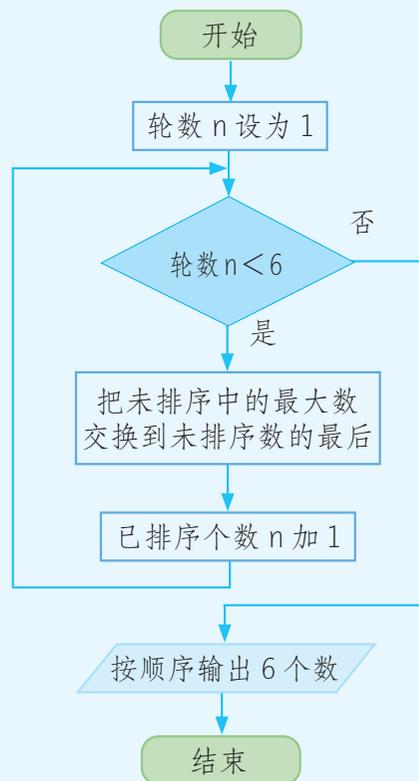
第二轮：返回最左边，在第一个至第五个数中，通过比较、交换找到它们中的最大数，把它排到第五的位置。



第三轮：返回最左边，在第一个至第四个数中，通过比较、交换找到它们中的最大数，把它排到第四的位置。



以下是冒泡排序中多轮比较、交换，把相应最大数排列到指定位置的参考流程图。



算法的流程图描述

在这个冒泡排序的例子中，一共有6个数，第1轮需要比较 $6-1=5$ 次，第2轮需要比较4次，第3轮需要比较3次，第4轮需要比较2次，第5轮需要比较1次。

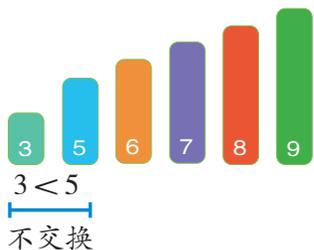
因此，总的比较次数为 $5+4+3+2+1=15$ 次。

第四轮：返回最左边，在第一个至第三个数中，通过比较、交换找到它们中的最大数，把它排到第三的位置。



第四轮操作

第五轮：返回最左边，比较第一个和第二个数，把较大的数排到第二的位置。



第五轮操作

## 学习活动2

### 验证冒泡排序的算法

下面通过观察、运行冒泡排序的程序，体会算法的实现。

第1步：打开配套资源中的“冒泡排序.py”程序。

第2步：观察、运行这个程序，体验冒泡排序的算法执行过程。

第3步：大致了解程序与算法的对应关系。

第4步：尝试修改程序中的排序数据，再运行程序，观察变化。

事实上，交换数的次数和数的排列顺序有关。假设出现某种极端情况，例如，这些数正好以从小到大的顺序排列，那么便不需要任何交换操作；反过来，数据要是以从大到小的顺序排列，那么每次比较数值后就都需要进行交换。

通过步骤描述，分析了冒泡排序每轮比较操作过程和固定的最大数，引导学生感受算法对用计算机解决问题的重要性。

必要时可以补充，循环比较结束的条件可以描述为：当某一轮比较时，如果未发生任何数的交换，就可以结束循环。

这是一个双重循环结构的程序。通过运行程序，引导学生进一步观察、感受冒泡排序算法，探索编程让计算机执行算法的过程。这里不要求学生读懂所有程序语句，只需引导他们从功能实现的角度，认识程序与算法的关系。比如，哪些程序语句实现了把最大数交换到最后？哪些语句控制排序的轮次？

为了便于学生对照理解算法与程序的关系，能够找出循环的判断条件，这里的程序仍然采用 while 循环来编写，而把编程常常采用的 for 循环作为另外的参考程序，教学时也可以引用。

## 拓展与提升

1. “报数”排序游戏。在10张卡片上分别写下一个50以内的自然数，随机分发给10位同学，然后选一位同学出来从1开始报数。当报的数与某张卡片上的数相同时，该同学就手持卡片站出来。继续报数，直到50为止，手持卡片的同学按先后顺序站成一排，这时发现，这10个数已经按顺序排列。

用计算机解决排序问题时，也可以用以上“报数”的方法。计算机按顺序数数，当数到的数与参与排序的某个数相同时，这个数就会被加入已排序数的队列中。

2. 在配套资源中，有用for循环实现的冒泡排序参考程序，打开并运行这个程序，观察它与while循环的不同之处，体会同一算法用不同程序实现的方法。

提示：在Python中，for循环语句的基本格式如下。

**for 变量 in 序列:**  
    **循环体**

1. 这是一个通过“报数”进行排序的活动，让被“叫到”的数按顺序“站”到有序队列中。该活动可以进一步巩固与拓展学生对排序算法的认识，同时知道：用计算机解决同一问题时，往往会有不同的算法和不同的程序实现方法。

2. 例如，利用下面的语句，可以输出0到9这10个数。

```
for a in range(10):
```

```
    print(a)
```

range()可以用来生成一个数据序列。range()的基本格式如下。

```
range(开始值, 结束值, 步长值)
```

其中，开始值默认从0开始，为0时可以省略；结束值不包括自身的值；步长默认为1，为1时也可以省略。所以，range(10)相当于range(0, 10, 1)，其中0和1均为默认值，可以省略。

大多数情况下，while循环和for循环两者可以互相改写，但每一种方式大都有与其相适应的应用场合。

## 第 20 课 化大为小桶排序

### 学习目标

1. 认识桶排序算法，能够使用自然语言描述桶排序算法的执行步骤。
2. 了解桶的数量和范围在桶排序中的作用，感受将大的数据量化大为小的分治思想。

这一课学习排序算法中较常用的又一种方法——桶排序算法。利用这个方法可以把较大的数据量分为较小的数据量，便于分析处理。

### 学习活动 1

#### 分析桶排序的算法

老师收到了 50 位同学参加朗诵活动的报名信息表，这些同学来自一至五年级各班。现在这些信息表处于混乱状态，老师希望把这些信息表按照报名同学的年龄从小到大排序。

提示：信息表中所填年龄大小是实际岁数。例如，10.08 表示 10 岁 8 月，9.11 表示 9 岁 11 个月。

要求：现在请你帮助老师想一个方法来完成任务。

桶排序中的“桶”代表的是一个数据区间范围，里面可以放置一个或多个数据，就像现实生活中的桶一样，能够作为容器使用。



生活中作为容器的桶

教学重难点：

1. 桶排序的基本过程；
2. 将大的数据量化大为小的分治思想。

本课学习桶排序算法，通过问题情境引入学习内容。

引导学生把一组较多的数据分为多个数值范围的子区间，然后分别对这些子区间进行排序，再按顺序组合起来。

教学时，可以提供有助于学生观察、动手实践的多样化素材，帮助学生理解桶排序的算法思想，引导他们探究桶的数量和范围对算法的影响，提高学生的学习兴趣。

借助问题情境，引领学生思考。使用排序算法处理较多数据时，如何用一种更快的算法来解决这个问题，从而引入将大的数据量分解为小的数据量进行处理的分治思想。

教学时，可以展示桶排序算法的动态演示过程，引导学生观察分析，初步了解桶排序算法。

先把信息表中的年龄数据整理为表格。

整理后的年龄数据表

年龄 数据	10.11	10.07	11.08	7.01	7.05	7.04	8.07	10.07	7.07	9.08
	11.05	10.02	11.07	7.08	10.04	8.07	11.02	9.04	9.10	8.04
	11.02	10.09	10.11	11.09	9.02	9.06	8.05	7.06	8.10	10.05
	9.07	9.05	8.11	7.04	11.06	10.03	10.09	9.11	9.09	11.11
	10.10	7.10	8.08	11.04	7.02	11.11	8.05	7.09	7.09	8.09

思考下面的问题。

1. 一至五年级的学生年龄通常有几个区间?
2. 划分的年龄区间如何排序?

通过观察分析,基本操作步骤描述如下。

第1步:设置5个桶(数据区间),用于放置不同年龄段的信息表。

第2步:把所有年龄数据逐个放入对应的桶里,填写表格。

按年龄分段的数据表

桶1: 7(含)至8岁	7.01	7.05									
桶2: 8(含)至9岁	8.07	8.07									
桶3: 9(含)至10岁	9.08	9.04									
桶4: 10(含)至11岁	10.11	10.07									
桶5: 11(含)至12岁	11.08	11.05									

第3步:把每个桶里放入的信息表按年龄大小排序。

每个桶里的信息表数量较少,用前面学习过的排序方法很快就能完成。

由于学生来自一至五年级,将信息表中低年级学生的年龄与高年级学生的年龄进行比较发现,他们不在同一个区间范围内。

在这个过程中,让学生探讨桶排序中桶的数量,即划分几个年龄区间的问题。划分区间后,可以将同一个区间的数据进行排序,形成一个有序的序列,最终再按顺序进行组合。这是对桶排序基本思想的探究。

教学时,可以给出部分已经填写好数据的表格,然后组织学生分组找出其他年龄段的数据并排序。

学生可能会问:年龄相同的信息表应该怎么排序呢?

这时,可以引导学生按年级、学号等条件来排序。

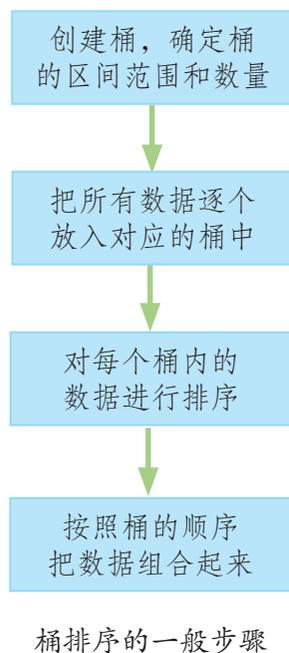
排序后按年龄分段的数据表

桶1	7.01	7.02	7.04	7.04	7.05	7.06	7.07	7.08	7.09	7.09	7.10
桶2	8.04	8.05	8.05	8.07	8.07	8.08	8.09	8.10	8.11		
桶3	9.02	9.04	9.05	9.06	9.07	9.08	9.09	9.10	9.11		
桶4	10.02	10.03	10.04	10.05	10.07	10.07	10.09	10.09	10.10	10.11	10.11
桶5	11.02	11.02	11.04	11.05	11.06	11.07	11.08	11.09	11.11	11.11	

第4步：依次取出5个桶里已经排好序的信息表，按桶号顺序组合到一起，全部信息表就排序完成了。

通过上述操作，可以总结桶排序算法的一般步骤。

1. 创建桶，确定桶的区间范围和数量。
2. 把所有数据逐个放入对应的桶中。
3. 对每个桶内的数据进行排序。
4. 按照桶的顺序把数据组合起来。



桶排序体现了化大为小、分而治之的问题分解思想。当要处理的数据较多而且数值分布较为均匀时，这种方法具有明显的优势。

引领学生结合桶排序算法思想进行思考实践，利用填表、任务单等记录算法的执行过程，这样的教学设计有助于学生理解。在这个过程中将桶排序算法的步骤进行直观呈现，使学生感受桶排序算法的分治思想，认识桶排序算法在解决大量数据问题时的优势。

桶排序也称箱排序，基本算法思想是假设待排序的一组数均匀分布在一个范围之中，并且可以将这一范围划分成几个子范围，也就是能够把这组数分到有限数量的桶里。然后每个桶里的数再分别进行排序，最后依次把各个桶中的数连接起来，形成一个有序数列。

桶排序的基本思想就是把大的问题分解为小的问题，在逐步解决小问题的过程中解决大问题。

对于桶中数据的排序，选择合适的排序方法也很重要。

## 学习活动 2

### 桶排序的优势与不足

讨论交流：桶排序有哪些优势？又存在哪些不足？

提示：

1. 可以从数据量来分析，数据量分别是20个、500个、1 000个、100 000个等时的排序情况。
2. 可以通过网络搜索这一问题并阅读，小组同学一起分析讨论，确定主要结论并在全班交流分享。

### 拓展与提升

1. 在学校组织的参观博物馆活动中，需要为来自不同年级五个班的同学安排车辆和座位。具体情况：如果每个班安排一辆车，车辆座位数不够；如果每个班安排两辆车，每辆车都会有空位置。

请你思考：如何规划同学乘车的问题？这个过程中是否存在算法？

2. 通过搜索引擎或生成式人工智能应用软件查找：还有哪些常用的排序算法？它们各有什么特点？排序算法可以解决哪些生活与学习问题？

主要优势：

1. 面对大量数据，没办法将所有数据一次处理完成时，可以分成一定数量的桶来分别处理；
2. 在数据分布均匀时，具有较高的排序效率，因为桶排序将数据分散到多个桶中独立进行排序，不需要逐个比较和交换数据；
3. 可以灵活调整桶的数量，优化桶排序的性能；
4. 通过在每个桶中使用稳定性较好的排序算法，可以保证桶排序的稳定性。

主要不足：

1. 桶排序需要预先知道待排序数据的范围，否则无法合理设置桶的数量；
2. 数据分布不均匀时影响排序效率，某些桶可能会比其他桶集中了更多的数据，导致排序效率下降；
3. 对于大量重复数据，因某些桶数据过多而增加排序时间。

1. 参考方法：利用桶排序的算法思想，可以把五个班所有同学按学号进行分组排序，然后按照车辆的顺序号及座位数安排座位。
2. 排序算法还有很多，如插入排序、快速排序等。如果学生对相关内容感兴趣，可以引导他们通过图片、视频演示等方式进一步学习。例如，围绕排序算法的特点、执行效率进行分析，深化不同算法具有不同执行效率、不同算法具有不同应用场合的认识，培养学生分析算法、应用算法的能力。

## 【引言】

通过枚举遍历所有数据是一种简单直接的数据查找方法。

本单元的遍历主要指将问题所有可能的数据都通过枚举法查找出来。在枚举过程中，检查每个结果是否是当前问题的正确答案。如果是，则问题解决完成；如果不是，则继续枚举，直到所有可能的数据都被访问和检查完成。

在进行归纳推理时，如果逐个考查了某类事件的所有可能情况后得出一般性结论，那么该结论通常是可靠和稳定的。遍历是一种访问数据、处理数据的方法，在算法实现中，可以用于搜索、排序、过滤等操作。

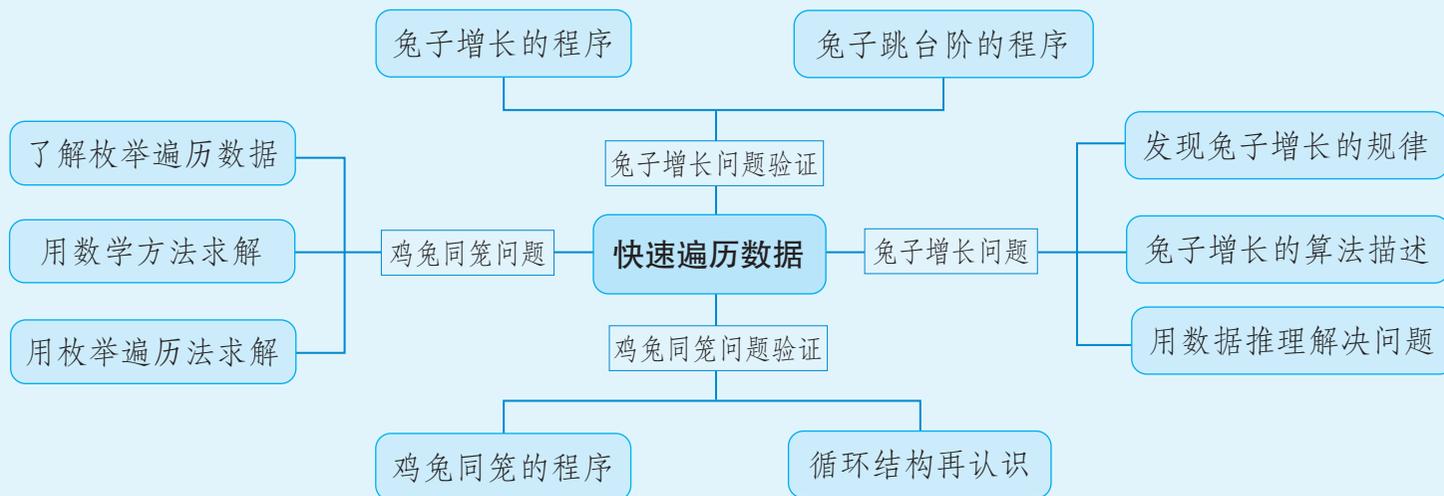
实际上，在前面的学习中，已经多次用到通过枚举法遍历数据的方法。例如，找出1到 $n$ 之间所有能被5整除的自然数，通过逐个检查1到 $n$ 之间的所有自然数，判断其是否能被5整除，如果能被整除就输出显示。又如，破解密码锁的问题，也是通过逐个检查相应范围内的所有字符组合，从而找到满足条件的答案。

在现实生活中，人们也常用这种方法思考问题、解决问题。

例如，鸡兔同笼问题，起初会用“凑数”的方法来寻找答案，这个“凑数”过程其实就是通过枚举法遍历所有可能数据的过程。但人思考、解决这个问题时，通常仅限于数据量较小的情况，如果数据量较大，这个“凑数”过程就会很耗费时间。如果编程让计算机通过遍历所有数据来求解，因为计算机运算速度非常快，数据处理能力很强，只需非常短的时间就能完成任务。

因此，用计算机解决问题时，通过枚举法遍历所有数据是一种常用而且有效的问题求解方法。本单元通过几个典型例子，让学生认识通过遍历数据解决问题的基本过程和方法，提升分析问题与解决问题的能力，培养适应数字时代的思维能力、创新能力。

## 【内容结构】



## 【教学实施条件】

1. 本单元建议在具备运行Python环境的信息科技实验室里教学，能够运行配套资源中的游戏、程序等。
2. 配备教学演示与互动的设备，准备用于教学的图片、流程图、程序、学习任务单、实物等。

## 【学习目标】

1. 通过表格描述数据变化的过程，发现数据变化规律。
2. 通过对几个经典问题的求解，认识通过枚举法遍历所有可能数据求解问题的一般过程。
3. 针对不同的问题，知道运用正向思维和逆向思维分析问题、解决问题。
4. 进一步认识算法的循环结构，能够找到并描述循环的判断条件。
5. 进一步了解算法与程序实现的关系。

## 【评价建议】

下表可作为单元学习评价的参考。

评价内容	学生自评	教师评价
能通过表格描述鸡兔同笼问题的数据变化过程	☆☆☆☆☆	☆☆☆☆☆
能发现鸡兔同笼问题的数据变化规律，找到满足条件的答案	☆☆☆☆☆	☆☆☆☆☆
能看懂鸡兔同笼问题的算法流程图，并用程序进行验证	☆☆☆☆☆	☆☆☆☆☆
能通过表格描述兔子增长问题的数据变化过程	☆☆☆☆☆	☆☆☆☆☆
能发现兔子增长问题的数据变化规律，找到满足条件的答案	☆☆☆☆☆	☆☆☆☆☆
能用程序验证兔子增长问题的算法，了解算法与程序的对应关系	☆☆☆☆☆	☆☆☆☆☆
能用程序验证兔子跳台阶问题的算法	☆☆☆☆☆	☆☆☆☆☆
用数据推理求解简单的问题	☆☆☆☆☆	☆☆☆☆☆
在遇到问题时，能和同学互相讨论交流	☆☆☆☆☆	☆☆☆☆☆

## 第 21 课

### 鸡兔同笼巧计算

#### 学习目标

1. 了解鸡兔同笼问题的求解方法，能通过表格列出数量变化，发现其中的规律。
2. 感受遍历法的应用，能看懂鸡兔同笼问题的算法流程图，了解算法与程序的对应关系。

我国古代典籍《孙子算经》中记载了许多有趣的问题，其中就有“鸡兔同笼”问题。

书中是这样描述的：今有鸡兔同笼，上有三十五头，下有九十四足，问鸡兔各几何？

这段话的意思是：有若干只鸡和兔关在同一个笼子里。从上面数，有 35 个头。从下面数，有 94 只脚。这个笼子里的鸡和兔各有多少只？

#### 学习活动 1

##### 用数学算式求解鸡兔同笼问题

为了便于理解，先把原问题的数量减少为：今有鸡兔同笼，上有 6 头，下有 18 足，问鸡兔各几何？

这里的 6 个头，表明是 6 只鸡或兔。

方法 1：假设 6 只全部是兔。

如果全部是兔，那么 6 只兔一共有 24 只脚，实际上只有 18 只脚，于是需要减少 6 只脚，即  $24 - 18 = 6$ 。这样，自然就是 3 只兔和 3 只鸡。

教学重难点：

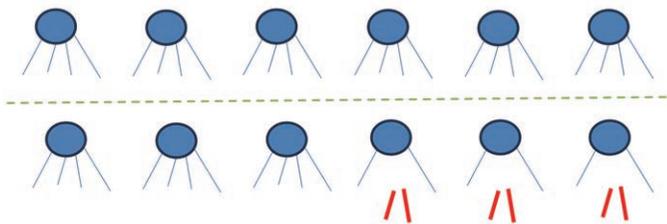
1. 通过表格列出鸡和兔的数量变化，发现其中的规律；
2. 鸡兔同笼问题的算法流程图描述。

本课的目的是通过枚举法遍历数据求解鸡兔同笼问题。鸡兔同笼问题对于学生来说不陌生。如果教学中直接讲解用枚举法求解的话，学生们很容易产生疑问。例如，数学课中为什么不使用这种方法呢？

为此，本课先引导学生用数学方法求解，在此基础上再学习用枚举法求解。这样，学生就很容易通过原有知识基础，在比较过程中理解枚举法的应用。

《孙子算经》中记录了古人的解法“抬腿法”，这一方法很巧妙，但是分析的过程比较复杂，小学生不易理解。

对于小学生来说，较容易的方法是图示或列表法。

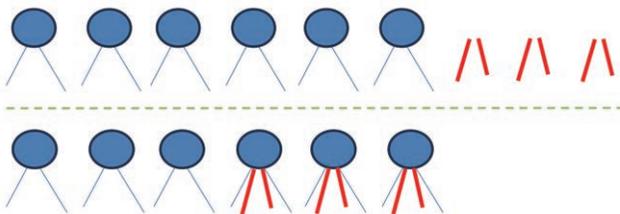


6只全部是兔的计算示例

鸡： $(6 \times 4 - 18) \div 2 = 3$ (只)    兔： $6 - 3 = 3$ (只)

方法2：假设6只全部是鸡。

如果全部是鸡，那么一共有 $6 \times 2 = 12$ 只脚，实际上有18只脚，于是少了6只脚，即 $18 - 12 = 6$ 。需要把6只脚添加上，自然就是3只兔和3只鸡。



6只全部是鸡的计算示例

兔： $(18 - 6 \times 2) \div 2 = 3$ (只)    鸡： $6 - 3 = 3$ (只)

还原问题的求解：

今有鸡兔同笼，上有三十五头，下有九十四足，问鸡兔各几何？

方法1：假设35只全部是兔。

鸡： $(35 \times 4 - 94) \div 2 = 23$ (只)    兔： $35 - 23 = 12$ (只)

方法总结：鸡的数量 = (总头数  $\times$  4 - 总脚数)  $\div$  2

方法2：假设35只全部是鸡。

兔： $(94 - 35 \times 2) \div 2 = 12$ (只)    鸡： $35 - 12 = 23$ (只)

方法总结：兔的数量 = (总脚数 - 总头数  $\times$  2)  $\div$  2

教学时，把问题规模缩小之后，便于图示讲解，有助于学生理解。在用图示描述时，可以全程展示数据变化的过程，让学生对全过程有一个完整的认识。

必要时，可以进一步引导学生，把这个计算方法“告诉”计算机，按照总结出来的算式进行计算。

程序提示如下。

```
a = input('一共有多少个头:')
b = input('一共有多少只脚:')
c = (int(a)*4-int(b))/2
print('鸡有:',int(c),'只')
print('兔有:',int(a)-int(c),'只')
```

借助问题引出后面的学习内容：如果不知道这个计算方法，就无法把表达式写出来，请思考还有其它方法求解吗？

借助枚举法，通过遍历相应的数据也可以求解。

除了用上面的算式求解，还可以借助枚举法来遍历相应的数据求解。

## 学习活动2

### 用枚举法求解鸡兔同笼问题

#### 1. 列表显示数据变化过程

先假设35只都是鸡，计算出脚的数量；如果数量不符合，则减一只鸡，增加一只兔，再计算脚的数量；如此循环遍历，直到找到正确的鸡和兔数量：23只鸡和12只兔。

用表格呈现鸡和兔的数量变化（单位：只）

鸡	35	34	33	32	31	……	25	24	23
兔	0	1	2	3	4	……	10	11	12
脚	70	72	74	76	78	……	90	92	94

#### 2. 算法描述

根据以上分析，用自然语言描述求解的算法。

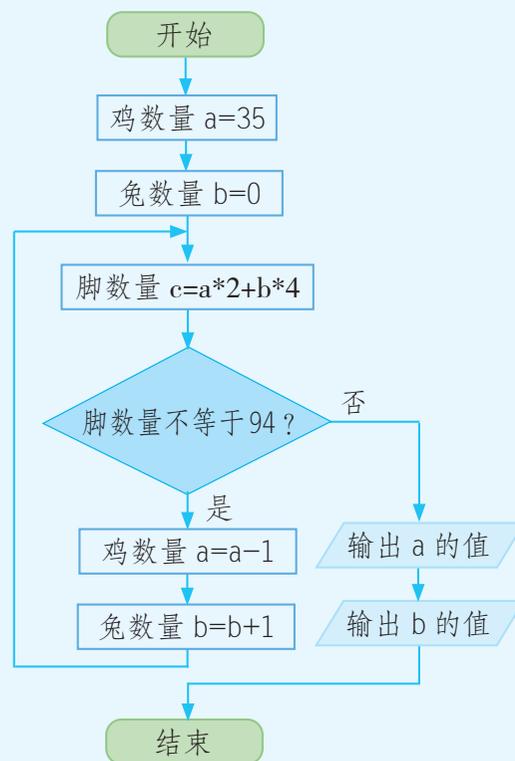
第1步：初始化鸡的数量“ $a = 35$ ”和兔的数量“ $b = 0$ ”。

第2步：计算脚的数量“ $c = a \times 2 + b \times 4$ ”。

第3步：把脚的数量与94进行比较。如果不相等，将鸡的数量减1，将兔的数量加1，并回到第2步继续循环；如果相等，则输出当前鸡的数量和兔的数量，结束循环。

引导学生分析。假设有35只鸡，兔就是0只，那么一共有70只脚，但实际有94只脚，显然不符合题意；继续尝试，鸡减少1只，兔增加1只，即34只鸡和1只兔，那么一共有72只脚，依然不符合题意；继续改变鸡和兔的数量，直到找到答案。

每一步都很简单而且相似，让人来做显得很笨拙单调。然而计算机不怕繁琐并且计算速度快。这样的方法正好适合编程用循环结构来实现，用程序逐个尝试鸡和兔的数量组合。



算法的流程图

进一步思考：如果先假设35只都是兔，用枚举法遍历相应数求解时，应该对算法进行哪些调整？

### 学习活动3

#### 编程验证鸡兔同笼问题

可以用循环结构实现，通过不断增加兔的数量和减少鸡的数量，逐步接近正确答案。

下面是一个参考程序。

```
a = 35                # 设置鸡数量的初始值 a
b = 0                # 设置兔数量的初始值 b
while True:         # 用循环结构进行控制
    c = a*2+b*4     # 计算脚的数量
    if c == 94:    # 比较 c 的值是否等于 94
        print(' 鸡的数量:', a) # 输出获得的结果
        print(' 兔的数量:', b)
        break      # 结束循环
    else:
        a = a-1    # 鸡的数量减少 1 只
        b = b+1    # 兔的数量增加 1 只
```

第1步：初始化，鸡的数量“ $a = 0$ ”，兔的数量“ $b = 35$ ”。

第2步：计算脚的数量“ $c = a \times 2 + b \times 4$ ”。

教学时，可以结合流程图，运用前面学习的循环结构和分支结构进行编程示范。也可以让学生依据教师所给程序，自主探究学习。

在 Python 中，通常可以用 `while True` 语句创建一个无限循环，表示循环体内的指令会不断地重复执行，直到循环体内出现某种能够终止循环的指令，如 `break` 语句。

该程序利用 `if……else……` 语句作为判断脚数量是否等于 94 的条件，并在获得鸡和兔的正确数量后，通过 `break` 语句退出循环。

用枚举法遍历数据，是计算机在处理大量数据时非常有优势的方法。这个例子通过编程实现计算机替代人工完成查找数据的任务。

用计算机解决问题时，应充分考虑计算机的运算特点来设计适合的算法。设计的算法恰当与否，直接影响程序实现的可行性、复杂程度和执行效率。

再次提醒：程序编写是算法转化为代码的过程，需要依据语法使用规则，准确输入名称、符号、对齐方式等。

## 拓展与提升

1. 韩信是我国历史上很有领兵能力的一名将领。在民间流传着这样的一个故事。

有一次，汉军统帅韩信带 1 500 名勇士与楚军交战，战死四五百人。为了再战，韩信快速地清点人数，他要求 3 人一排站队，结果多出 2 人；5 人一排站队，多出 4 人；7 人一排站队，又多出 6 人。韩信马上宣布，我军有 1 049 名勇士。汉军本来就信服韩信，这一来更相信他有神机妙算。于是士气大振，一鼓作气，击败楚军。

那么，在知道了这三次布阵的结果后，韩信是如何算出到底有多少名士兵呢？如何设计算法让计算机来找出这个数？

2. 在一千多年前的《孙子算经》中，也记录着这样一道算术题：今有物不知其数，三三数之剩二，五五数之剩三，七七数之剩二，问物几何？

用现代汉语描述这道题就是：现有一些不知道数量的物品，3 个一组数剩余 2 个，5 个一组数剩余 3 个，7 个一组数剩余 2 个，这些物品的数量是多少？

尝试描述求解这个问题的算法，找到 100 至 1 000 之间符合条件的物品数量。

1. 这是我国古代的一个故事，包含了一个实际问题，如何把实际问题转化为能用计算机解决的问题，需要进行问题的分析和描述。要从复杂的关系中，抽取明确的、可执行的步骤。

用数学方法来表述这个已知条件：

有一个数，除以 3，余 2；除以 5，余 4；除以 7，余 6。

从故事中还知道，士兵人数应该在 1 000 人以上，这样查找这个数的范围可以从 1 000 开始。

那么，应该怎么找这个数呢？

这个问题仍然可以用“笨办法”来寻找答案，就是一个数一个数地尝试。用计算机解决这个问题的方法非常简单，就是从 1 000 开始遍历每一个数，直到找到符合条件的数。

配套资源中有相应的参考程序。

2. 这道算术题古代称为“剩余定理”问题。利用枚举法遍历相应的数据，就可以求解这个问题。

其中，三个判断条件如下：

这个数除以 3 的余数是否等于 2；

这个数除以 5 的余数是否等于 3；

这个数除以 7 的余数是否等于 2。

配套资源中有相应的参考程序。

## 第 22 课 兔子增长有规律 (1)

### 学习目标

1. 结合问题情境，依据图示，能够通过表格列出兔的数量变化。
2. 观察列表呈现的数据，能够发现数据变化的规律，认识其中存在的算法。

上一课学习了我国古代的一个经典算法问题“鸡兔同笼”，这一课将通过探究“兔子增长问题”，学习国外的一个经典算法。

先从兔子农场的问题情境开始。

1 月，农场里有一对小兔；

2 月，这对小兔长成一对大兔；

3 月，这对大兔生了一对小兔，这时农场里有 2 对兔；

4 月，大兔又生了一对小兔，上个月的小兔长成了大兔，这时农场里有 3 对兔；

5 月，两对大兔各生了一对小兔，上个月的小兔长成了大兔，这时农场里有 5 对兔；

以此类推……直到 12 月。

到 12 月的时候，假设兔都活着，那么农场里有多少对兔？

教学重难点：

1. 依据图示，通过表格列出数据的变化；
2. 发现数据的递增规律。

兔子农场问题是斐波那契数列的起源问题，该问题的内涵学生理解起来有一定的难度。所以在提出问题后，还需要引导学生通过用表格列出数据的方法，观察数据的变化，发现存在的规律，及时解决学生理解上的困惑。

学生的困惑主要集中在时间上，所以需要明确：第一对兔 1 月时是小兔，2 月时长大为大兔，3 月时才能生出一对小兔。

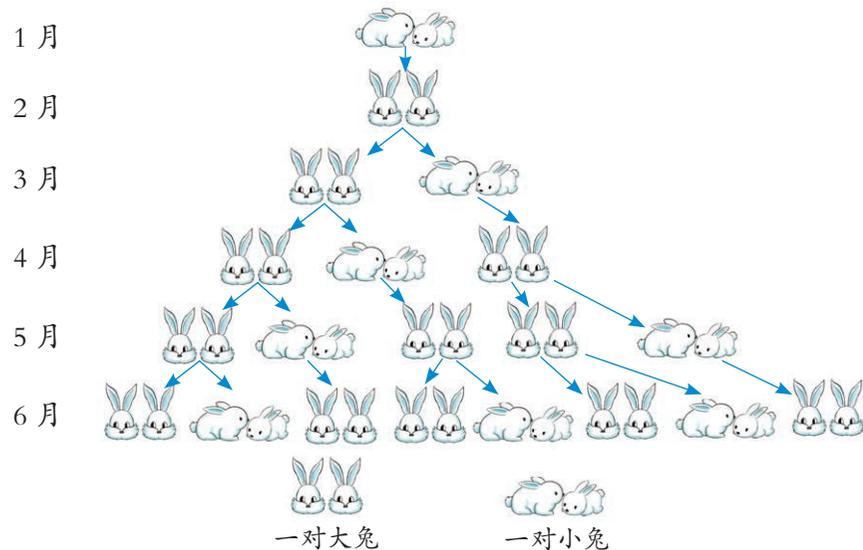
另外，还要强调兔都活着。

有学生可能会说，一对兔一次可以生出多只兔，这时就要及时引导，让学生知道这里是为讲解问题而假设的一种特殊情境，不必比照现实情境。

## 学习活动 1

### 探寻兔的数量变化规律

下面结合图示，通过列表的方法进行推演。



1月至6月兔的数量变化

分析故事情境，可以发现：

1月兔的数量是1对，2月兔的数量是1对，从3月开始，每个月兔的数量是前两个月的数量之和，把每个月兔的数量进行列表统计。

每个月兔的数量变化表（单位：对）

月份	1月	2月	3月	4月	5月	6月	7月	8月	9月	10月	11月	12月
小兔	1	0	1	1	2	3	5	8	13	21	34	55
大兔	0	1	1	2	3	5	8	13	21	34	55	89
总数	1	1	2	3	5	8	13	21	34	55	89	144

教学过程可以安排如下。

1. 用表格列出6个月兔的数量变化。
2. 结合算法流程图，分析数据变化的规律。
3. 分析算法，描述算法。

教学时，可以先用图示呈现前六个月兔的增长情况。学生理解起来如果还存在困难，教师可以鼓励学生互相分享和讨论，这样他们更容易达成共识，加深对问题要求的理解。一旦理解了图示内容和兔的增长规律，再列表呈现就变得相对简单了。

在寻找每个月兔的数量变化规律时，仍然建议组织学生讨论。教师适时启发与帮助，让学生不断经历“假设—验证”的过程，从而发现规律，得出结论。

学生在低年级时学习过“找规律”的知识，但没有建立“数列”的概念，因此，学习过程中更多引导他们发现规律，暂时不必引入数列的概念。

那么，如何描述兔子增长的数据变化规律呢？

## 学习活动2

### 用算法描述兔的数量变化

可以用  $F[1]$ 、 $F[2]$ 、 $F[3]$ …… $F[12]$ ，分别表示从1月到12月兔的数量。

$$F[1] = 1$$

$$F[2] = 1$$

$$F[3] = F[2] + F[1]$$

$$F[4] = F[3] + F[2]$$

……

$$F[12] = F[11] + F[10]$$

找到这些规律，算法就比较容易总结了。

通过创建循环结构，就可以描述算法。

1.  $F[1]$ 和 $F[2]$ 的初始值都设为1，表示1月和2月的兔数量。
2. 用循环变量 $n$ 表示月份，初始值设为3，表示从3月开始计算。
3. 建立以下循环结构：
  - (1) 如果 $n \leq 12$ ，就进入循环；
$$F[n] = F[n-1] + F[n-2]$$
即 $F[3] = F[2] + F[1]$ ， $F[4] = F[3] + F[2]$ ， $F[5] = F[4] + F[3]$ ……
  - (2) 将 $n$ 加1，继续循环。
4. 当 $n > 12$ 时，结束循环。
5. 输出 $F[12]$ 的值。

数据递推是按照一定的规律来计算、推导一组数的变化规律，这种规律通常是将一个复杂的计算过程转化为简单过程的多次重复。用递推算法推演能充分体现计算机快速运算和不知疲倦工作的特点。

兔子农场问题是一个典型的递推问题，从已知数据开始，找出规律，一步一步地求出未知数据，这属于一种用正向思维解决问题的方式。

相比之前的算法描述，此处的算法描述采用了更接近程序实现的描述方式，这不仅可以为后续的程序实现做好准备，还能让学生把算法设计、算法描述与程序实现建立关联。

教学时，要注意引导学生进行思维方式的转化，把自然语言的描述尽可能地贴近能够编程让计算机解决问题的描述方式。

## 拓展与提升

假设兔子农场里有一只兔子在玩跳台阶。它一次可以跳1级或者2级台阶，现在总共有8级台阶，兔子跳完这些台阶，有多少种不同的跳法？



兔子跳台阶情境示意图

先逐个台阶分析这个问题。

1. 要跳到第1级台阶，只有1种跳法。
2. 要跳到第2级台阶，有2种跳法，即跳1级再跳1级和一次跳2级。
3. 要跳到第3级台阶，跳法更多了。

用逐级分析的方法有点复杂。这时，我们换种思路，采用逆向思维来思考这个问题：

兔子可以从哪几级台阶跳到第3级上来呢？

兔子一次可以跳1级或者2级台阶，那么只可能从第1级台阶或第2级台阶上跳到第3级上。如果用  $F[1]$ 、 $F[2]$ 、 $F[3]$  分别表示从地面跳到第1、2、3级台阶的跳法数量，那么跳到第3级台阶的跳法数量就可以计算如下。

$$\text{已知： } F[1] = 1$$

$$F[2] = 2$$

$$\text{求解： } F[3] = F[1] + F[2] = 3$$

依据上述规律，跳到第4级至第8级的跳法数量如下。

$$F[4] = F[2] + F[3] = 2 + 3 = 5$$

$$F[5] = F[3] + F[4] = 3 + 5 = 8$$

$$F[6] = F[4] + F[5] = 5 + 8 = 13$$

$$F[7] = F[5] + F[6] = 8 + 13 = 21$$

$$F[8] = F[6] + F[7] = 13 + 21 = 34$$

## 第 23 课

### 兔子增长有规律 (2)

#### 学习目标

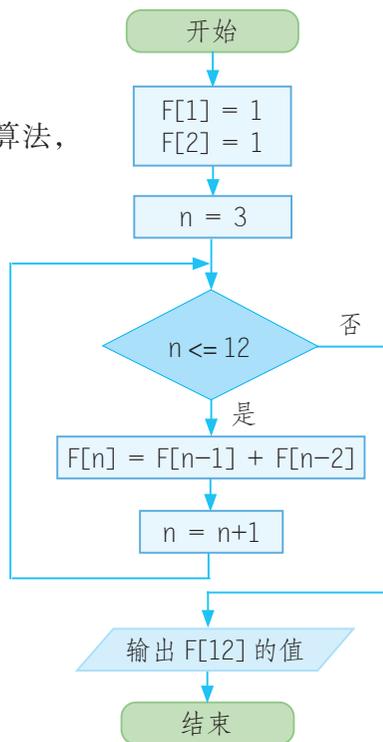
1. 进一步了解兔子增长的数据递增规律，能够用流程图描述相应的算法。
2. 能够利用程序验证兔子增长问题的算法，体会算法与程序的对应关系。

这一课继续探究“兔子增长问题”，通过用流程图描述算法，并用程序进行验证，感受用算法解决问题的基本过程。

#### 学习活动 1

##### 用流程图描述兔子增长算法

依据上一课中用自然语言描述的算法，用流程图描述。



算法的流程图

教学重难点:

1. 依据自然语言描述的算法补充流程图;
2. 利用程序验证兔子增长的算法。

教学时，依据学生的学习能力和认知基础，把自然语言描述的算法转化为流程图时，可以引导学生尝试画完整流程图，也可以用填空的方式进行补充。必要时，也可以直接提供完整流程图。

## 学习活动 2

### 用程序验证兔子增长算法

打开参考程序，观察、运行，分析程序与算法的对应关系。

```
F = [0,1,1]      # 把 1 月、2 月兔的数量放入列表，第 0 项设为 0
n = 3           # 设置要计算的起始月份
while n <= 12:  # 如果月份数 n 小于等于 12，就进入循环
    F = F + [0]  # 为列表增加一项，默认值为 0
    F[n] = F[n-1] + F[n-2] # 计算当前月份兔的数量
    n = n+1     # 月份数增加 1，继续计算
print('第 12 个月兔的数量是：', F[12]) # 输出计算结果
```

以上求解的问题就是斐波那契数列问题，这组数因以兔子繁殖为例而引入，故又被称为兔子数列。

教学时依据学生的学习能力，可以把算法验证与实现的参考程序，用填空的方式引导学生补充，也可以直接提供完整的程序，并带领学生分析程序与算法的对应关系，体会程序语句的功能。

因为 Python 的列表从第 0 项开始，因此这里先把第 0 项设为 0，第 1 个月、第 2 个月兔的数量分别设为 1，并从第 3 个月开始计算，直到第 12 个月。

“F = F + [0]”表示为名为 F 的列表增加一项，默认值为 0，用于存储将要计算月份兔的数量。

如果时间允许，可以适当介绍本课这一组数列的故事，但不必过多地深入。

意大利数学家莱昂纳多·斐波那契 (Leonardo Fibonacci) 在 1202 年所著的《计算之书》中，提出了有趣的兔子增长问题，书中这样描述：一般而言，兔在出生两个月之后，就会有繁殖能力，一对大兔每个月能生出一对小兔。如果所有的兔都活着，那么一年以后可以繁殖多少对兔？

斐波那契数列指的是这样一组数：1、1、2、3、5、8、13、21、34、55、89……这组数从第 3 个数开始，每一个数都等于前两个数之和。

## 拓展与提升

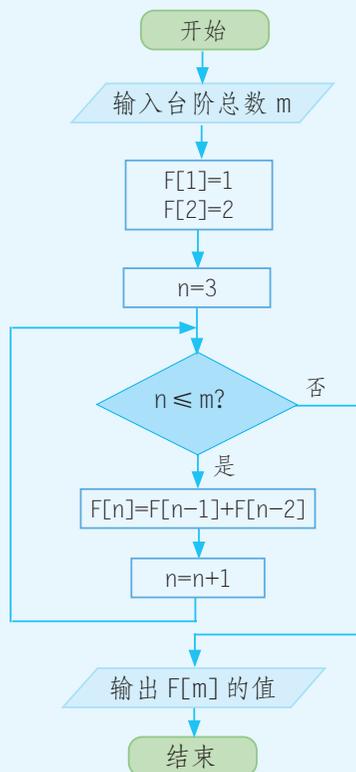
1. 配套资源中有上一课“兔子跳台阶问题”的参考程序。打开这个程序并运行，输入不同的台阶总数，观察输出结果，体会数据的变化规律。

2. 阅读下面的内容，了解自然界中一些符合斐波那契数列的独特现象。斐波那契数列存在于许多自然现象之中。

例如，有些树木的生长，由于新生的枝条往往需要一段休息时间，供自身生长，而后才能萌发新枝。所以一株树苗会在间隔一段时间后长出一条新枝。第二年新枝休息，老枝依旧萌发。此后，老枝与休息过一段时间的新枝同时萌发，当年生的新枝则次年休息。这样，一株树木各个年份的枝干数，就会构成斐波那契数列。

又如，一些植物的种子排列也符合斐波那契螺旋，向日葵的种子排列形成的斐波那契螺旋有时能达到 89 个。这些植物按照自然规律进化成这样，这似乎是植物排列种子的优化方式。它能使所有种子具有差不多的大小却又疏密得当，不至于在中心处挤了太多的种子。

1. 分析兔子跳台阶问题时的逆向推理，在现实生活中有广泛的应用。逆向推理可以帮助我们从不或结果出发，思考问题产生的原因或解决的条件，从而更好地理解 and 解决问题。采用逆向推理，可以更好地理解问题本质，发现问题的突破口，从而找到更好的解决方案。



算法的流程图

2. 可以启发学生，观察生活环境中的花草、树木，发现更多符合斐波那契数列的现象。如梅花、飞燕草、菊花、百合花、蝴蝶兰等植物，它们的花瓣数目也具有斐波那契数列特征。叶子的生长方式也是如此。许多植物的叶子从枝干生长出来时，为了在生长的过程中最佳地利用空间和保障每片叶子都获得阳光照射，叶子之间会形成一个“黄金角度”。

### 【引言】

在前面单元中，学生学习了一些常见的算法，对分析问题、算法描述和用程序实现算法有了一定的了解。本单元旨在引导学生学习更多的算法，但基于学生的认知基础、课时等原因，这里只列举两种常见的算法——规划算法和网页排名算法。

首先，规划思想是人们思考问题、解决问题的一种重要方式，体现了人类的智慧。基于规划思想的规划算法广泛存在于社会生活中。例如，工业制造、农业生产、交通运输、航空航天以及药品研发、艺术设计等领域，都可以利用计算机解决一些人解决起来较困难的规划问题。规划算法通常用于确定一系列动作或决策，以从给定的一组初始状态到达目标状态。一般涉及对系统行为的预测和控制，以优化某种性能指标，如时间、成本、效率或质量等。

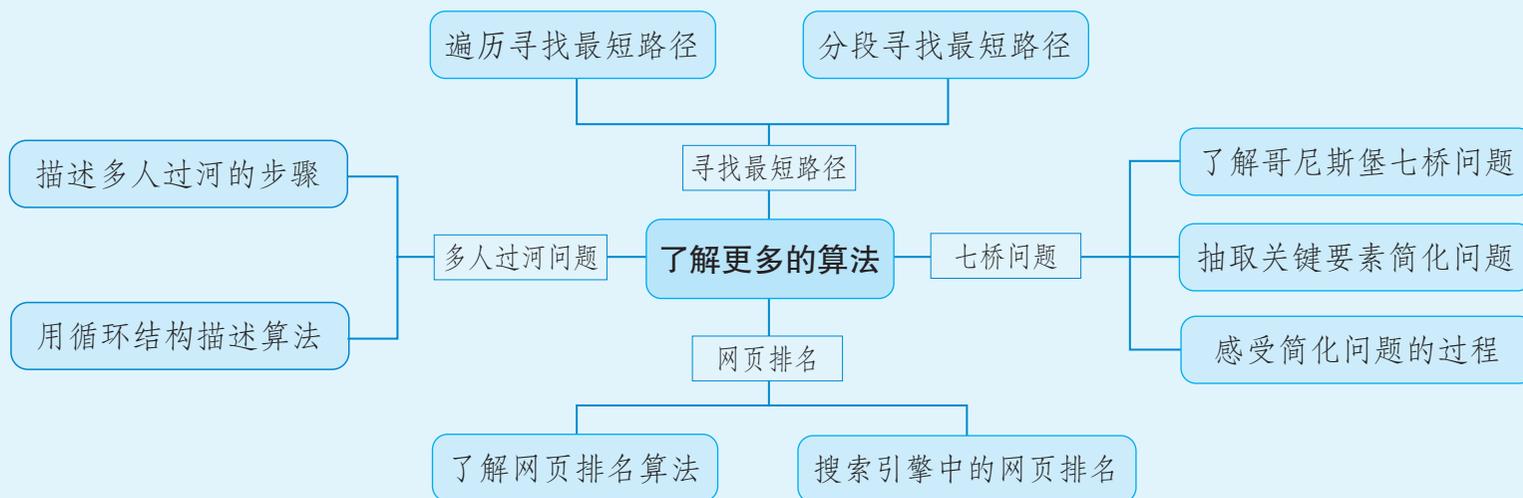
动态规划通过将原问题分解为若干个小问题，在逐步解决小问题的过程中解决原问题，这一过程中可以避免重复计算，从而提高算法效率，适用于具有重叠子问题和最优子结构的问题。智能规划利用计算机技术实现智能化的生产、运转过程，它涉及很多算法，如遗传算法、模糊理论和人工神经网络等。这些算法可以自动控制各类信息系统完成各种任务，提升工作效率和质量。路径规划算法用于确定从起点到终点的最佳路径，常用于智能汽车、智能机器人以及物流配送等。

计算思维主要分为四个基本步骤：分解、模式识别、抽象和算法。分解就是将复杂的、庞大的问题分解成若干小问题，并分别通过问题抽象、算法设计等逐个解决，这其中就包含了规划算法思想。

网页排名是指搜索引擎根据一系列因素对网页的重要性和相关性进行评估，并将这些网页按照一定的顺序排列出来。网页排名算法是用于评估网页并决定其在搜索引擎或其他应用软件中呈现顺序的规则和方法。

本单元从实际问题情境出发，在引导学生分解问题、简化问题的基础上，感受用相关算法解决问题的过程，培养解决问题的意识和能力。

## 【内容结构】



## 【教学实施条件】

1. 本单元可以在普通教室实施教学，配备教师用的计算机，能够运行配套资源中的游戏、程序等。
2. 配备教学演示与互动的设备，准备用于教学的图片、流程图、程序、学习任务单、实物等。

## 【学习目标】

1. 利用规划思维，将大问题分解为小问题，通过解决一个个小问题实现大问题的解决。
2. 通过分析问题，能够抽取关键要素来进行分析，从而简化问题并解决问题。
3. 利用规划思维，把全局问题转化为局部问题，通过逐步解决局部问题来解决全局问题。
4. 知道网页排名中的算法，了解提升网页价值的意义。
5. 了解网页排名、内容推荐等算法应用需要遵守的规范。

## 【评价建议】

下表可作为单元学习评价的参考。

评价内容	学生自评	教师评价
知道多人过河问题中体现的规划思想，即把大问题分解为小问题来解决	☆☆☆☆☆	☆☆☆☆☆
了解七桥问题中抽取关键要素的过程	☆☆☆☆☆	☆☆☆☆☆
知道解决七桥问题的本质是能否实现一笔画	☆☆☆☆☆	☆☆☆☆☆
了解寻找最短路径的算法思想——把全局问题转化为局部问题来解决	☆☆☆☆☆	☆☆☆☆☆
在寻找最短路径中，知道局部每个点最短用时的计算方法	☆☆☆☆☆	☆☆☆☆☆
知道网页排名算法以及提升网页价值的意义	☆☆☆☆☆	☆☆☆☆☆
知道搜索引擎中存在的网页排名算法	☆☆☆☆☆	☆☆☆☆☆
了解网页排名、内容推荐等算法应用需要遵守的规范	☆☆☆☆☆	☆☆☆☆☆
学习过程中，认真独立思考，积极与同学协作交流	☆☆☆☆☆	☆☆☆☆☆

## 第 24 课

### 多人过河巧安排

#### 学习目标

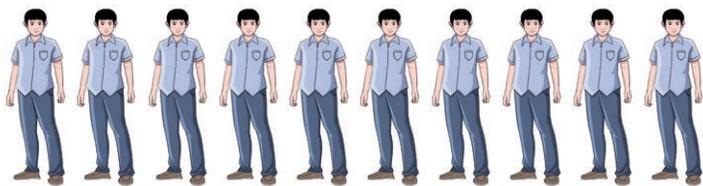
1. 尝试把一个大问题分解成小问题来解决，培养解决问题的能力。
2. 通过发现解决小问题的方案来解决大问题，初步了解规划算法的思想。

这一课通过一个由 10 个人组成的探险队过河的情境，了解问题分解的基本过程。

#### 学习活动 1

##### 规划探险队过河的方法

一支由 10 个成年人组成的探险队要过河，但河宽水深，没法游过去，附近也没有桥。这时，他们看到 2 个少年划着 1 条小船过来，就请求帮助。但这条船太小，每次只能承载 1 个成年人或者 2 个少年。



过河的问题情境示意图

如果 2 个少年与探险队成员都能单独划船过河，这支探险队如何过河？

教学重难点：

1. 把大问题分解为小问题的方法；
2. 通过发现解决小问题的方案来解决大问题。

教学时，要向学生说明，本案例是为分析问题而设计的情境。在实际生活中，未成年人划船时，需要在成年人的陪同下进行。

本课的问题初看起来有点复杂。教师带领学生分析问题时，要重视学法指导，把分析问题的过程和方法作为一个重点内容。当学生碰到类似问题时，也能用同样的方法来分析，逐步培养迁移解决问题方法的能力。

对问题的分析，可以从对文字的解读开始，给出分析的思路，让学生补充，引导他们根据问题找到准确的已知条件和目标。

除明显的已知条件，隐含的条件是什么？

很显然，就是这 10 个成年人只能一个一个地过河。

所以，解决问题的关键是什么呢？

如果 1 个成年人能成功过河，其他人也可以按照相同的方法过河。这就是把大问题分解成小问题的过程。

那么，如何解决 1 个成年人过河的问题呢？过河时 2 个少年的位置是怎样的？

解决问题的过程也是思维发展的过程。用图示的方法可以更直观地呈现解决问题的过程。

通过阅读问题要求，发现隐含的条件是：这10个成年人只能一个一个地过河。

这样，10个人过河的问题就变成了解决1个成年人过河的问题。

下面分析1个成年人过河的算法。

1. 2个少年划船到对岸。
2. 少年A划船回来，少年B留在对岸。
3. 1个成年人划船到对岸。
4. 少年B划船回来。

按上述方法操作后，1个成年人就到达了河对岸。

分析发现，可以用重复多次的方法来解决探险队过河的问题。

## 学习活动2

### 用流程图描述过河的过程

只需把1个成年人过河的步骤，循环10次，就可以把10个成年人送到对岸，2个少年也可以划船离开了。

下面用流程图来描述过河的过程。

首先，列出河两岸的所有状态以及可以转变状态的方式。

可以尝试将描述对象分为：成年人、少年A、少年B和船。

引导学生进行组合，看看如何转换成目标位置：1个成年人在对岸，2个少年仍然在原岸。

接下来，思考问题解决的步骤。分析过程可以根据学生的反应来灵活选择。提示学生重点关注状态转换的过程。

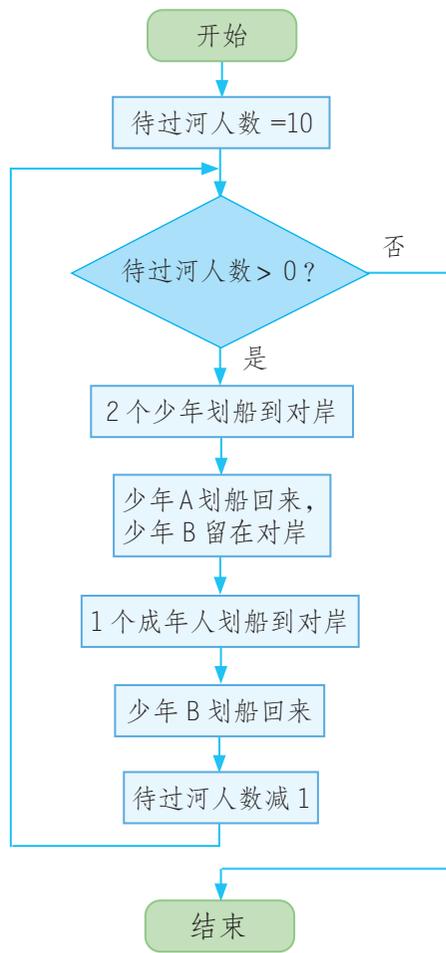
1. 只有让2个少年到对岸，才能继续。
2. 必须让1个少年回到原岸接成年人，否则将回到初始状态。
3. 只有让1个成年人划船到对岸，才能继续。
4. 对岸的少年回到原岸，2个少年汇合，就可以运送1个成年人到对岸。

1. 把大问题分解成小问题。有时每个小问题的解决方法是一样的，这里10个成年人过河问题转化成1个成年人过河问题，每个人的过河方式相同。

2. 规划的思想，就是引导问题解决朝着规划好的小问题方向发展。这样就可以通过逐个解决小问题来解决整个大问题。

3. 根据需求，小问题可以再继续分解，当问题分解到足够小时，解决的难度就会大大降低。

4. 如果问题分解过程中找到解决问题的规律，就可以依据规律采用合适的方法来解决。



算法的流程图

进一步思考:

1. 如果是20个成年人过河或者更多人过河, 如何修改算法?
2. 日常生活学习中, 还有哪些类似的问题?

总的看来, 本课内容在本单元中的关系如下。

与之前的循环结构算法相比, 这个算法的难点在于用什么步骤作为每一次循环要做的操作。关键点是把大问题分解成小问题的思想, 即运送10个成年人过河变成运送1个成年人过河。

与后续的寻找最短路径相比, 这个学习活动可以看作是两点之间的行动路径, 是一个简单的地点问题, 但对行动规则有较详细的要求。

引导学生从算法的角度, 思考更多人的过河问题, 其实是让学生感知在数量增大后用算法解决问题的优势。只需用循环结构反复操作, 不必改变循环体的操作内容, 就可以完成任务。

## 拓展与提升

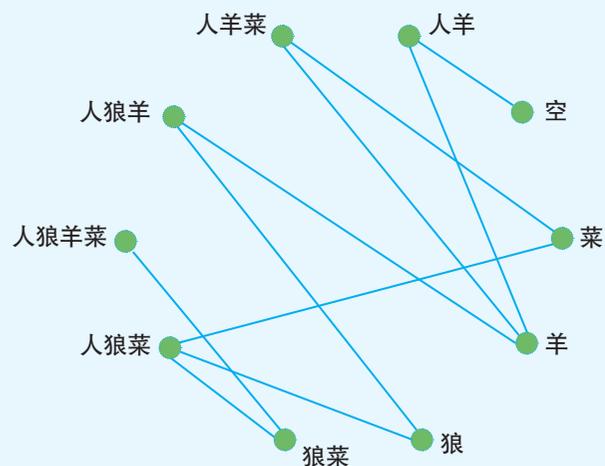
有一位猎人带着一只羊、一头狼和一筐白菜过河。河边有一条小船，但船太小，每次猎人只能载狼、羊和白菜三者中的一个过河。猎人不在旁边时，狼会吃羊，羊会吃白菜。

请问猎人怎样才能把狼、羊、白菜都安全运过河？

1. 这是另一个经典的过河问题，问题的关键是猎人不在时，不能让狼和羊、羊和菜在一起，所以这时只能让狼和菜在一起。

河边共有10种可能的状态：人狼羊菜、人狼羊、人羊菜、人羊、空、菜、羊、狼、狼菜、人狼菜。

以每种状态作为1个节点，状态之间如果可以转化，那么这2个状态对应的节点之间连接一条线。



猎人过河问题的方法描述

其中步骤最少的运法就是从“人狼羊菜”这个点到“空”这个点的最短路径。

## 第 25 课 有趣的七桥问题

### 学习目标

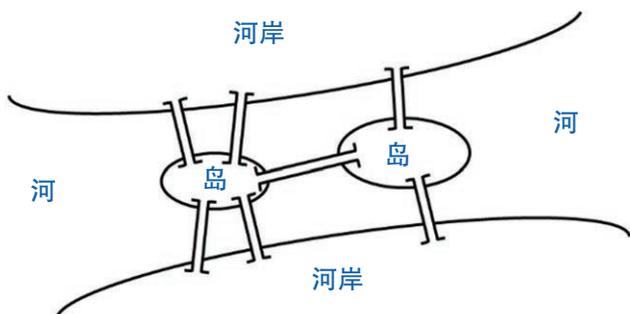
1. 认识哥尼斯堡七桥问题，能够通过分析问题抽取关键要素进行判断处理。
2. 知道哥尼斯堡七桥问题本质上是能否实现一笔画的问题，认识实现一笔画的判断方法。

这一课从经典的“哥尼斯堡七桥问题”引入，了解分析问题、抽取问题关键要素的过程。

### 学习活动 1

#### 认识哥尼斯堡七桥问题

18 世纪初普鲁士的哥尼斯堡，有一条河穿过，河上有两个小岛，一共有七座桥连接这两座小岛和河两岸。当地居民和游客都想尝试做到这样一件事：从一个地点出发，走过这七座桥，再返回起点，而且每座桥只经过一次。这就是经典的“哥尼斯堡七桥问题”。



哥尼斯堡七桥问题示意图

居民和游客都想尝试的这件事能否实现呢？

教学重难点：

1. 通过分析问题抽取关键要素进行判断处理；
2. 实现一笔画的判断方法。

哥尼斯堡现称为加里宁格勒，是位于波罗的海海岸的俄罗斯海港城市。

关于哥尼斯堡七桥问题，有的学生知道，但多数学生比较陌生。

教学时，先呈现故事情境并提出问题，让学生先进行分析并分享自己的想法。这类问题具有趣味性，学生比较感兴趣。

在学生分析思考之后，会发现很难实现。这时开始讲解这一问题的解决方法。

这里不要求学生获得肯定性的答案，只需引发学生思考和进一步探究的兴趣。

先来进行问题分析。

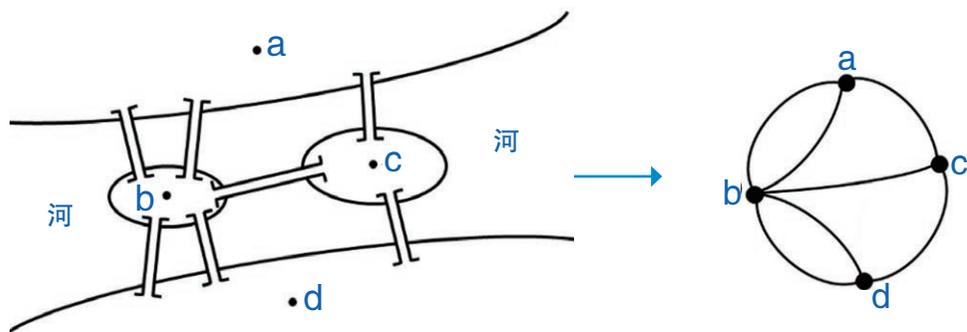
任务中共有两类描述对象：一类是桥，另一类是陆地——岛、两岸。桥共有7座，陆地共有4块。

从任意一个地点出发，每座桥只经过1次，并要求回到起点。

这样，根据给定的图形，问题转化为：能否画出一条路径，每两个地点的连线只通过一次，最后还回到起点。

事实上，后续故事是数学家欧拉巧妙地解决了这个问题。

欧拉认为：岛和岸都可以看作一个点，而桥则可以看成是连接这些点的一条线。他在这个地图上标记了a、b、c、d四个点，把这个地图简化成了一个图形，并给出了判断方法。



哥尼斯堡七桥问题的简化

如果想从一个点出发，经过所有的边，而且每条边只经过一次，再回到起点，那么每个点连接的边数必须是偶数。

然而，这个图上所有的点连接的边数都是奇数，因此，哥尼斯堡七桥问题是无解的，不可能实现。

通过分析讨论，引导学生观察发现，抽取问题的关键要素，这是一个难点，也是一个把问题进行抽象的过程，需要把整体问题简化为局部问题。

经过简单分析发现，桥本身并没有特别的研究之处，而陆地则有不同的描述特征，如所连接的桥数量。

学生思考分析之后，教师适时介绍后续故事，也就是欧拉给出的结论。

哥尼斯堡七桥问题看起来这样的简单，人人都乐意尝试，但都没有找到合适的路线。问题传开之后，欧洲许多有学问的人也参与思考，同样一筹莫展，于是有人想到了当时的数学家欧拉，请他帮助解决。

欧拉的问题简化过程其实就是对问题的抽象过程，通过把解决问题的关键要素抽取出来，简化问题，从而方便找到解决方案。

通过引导学生对照分析图形的简化，就可以感受抽象的过程。在这个过程中，去掉那些无关属性，抽取最重要部分——4个点和7条边。不必考虑地图的复杂性，只对4个点连接的7条边进行判断，就可以得出结论。

以上是一个实际问题转化为一个几何图形能否一笔画出的问题，即图形的一笔画问题。

## 学习活动2

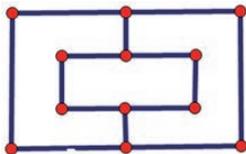
### 图形的一笔画分析

所谓图形的一笔画，主要指从图形的一个点出发，笔不离开图形的线条，连续画出整个图形，而且每条线条只能画一次，不能重复。

首先，能够实现一笔画的图形应该是连通图形。



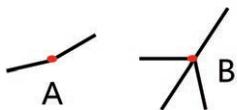
非连通图形



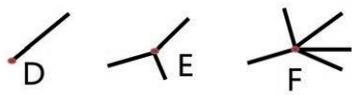
连通图形

其次，在能实现一笔画的图形中，有偶点和奇点。

偶点是与偶数条边相连的点。奇点是与奇数条边相连的点。



偶点



奇点

通过观察分析后发现一笔画图形具有以下规律。

1. 奇点个数为0的连通图形，通常是能实现一笔画的图形，可以任选一点为起点，起点和终点可以是同一点。
2. 奇点数为2、偶点数为任意数的连通图形，通常也是能实现一笔画的图形，可以选其中一个奇点作为起点，而终点必须是另一个奇点，即一笔画后不可以回到出发点。

问题转化后，通过对一笔画问题的分析，得出结论。

继续提出问题。

什么是一笔画？

什么样的图形可以一笔画出？

这里可以让学生自由发挥，说出自己的想法，不论对与错。

当然，多数学生不能完全回答以上问题，但或许会产生对问题解决的兴趣，教师可借此引入一笔画的相关知识。

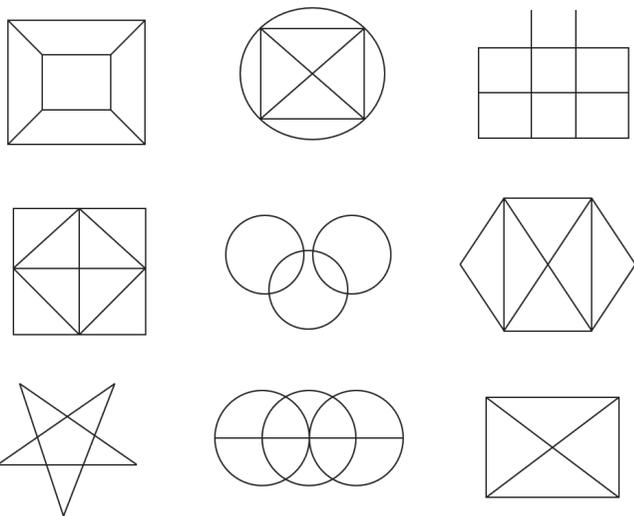
教学时，可以给学生提供更多的图形，引导学生分析能实现一笔画的规律，从中分析相关的判断条件，进一步认识判断一笔画的规则。

一笔画的判断看似简单，其实包含着相对复杂的条件，需要学生认真观察、分析和思考。

教学时要注意把握的关键点。

1. 将全局问题的解决简化为局部问题的求解。
2. 对待规划类问题，需要考虑问题描述的各个方面，从中找到对解决局部问题有用的关键信息来解决问题。
3. 给学生展示更多的案例，引导他们多练习多思考，可以提高他们将问题进行抽象的能力。

分析下面的图形能否实现一笔画。



9个不同的图形

实际应用中的许多规划问题，都可以转化为一笔画问题来解决。

在城市规划或道路网络设计中，一笔画可以用来检查是否存在一个路径，这个路径可以遍历城市的所有主要道路而不重复。这对于执行紧急任务的车辆（如消防车、救护车）的路径规划尤为重要。

在迷宫游戏设计中，可以使用一笔画来设计具有挑战性的迷宫。游戏时需要找到一条路径，能够遍历迷宫中的所有房间或通道而不重复。

在电路设计中，工程师需要确保电流能够流经每个必要的组件而不形成短路。一笔画有助于设计出最优的布线方案。

在计算机网络中，数据包往往通过不同的路径进行传输。一笔画可以用来分析、检测有效路径，使得数据包可以遍历网络中的所有节点而不产生冲突。

教师可以分小组布置任务，主要考查学生能否根据一笔画规律判断一笔画。

提示：从左上到右下，答案依次如下。

不能、不能、不能

不能、能、能

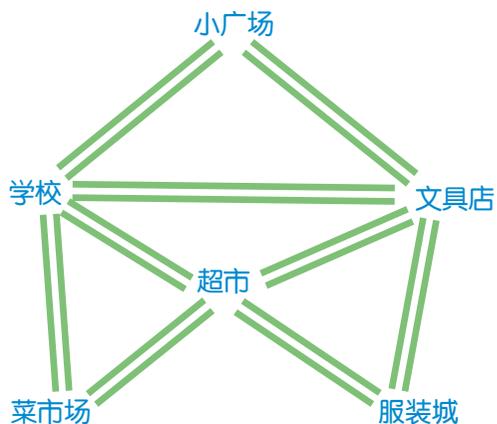
能、能、不能

教学时，依据学生的认知特点和本校的实际，可以适当列举其他的例子，同时进一步引导学生思考、发现，主动说出自己知道的一笔画应用实例。

这一课通过分析问题，抽取关键要素进行分析，其实延续了前面所学的问题分解思想——把大问题分解为局部小问题来解决；通过分析图形中的点线关系解决问题，为后续寻找最短路径的学习做准备。

## 拓展与提升

一辆洒水车要给某城市的街道洒水，街道地图见下图。请为洒水车设计一条洒水路线，使洒水车能走过所有道路，但不重复走任何街道，还能回到出发点。



一个街道示意图

首先，将街道地图抽象成简单的点线图形。

其次，判断能否实现一笔画。

最后，进行路线判断与选择。

例如，小广场——学校——菜市场——超市——服装城——文具店——超市——学校——文具店——小广场。

## 第 26 课 寻找最短的路径

### 学习目标

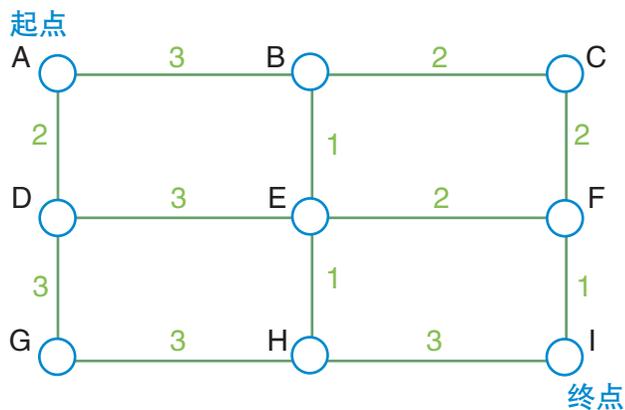
1. 进一步了解规划算法的思想，体会把全局问题分解为局部问题的过程。
2. 通过寻找最短路径的算法描述，初步了解路径规划算法的应用。

日常生活中，人们出门时，常常用导航软件查询线路并选择到达目的地的方式。本课通过在一个简单地图上寻找最短的路径，体会相关的算法。

有一个街道地图，共有9个地点，路线正好能形成2行2列的网格。其中，每个点可以对应到不同地点。例如，起点是家，终点是学校，中间有超市、体育馆、公园、书店、博物馆等。

每条边上的数代表走这条路需要用的时间，如3代表3分钟。这些道路都是单行线，在图上只能从左往右走或者从上往下走，不能反方向走。

计算从起点走到终点的最短时间。



有 9 个地点的街道示意图

教学重难点：

1. 寻找最短路径的算法描述；
2. 路径规划算法的应用。

这一课继续学习把全局大问题转变成局部小问题的过程与方法。

其中，寻找最短路径可以看作是全局问题，而到达每一个地点的最短时间，就是其中一个小问题。

到终点的最短路径就是问题的答案。

1. 对象分析：一共有两类描述对象，一类是代表所需时间的边，另一类是用边连接的点，也就是地点。边共有 12 条，点共有 9 个。
2. 要求：从起点出发到终点结束，只能走下方或右侧的边。

1. 学生往往认为，解决任务最简单的方法就是列举出所有的行走路径，计算时间后找到用时最少的路径。

存在的问题：种类多，容易有遗漏。

2. 将全局问题分解为局部问题。

计算出从起点到每一个点的最少时间就是小问题。最终求得到终点的最少时间，即是全局问题的解决。

## 学习活动 1

### 用枚举法寻找最短路径

先来尝试用枚举法遍历所有可能的路径。

$$A \rightarrow B \rightarrow C \rightarrow F \rightarrow I \quad 3 + 2 + 2 + 1 = 8$$

$$A \rightarrow B \rightarrow E \rightarrow F \rightarrow I \quad 3 + 1 + 2 + 1 = 7$$

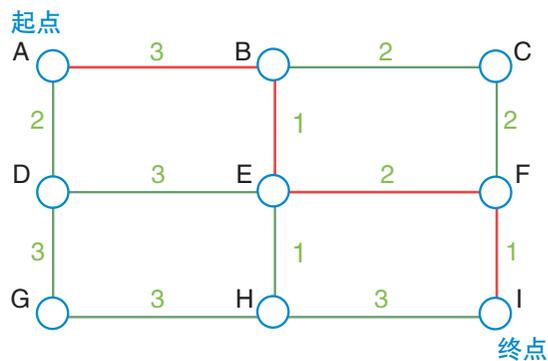
$$A \rightarrow B \rightarrow E \rightarrow H \rightarrow I \quad 3 + 1 + 1 + 3 = 8$$

$$A \rightarrow D \rightarrow E \rightarrow F \rightarrow I \quad 2 + 3 + 2 + 1 = 8$$

$$A \rightarrow D \rightarrow E \rightarrow H \rightarrow I \quad 2 + 3 + 1 + 3 = 9$$

$$A \rightarrow D \rightarrow G \rightarrow H \rightarrow I \quad 2 + 3 + 3 + 3 = 11$$

这样获得的路径是  $A \rightarrow B \rightarrow E \rightarrow F \rightarrow I$ ，用时7分钟。



获得的最短路径

人工用枚举法遍历寻找路径时，随着地点的增加，路径数量会增加更多，逐个枚举就会很耗费时间，而且很容易遗漏一些路径。

因此，要用一个计算次数尽可能少，且确保不会遗漏路径的算法。

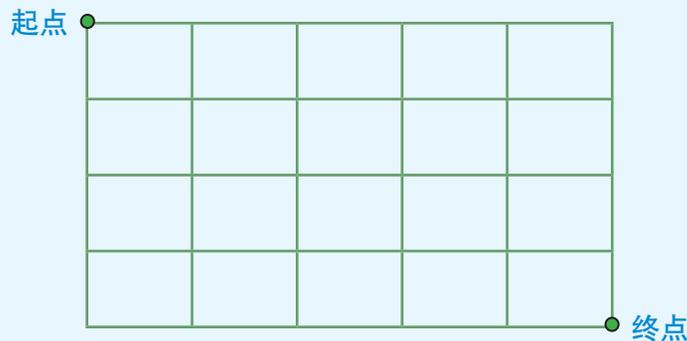
提问：根据给定的图形，你能列举出所有符合要求的路径并找到用时最少的一条路径吗？

教学时，建议让学生自主思考完成，教师补充全部路径，最短路径就显而易见了。

继续提问：这样的解法有没有问题呢？

问题比较明显：随着地点的增加，路径的数量会快速地增长，如果人工用这种方法操作，就会很耗费时间，而且容易遗漏路径。

例如，用遍历的方法列举以下路径，你还能完全列举出来吗？



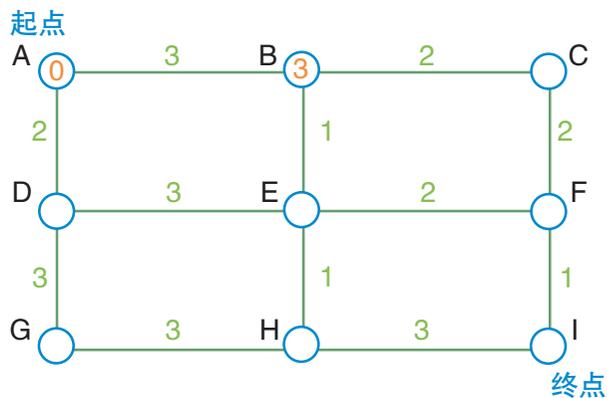
有更多个地点的街道示意图

这里可以留一些时间，由学生尝试写出所有路径。当他们遇到问题时再帮助解决，让他们感受到选择合适方法的重要性。同时引出后面的内容。

## 学习活动 2

### 用分段用时寻找最短路径

下面把计算整个地图最短路径的用时，转变为计算到具体一个点的最短路径的用时。用圆圈中的数表示从起点到该点的最短用时。



转变思路后，到一个点的用时最多有两个来源。

一是：上方节点用时+上方路径用时

二是：左方节点用时+左方路径用时

如果一个点有两个来源，那么选其中用时较少的一个。

具体步骤如下。

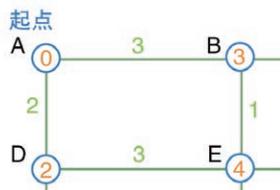
第 1 步：计算第一个局部，A、B、D、E 四个点。

(1) 起点 A 的用时记为 0

(2) B 点只能从 A 点向右，最短路径用时为：

左边 A 点的用时+A 点到 B 点的用时

可以表示为： $A + (A \rightarrow B) = 0 + 3 = 3$



规划算法的主要思想是将全局问题转化为局部问题，并通过解决局部问题，逐渐解决全局问题。

对于本课的问题来说，寻找最短路径是全局问题，局部问题就是计算从起点到其中每个点的最短用时。这样，通过计算出每个点的最短用时，最终获得从起点到终点的最短路径。

这一课本质上是动态规划算法，而计算最短路径是动态规划算法最简单的应用场景。本课局限在小规模数据上，是为了便于学生理解，实际应用中往往更加复杂。

动态规划通常用于解决包含重叠子问题和最优子结构的问题。例如，用递归算法自顶向下求解问题时，每次产生的子问题并不总是新的，有些子问题会被重复处理多次。动态规划对每个子问题只处理一次，然后将结果保存起来，当再次需要处理这个子问题时，只需直接查看保存的结果。

(3) D点只能从A点向下, 最短路径用时为:

$$A + (A \rightarrow D) = 0 + 2 = 2$$

(4) E点可以从B点向下, 也可以从D点向右, 分别表示为:

$$B + (B \rightarrow E) = 3 + 1 = 4$$

$$D + (D \rightarrow E) = 2 + 3 = 5$$

选较短的路径用时:  $B + (B \rightarrow E) = 3 + 1 = 4$

第2步: 计算第二个局部C点和F点。

(1) C点只能从B点向右, 最短路径用时为:

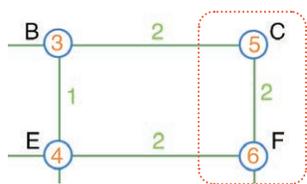
$$B + (B \rightarrow C) = 3 + 2 = 5$$

(2) F点可以从C点向下, 也可以从E点向右, 分别表示为:

$$C + (C \rightarrow F) = 5 + 2 = 7$$

$$E + (E \rightarrow F) = 4 + 2 = 6$$

选较短的路径用时:  $E + (E \rightarrow F) = 4 + 2 = 6$



第二个局部

第3步: 计算第三个局部G点和H点。

(1) G点只能从D点向下, 最短路径用时为:

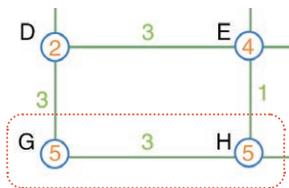
$$D + (D \rightarrow G) = 2 + 3 = 5$$

(2) H点可以从E点向下, 也可以从G点向右, 分别表示为:

$$E + (E \rightarrow H) = 4 + 1 = 5$$

$$G + (G \rightarrow H) = 5 + 3 = 8$$

选较短的路径用时:  $E + (E \rightarrow H) = 4 + 1 = 5$



第三个局部

最短路径算法是一种用于在图形结构中找到两个节点之间最短路径的算法。在图形结构中, 通过计算从起点到相关节点的距离、时间或成本等指标来确定最短路径。本课以计算时间为例。

教学时, 引导学生分析出第一个局部后, 其余三个局部可以组织学生以小组形式来进一步分析, 让学生自己获得答案。

动态规划在演算过程中, 可能会出现多个选择, 需要进行局部的最优选择。实际解决问题时, 局部的范围也可以根据问题解决的需要来确定。可以引导学生尝试选择不同的局部范围。

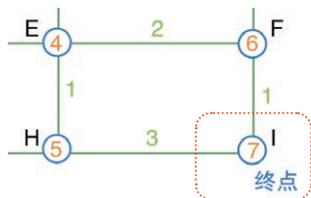
第4步：计算第四个局部，只剩下I点。

I点可以从F点向下或者从H点向右。

$$F + (F \rightarrow I) = 6 + 1 = 7$$

$$H + (H \rightarrow I) = 5 + 3 = 8$$

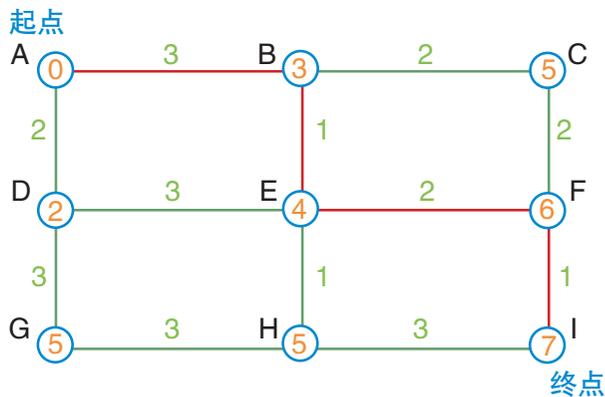
选较短的路径用时： $F + (F \rightarrow I) = 6 + 1 = 7$



第四个局部

最后获得结果，从起点到终点最短用时为7分钟，路径为：

$A \rightarrow B \rightarrow E \rightarrow F \rightarrow I$



路径规划算法在现实生活中有广泛的应用，举例如下。

**导航系统：**路径规划算法可以帮助导航系统找到两个地点之间的最短路径，并标注相应的路线，从而提供导航服务。

**物流配送：**在物流配送过程中，路径规划算法可以帮助物流人员确定最优的配送路线，从而节约时间和成本；还可以帮助物流企业规划仓库的位置，让仓库与客户的距离更近，提高配送效率。

**电力网络：**电力网络中的电线杆和变电站可以看作是节点，它们之间的电线可以看作是路径，路径规划算法可以帮助确定节点之间的最短电线布局，从而降低电力损耗和成本。

本课重在分析算法的思路，这个算法不必用程序来实现。如果要用程序实现，建议作为拓展内容，同时需要做一些转换。

方法是：在3行3列的九宫格内填写数，代表在这块区域所用的时间，代替在各条边上填写数。这样可以比较容易地采用二维数组来保存数据。

在数学课中，学生已经学过“位置”表示，如用数对(2, 3)来表示教室座位上第2列、第3行的同学。因此，用二维数组表示地图上的地点，学生是能够理解的。

配套资源中提供了一个实现本课寻找最短路径的参考程序，供教师参考。在参考程序中，使用二维列表存储两个地点之间的最短距离，不能直接连接的地点用10 000表示。通过枚举任意两个地点之间的中间点，也就是哪个点可以使两个地点的距离得到优化，选择距离最短的点作为中间过程点，最终计算出两个地点之间的最短路径。

最短路径算法有多种实现方法，适用于不同的场景和需求。

此外，社交网络中的用户可以看作是节点，他们之间的关注和好友关系可以看作是边。最短路径算法可以帮助社交网络推荐好友或者关注对象，让用户之间的连接更加紧密。

路径规划算法还常用于城市规划、交通网络优化、通信网络设计等领域，帮助人们找到最优的路径，从而优化资源分配、提高系统效率。

## 拓展与提升

篮球赛中重要的就是队员互相配合。现在知道对方球队有著名的三人组，这三个人之间配合相当默契。假设三人分别为球员 A、球员 B、球员 C，在进攻时他们组成三角形进攻。请帮助我方球队分析，如果在一轮进攻中，球员 A 拿到球，然后把球传给球员 B 或球员 C，三人之间一共有 10 次传球，那么第 10 次传球仍然能传到球员 A 手中的可能性有多少种？

打开配套资源中的程序，依据程序的提示，观察、运行程序，分析程序与算法的关系，感受利用算法求解问题的过程。

三个球员组成三角形结构，从传球路径上看，存在以下关系。

球员 A 可以从球员 B、球员 C 处获得球。

球员 B 可以从球员 A、球员 C 处获得球。

球员 C 可以从球员 A、球员 B 处获得球。

为此，其中某个球员获得球的可能性就是上一次传球时另外两个球员能获得球的数量之和。利用循环结构重复计算指定的传球次数，就可以获得球传到球员 A 手中可能的次数。

10 次传球过程三个球员可能拿到球的次数分析

传球次数	1	2	3	4	5	6	7	8	9	10
球员 A	0	2	2	6	10	22	42	86	170	342
球员 B	1	1	3	5	11	21	43	85	171	341
球员 C	1	1	3	5	11	21	43	85	171	341

五年级学生求解这道题是有难度的，但这里不要求学生用数学方式求解。可以启发学生：在求解问题时，如果自己用数学方法难以解决，可以尝试让计算机来帮忙解决，只需依据规则设计算法，再编写程序让计算机执行就能顺利解决问题。

## 第 27 课

### 网页排名有策略

#### 学习目标

1. 初步了解网页排名算法，知道搜索引擎中存在的网页排名算法。
2. 了解提升网页价值的意义，知道网页排名、内容推荐等算法应用应遵守的规范。

虽然我们很少直接使用网页排名算法，然而它就隐藏在互联网的背后，在我们日常使用互联网时发挥着重要作用。例如，当我们用搜索引擎搜索信息时，搜索引擎会根据网页排名算法，把内容相关度较高的网页显示在前面，帮助我们更快地找到所需信息。

#### 学习活动 1

##### 找出被链接最多的网页

选出 4 位同学，每人发一张卡片，分别是 A 卡、B 卡、C 卡和 D 卡。每位同学在卡片上写上其他 3 位同学中，自己有联系方式的卡片号。

为了方便对比，我们把这些卡片看成是网页。例如，A 网页上写着“C、D”，表明 A 同学有 C 同学和 D 同学的联系方式。

同样地，把这些联系方式看成链接，因此 A 网页上有 C 和 D 两个链接。

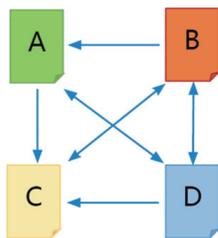
假设其他 3 个网页上的链接情况如下。

B 网页上写着“A、C、D”。

C 网页上写着“B”。

D 网页上写着“A、B、C”。

使用网页排名算法找出被链接次数最多的网页。



4 个“网页”的关系

教学重难点：

1. 搜索引擎中存在的网页排名算法；
2. 提升网页价值的意义。

本课安排了三个学习活动，教学时可以选取其中两个作为重点展开，不必面面俱到。这样的安排是为了便于分层教学。

在任何场合、利用任何设备登录互联网，都会用到算法。从日常生活中刷到的短视频、看到的广告推送、购物网站推荐的商品、搜索引擎展示的网页、导航的最优路线、处理交易的支付系统等，算法都在互联网中默默地工作，发挥着重要的作用。

为了便于学生理解网页排名算法，通过简单的情境扮演来模拟网页排名算法的步骤，帮助学生认识和理解。由于实际应用中算法的计算过程非常复杂，五年级学生难以理解，此处把算法进行了简化，用类比方式帮助学生理解这一过程。

活动过程如下。

第1步：准备一个表格，用来记录网页被链接的情况。

第2步：查看A网页，其中写着C和D，在记录表中为C和D记录1，表示C和D分别被链接一次，同时记录链接的网页为A。

第3步：继续查看B网页、C网页和D网页，在记录表中记录它们被链接的次数和链接的网页。

网页被链接情况记录表

同学	被链接的次数	被链接的网页
A	2	B、D
B	2	C、D
C	3	A、B、D
D	2	A、B

所有网页都被检查过后，可以看出，被链接次数最多的是C网页，表示C同学被记录联系方式最多。

通过前面的学习，对于情境不太复杂的问题来说，学生已经具备一定的分析能力。这里仍然可以采用问题引导与分析的方法来展开学习。例如，可以提出以下问题进行引导。

1. 你是否在前面学习过与网页排名相关的知识？
2. 所学知识在学习网页排名算法时是否能起到一定的作用？
3. 学习网页排名算法，能否借鉴一些经典算法的知识？

带领学生对问题进行分析，其实也是培养学生学会思考问题的过程，需要将学生从已知带入未知，或者从未知回溯到已知。

通过阅读任务，能够找到已知条件。

确定问题目标：从4个网页中找出被链接最多的网页。

这个活动采用学生比较容易接受的表格记录方式，能够直观地记录数值，操作较简单。

适当总结问题解决的过程，用自然语言描述算法步骤。

第1步：在未检查的网页中选定一个网页。

第2步：统计选定网页中的链接次数。

第3步：重复以上两步，统计其余网页被链接的次数。

第4步：输出被链接次数最多的网页编号。

## 学习活动2

### 搜索引擎中的网页排名

打开一种搜索引擎，输入关键词，打开并观察网页中显示的内容。

操作后，搜索引擎会根据输入的关键词找到大量相关的网页，并按照相关排名算法的规则进行排序。

在这个过程中，搜索引擎为我们做了哪些工作呢？

第1步：搜索引擎会记录相关网页的“票数”，如被其他网页链接的次数。“票数”越多，表明这个网页越被关注。

第2步：搜索引擎会把那些“票数”较多的网页排列在前面，让搜索信息的人先看到这些网页。

第3步：搜索引擎会将其他搜索到的相关网页，按照“票数”多少依次排列显示。

实际应用中，搜索引擎往往会提前搜索很多常用的词、热点事件等，并提前做好网页排名。这样，在我们进行搜索时，就能更快地提供搜索结果。

在很多社交媒体的应用软件中，也会采用类似的算法。当我们输入关键词搜索信息时，应用软件会根据网页排名算法来显示搜索结果。那些被更多人转发、点赞或评论的内容，往往会出现在首页，因为这些内容被认为更受欢迎，是多数人需要的内容。

教学时，可以引导学生用不同的搜索引擎来搜索同一内容，观察搜索结果的异同，体会不同的搜索引擎所使用的排名算法也会有所不同。

这个活动可以帮助学生更加具体地了解网页排名算法的作用，知道在日常使用搜索引擎时，搜索结果的排名顺序与算法有关。同时，也可以强调网页排名算法的重要性，它帮助我们更快地找到有用的信息，提高了我们使用互联网搜索信息的效率。

### 学习活动3

#### 了解网页排名算法

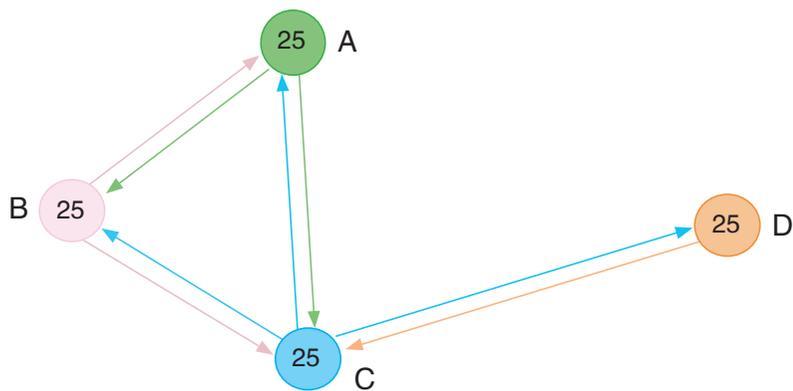
搜索引擎通常以关键词和网页内容的关联性来决定搜索结果的排列顺序，这种方法没有考虑网页是否包含更有价值的内容，因此搜索精度较低。

网页排名算法的出现，使搜索引擎的搜索性能变得更加快速、精准。网页排名算法通过大量的计算，利用网页链接数量和结构判断网页价值。目前，决定搜索引擎搜索结果排序的已不仅仅是网页排名这一种算法，通常会用多种算法来进行计算和判定。

网页排名算法有很多，但较有名的是PageRank 算法。

下面通过模拟100个访问者的上网行为，观察他们最后停留在哪个网页上，从而确定吸引访问者最多的网页，了解PageRank 算法。

第1步：把100个访问者平均分配给4个网页。



平均分配4个网页的访问者数量

这里可以继续沿用前面的操作作为引入，通过演示在搜索引擎中输入关键词，呈现搜索结果的网页，继续向学生提问。

搜索引擎搜索到相关网页后，具体先后顺序是怎么确定的？

引导学生对熟悉的网页搜索进行思考，从而引出主题，自然而然地过渡到认识网页排名算法，知道它是隐藏在互联网背后的常用算法之一。

PageRank 算法于1996年被提出。当时，有研究人员发现互联网上有很多网页，要找到最有用的信息，需要一种方法来判断哪些网页更重要、更有价值。为解决这个问题，他们想到了一种“投票”系统，认为一个网页被其他网页链接次数越多，这个网页就越重要。

在PageRank 算法中，决定一个网页排名的因素如下。

#### 1. 被链接的次数

被其他网页链接的次数越多，该网页被访问的可能性就越大。

#### 2. 链接强度

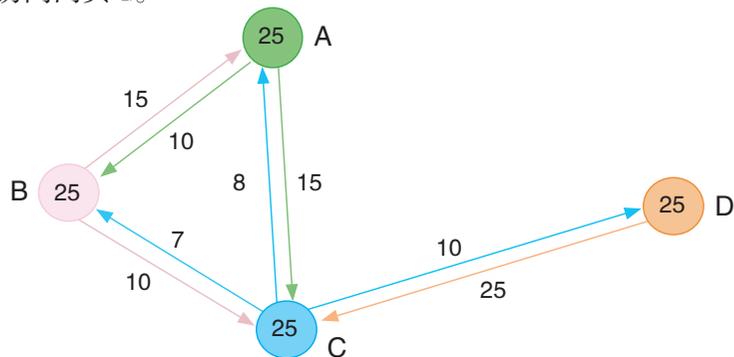
网页相关链接被访问次数越多，流量增加的可能性就越大。

#### 3. 链接来源

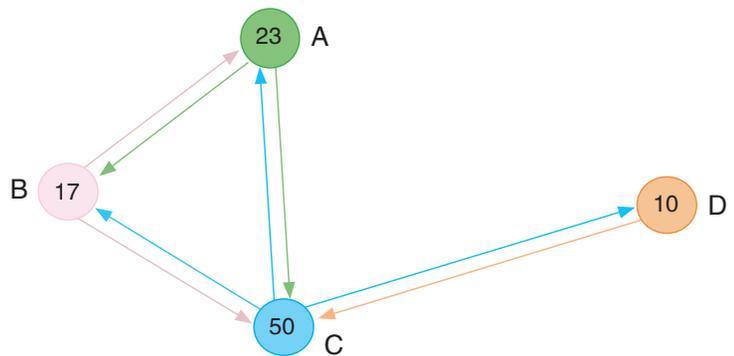
如果被其他有较高排名的网页链接，那么该网页的排名也会随之提升。

第2步：根据链接情况为每个网页重新分配访问者。

例如，在A网页的访问者中，有10个访问者会访问网页B，有15个访问者会访问网页C。



各条边上显示进出各个网页的访问者数量



重新分配后的访问者人数

重新分配访问者后，网页A还有23个访问者，其中15个来自于网页B，8个来自于网页C。

网页B只剩下17个访问者，其中10个来自网页A，7个来自网页C。

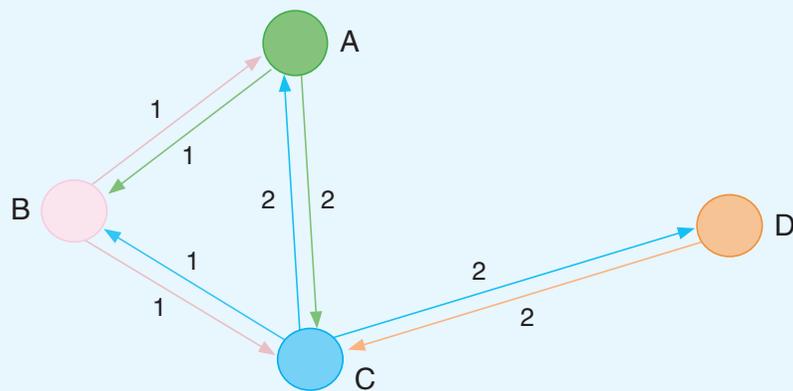
网页C增加到了50个访问者，其中25个来自网页D，15个来自网页A，10个来自网页B。

网页D只剩下10个访问者，全部来自网页C。

下图展示了 PageRank 算法的一般原理。

节点代表网页，边代表超链接。超链接的权重越大，表示链接强度越强，则其箭头所指方向的流量就会越大。

可以看到，对于网页A的访问者而言，访问网页C的可能性是访问网页B的两倍，而访问网页D的可能性为零。



PageRank 算法的一般原理示意

尽管 PageRank 算法易于使用，但它也有局限性，那就是偏向于旧节点。如果一个新网页包含非常好的内容，但一开始时访问人数很少，那么它的排名就较低，这有可能导致该网页不在推荐之列。为了避免出现这种情况，要定期更新排名，让新网页随着自身知名度的提高获得提升排名的机会。

然而，这种排名的偏向有时却是有益的，尤其是对那些有着长期良好影响力的网页。这表明，一个算法的局限性在某种情况下可能正是它的优点。因此，要多角度多层面地分析一个算法。

第3步：为了得到每个网页的排名，通过迭代计算与优化，重复上述分配过程，直到每个网页的访问者人数不再发生改变。

每个网页的最终访问者人数体现了该网页的PageRank排名，即访问者越多，排名就越高。

使用PageRank算法进行排名时，迭代计算是一个关键步骤。

在每次迭代中，都会根据上一步计算出的数值来重新计算每个页面的数值。通过多轮的反复迭代计算，直到所有页面的数值不再发生明显变化或达到预设的迭代次数，就得到每个页面最终的排名数值。

通过迭代计算，PageRank算法能够量化相关内容在整个网页中的重要性或影响力，并根据这些值对页面进行排名。

讨论交流：如何辩证地看待网页排名、内容推荐等算法的应用？这些算法对社会生活有什么影响？

## 拓展与提升

1. 上网搜索“网页排名算法的意义”，进一步了解有关的内容。
2. 配套资源中有实现本课介绍的4个网页排名的参考程序，打开程序，观察并运行，进一步体会网页排名算法与程序实现的关系。

总之，PageRank算法根据链接的数量、强度以及来源对网页中的节点进行排序，有助于找出网络中价值较高的网页。

问题讨论活动主要引导学生从网络排名算法切入，关注算法应用在给社会生活带来方便的同时，也可能带来负面的影响，如损害网络资源的公共利益、泄露隐私信息、危害公共安全等。

教师可以提前设计好几个问题，引导学生以小组形式自主探究，同时每个小组可以把主要观点整理成文档展示。

算法应用在给社会发展注入新动能的同时，算法歧视、大数据杀熟、诱导沉迷等算法不合理应用导致的问题也深刻影响着正常的社会秩序。

教学时，可以引导学生阅读相关国家文件规定。例如，自2022年3月1日起施行的《互联网信息服务算法推荐管理规定》要求：坚持正确导向，算法向上向善；坚持风险防控，推进算法规制；坚持规范发展，防止算法滥用。

规定还要求：不得利用算法推荐服务诱导未成年人沉迷网络，应当便于老年人安全使用算法推荐服务，应当建立完善平台订单分配、报酬构成及支付、工作时间、奖惩等相关算法，不得根据消费者的偏好、交易习惯等特征利用算法在交易价格等交易条件上实施不合理的差别待遇等。

1. 建议教师教学前先搜索并查看获取的内容，以便教学时给学生提供适当的搜索关键字，让学生能够更准确地获取相关内容。
2. 引导学生从功能实现上分析程序与算法的对应关系，不必深究每条程序语句的用法。

### 【引言】

人工智能正逐步走进我们的生活。在各种各样的人工智能应用中，算法起到了关键的作用。人工智能中的算法是实现各种智能功能的核心，它们通过分析和处理数据，为社会生活带来更多的便利和创新。

人工智能算法能够通过分析大量数据，进行自主决策和预测。例如，在智能家居中，算法可以根据用户的历史行为和习惯，自动调整与控制智能冰箱、智能电视、智能洗衣机、智能灯具、智能厨具等家居设备的设置，提高使用的舒适性和便捷性。

图像识别、语音识别等算法使得人工智能系统能够准确地识别图像中的物体、人物以及语音中的指令，便于进一步的处理与控制。这些算法在安全防范、医疗健康、自动驾驶、工业生产、农业种植等领域都有广泛的应用。例如，广泛使用的人脸识别、物体检测等，都是通过图像识别算法实现的。

自然语言处理算法使得人工智能系统能够理解和生成自然语言文本，从而实现智能客服、智能翻译等功能，提高了人机交互的效率，使人们能够自然地与机器人聊天。

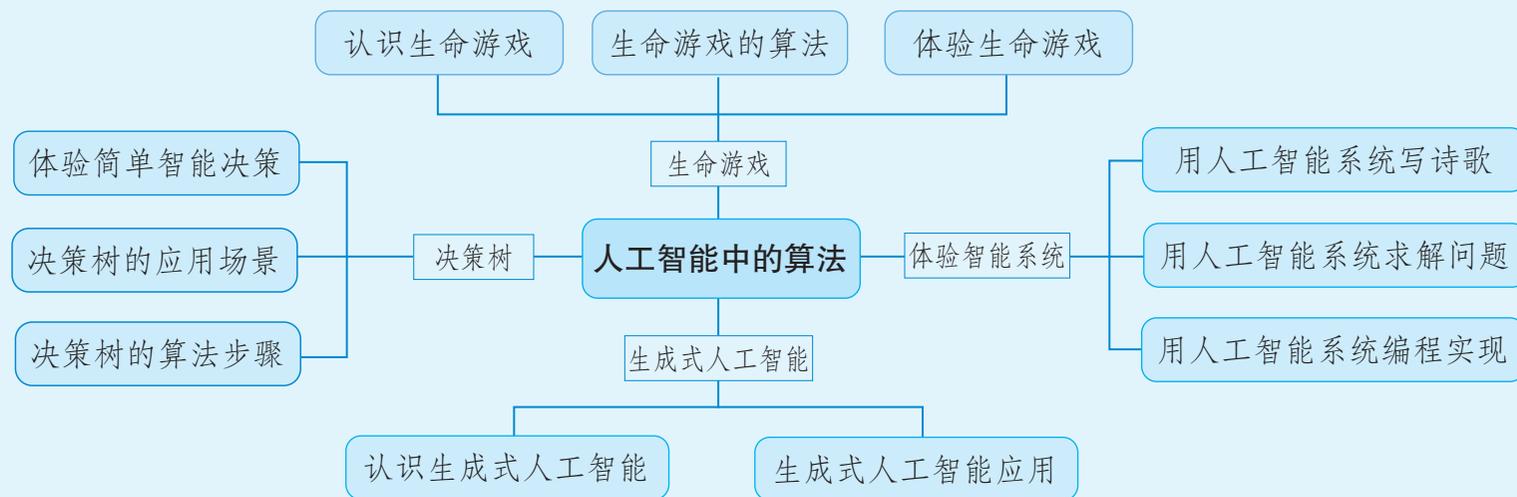
各种优化与搜索算法，在人工智能中用于解决各种优化问题和寻找最优解，在机器人路径规划、物流管理等方面发挥着重要作用。

生成式人工智能是一种可以生成新的创意内容的算法，包括文本、图像、音频、视频等各类作品。生成式人工智能模型基于大规模数据进行训练，通过学习数据中的模式并使用这些模式生成新的输出。生成式人工智能正在加速工作与创新的进程，带来更多自动化服务，提高社会生产力，推动社会的进步。

随着技术的不断发展与创新应用，新的算法和模型将不断涌现，为人工智能的应用提供更加广阔的发展空间。

在前面的学习中，我们已经认识了很多算法，本单元从应用的角度，引导学生了解人工智能领域中一些算法的初步知识。

## 【内容结构】



## 【教学实施条件】

1. 本单元建议在具备运行Python环境的信息科技实验室里教学，能够运行配套资源中的游戏、程序等。
2. 配备教学演示与互动的设备，准备用于教学的图片、流程图、程序、学习任务单、实物等。

## 【学习目标】

1. 初步认识决策树，了解决策树算法的思想。
2. 了解决策树算法的应用场景，能够创建简单的决策树。
3. 通过利用生成式人工智能工具来解决简单的问题，体验生成式人工智能工具的用法。
4. 尝试利用生成式人工智能工具实现与验证所学的算法。
5. 了解算法自主可控的重要性。
6. 了解生命游戏，知道其中的算法规则。
7. 培养合理利用技术解决问题的思维方式和能力。

## 【评价建议】

下表可作为单元学习评价的参考。

评价内容	学生自评	教师评价
初步认识决策树	☆☆☆☆☆	☆☆☆☆☆
针对具体情境，能描述决策树算法的应用场景	☆☆☆☆☆	☆☆☆☆☆
能创建简单的决策树解决问题	☆☆☆☆☆	☆☆☆☆☆
知道生成式人工智能工具并用来辅助自己学习	☆☆☆☆☆	☆☆☆☆☆
尝试用生成式人工智能工具对所学算法进行实现与验证	☆☆☆☆☆	☆☆☆☆☆
了解生命游戏，知道其中的算法规则和进化规律	☆☆☆☆☆	☆☆☆☆☆
对从算法描述到程序实现有了进一步的认识	☆☆☆☆☆	☆☆☆☆☆
了解算法自主可控的重要性	☆☆☆☆☆	☆☆☆☆☆

## 第 28 课

### 认识决策树算法

#### 学习目标

1. 初步体验智能决策的过程，了解决策树算法的应用场景。
2. 通过实例分析，认识决策树算法的基本步骤，体会决策树算法发挥的作用。

人工智能在我们的生活中发挥着越来越重要的作用。人工智能应用中使用了很多算法，决策树算法是其中非常重要的一种。

#### 学习活动 1

##### 初步体验决策过程

小智的爸爸准备给小智买一个电话手表，正在征求他的想法。以下是小智和他爸爸的对话。

爸爸：你喜欢什么颜色的电话手表？

小智：我喜欢深色的。

爸爸：黑色、蓝色还是深灰呢？

小智：我想还是蓝色更好。

爸爸：你想要游泳时也能戴吗？

小智：当然啦，我游泳时摘下来会弄丢的。

爸爸：你还有什么要求？

小智：我要拍照方便、充电后能长时间使用。

爸爸：好的，我知道了。

在以上情境中，实际上已经包含一个简单的决策过程。

爸爸根据小智的回答，知道了小智的要求：小智想要一个防水性能好、前后都有摄像头、待机时间长的蓝色电话手表。



电话手表示意图

教学重难点：

1. 决策树及其作用；
2. 决策树算法的应用场景。

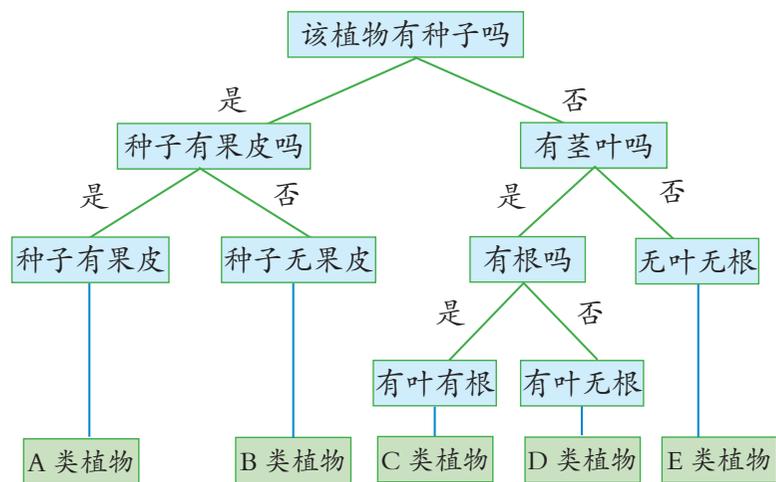
从生活情境出发，展现人们日常进行简单决策的一般过程，让学生对隐藏在思考过程中的决策分析有一个初步认识，再引导学生归纳这个过程中存在的决策条件。

这类来自生活情境的案例，先不必按照决策树算法来分析，这里的关键点是一步一步思考并决策的过程。也就是用贴近学生实际生活的案例来引发学生思考，不要求按照决策树严谨的“是”与“否”的判断过程来分析。

这类案例在现实生活中还有很多，教学时可以依据学生的认知基础和本地的生活实际，灵活取材，选用更合适的案例。

在人工智能应用中，决策树算法能够通过类似于人类决策的方法，来解决问题和进行预测，为人们面对各种复杂问题时提供了一种简单、高效且可解释的解决方案。

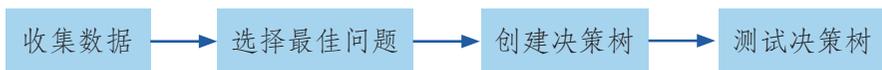
那么，什么是决策树呢？下面就是一个简单的分类决策树。



一个简单的植物分类决策树

通常，决策树的每个分支只存在两个答案，比如“是”或“否”。如果有两个以上的答案，比如“是”“否”或“有时”，可以沿着分支继续添加更多的分支，形成不同层次的分类。

决策树算法的步骤很简单，主要步骤如下。



这个分类决策树，从植物是否有种子、有茎叶、有根等进行分类，都是学生日常观察所见，学生通常都能理解。但此处只需要让学生知道决策树是通过“是”与“否”两个分支来进行决策，不必深究其中的相关知识。

这部分内容相对较难，学生不易理解。教学过程中，可以通过图片、视频资料等进行展示讲解，目的是让学生对决策树有一个初步了解。

1. 收集数据：需要收集一些关于问题的数据，就像是问题游戏中的问题和答案一样。
2. 选择最佳问题：要找到最好的问题来帮助我们进行决策。这就像是在玩游戏时，选择最有助于获胜的问题一样。
3. 创建决策树：根据收集到的数据和选择的问题，创建决策树。这个决策树就像是一个问题的图表，帮助我们完成正确决定。
4. 测试决策树：用一些数据来测试决策树的准确性，验证决策过程的正确性。

通过“购物网站推荐商品”的实例，体会决策树的作用。

首先，购物网站需要收集用户的一些信息，比如年龄、性别、购买历史、浏览记录等。这些信息将用于创建决策树，以预测用户感兴趣的商品。

其次，购物网站要选择一个与用户的特征相关的问题，从而区分不同兴趣群体，如用户的年龄、性别、爱好等。

最后，购物网站根据收集到的信息，创建决策树来帮助推荐商品。假设以年龄作为最佳问题，决策树的第一个节点就是确定年龄段，然后有四个分支，分别对应不同年龄段的商品推荐选项。

通过决策树算法，购物网站可以根据不同年龄、性别等信息，为用户推荐个性化商品，提高用户购物体验的满意度。同时，购物网站还可以不断收集用户反馈的数据，更新决策树，使商品推荐更准确。

## 学习活动2

### 决策树算法的应用场景

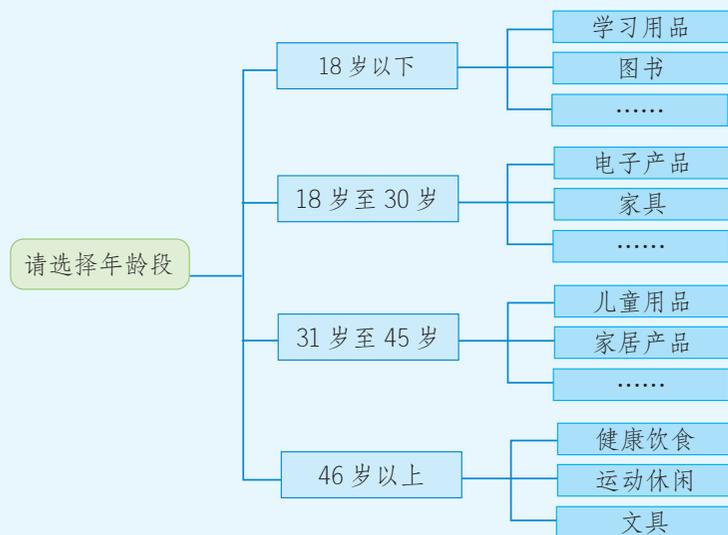
现实生活中，许多场合都使用了人工智能中的决策树算法，除了上面介绍的商品推荐，再举例如下。

#### 医疗诊断

决策树算法在医疗诊断中有广泛应用。医生可以使用决策树，根据患者的症状和检查结果，辅助判断可能的疾病或诊断结果。例如，决策树可以帮助判断一个病人是否患有某种疾病。

#### 自助服务机器人

在火车站、机场、医院等场所，有时你会看到自助服务机器人。只需告诉它你需要什么帮助，它就会告诉你怎么做。因为机器人里的智能决策系统会根据你的问题，自动搜索并筛选相关信息，从而回答问题。



一个按年龄分段的简单决策树

创建决策树是难点。此处不必要求学生能够独立创建决策树。当然，如果学生有学习基础，教师可以带着学生创建部分决策树，再由学生补充，然后交流分享，引导他们说出创建过程中自己的思路 and 想法。

教学时，可以先重点讲解一两个应用场景，再引导学生以小组讨论的方式，寻找生活中其他的应用场景。在启发与帮助学生发现、思考、讨论的过程中学习相关知识。

### ▶ 预测公交车到站时间

有时你会看到公交站牌上有公交车到站的时间。这是因为公交智能系统通过实时获得的信息，考虑路况、其他车辆的影响、乘客高峰出行时间等因素，通过决策树等算法来预测公交车到达各个站点的时间。

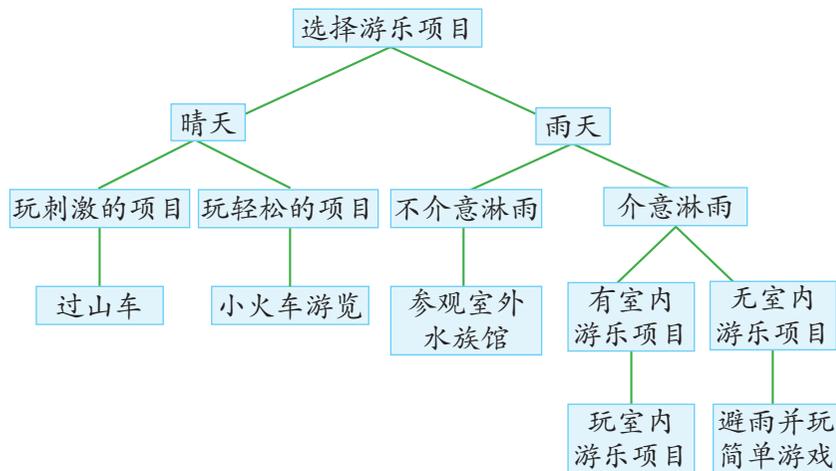
### ▶ 自动驾驶汽车

决策树算法在自动驾驶汽车中具有关键作用。车辆需要根据交通规则和感知到的环境信息做出决策，如是否加速、减速、转弯等。决策树可以帮助自动驾驶汽车做出安全可靠的驾驶决策。

## 拓展与提升

1. 周末，小夏和小智决定一起去游乐园玩，但是不确定应该玩哪个游乐项目，于是他们创建了下面的决策树来帮助选择适合的游乐项目，享受一个愉快的周末。

小夏和小智的做法有什么值得学习的地方？他们进行决策的主要依据是什么？



现在的学生大都见过或使用过智能设备，或多或少都有一些应用体验。人工智能在我们的生活中发挥了越来越重要的作用。人工智能领域的不同应用往往会使用不同的算法。决策树算法是其中常用且重要的一种。

当然，实际应用中的决策树算法都相对复杂，教学时需要适当提炼、简化，便于学生认识与理解。

在这个例子中，小夏和小智首先根据天气情况进行初步判断。如果是晴天，他们会根据自己的喜好选择刺激或者轻松的项目。而如果是雨天，他们会先考虑自己是否介意淋雨，然后根据情况选择雨中的活动或者寻找室内的娱乐项目。

通过这个简单的决策树，小夏和小智可以更加清晰、有条理地做出自己的选择，确保无论天气如何，都能找到适合自己的游乐项目，享受一个愉快的周末。

2. 假期里学生们选择何种体育运动成为老师和家长较关注的问题。如何解决这个问题呢?

建立决策树是一个好方法。决策树解决问题的关键,在于合理设定决策的条件,如年龄、体重指数、天气情况、运动时长、设施投入等。请尝试创建“假期的运动”的决策树。

合理设定影响决策树的条件以及选择的顺序。

年龄: 依据年龄选择运动项目,如跳绳、跑步、游泳、篮球、羽毛球、足球、排球、单杠、骑行等。

体重指数: 依据体重选择适合的项目。

天气情况: 决定室内还是室外运动。

运动时长: 可以分成不同的时间段。

设施投入: 可以分成需要设施的项目和不需要设施的项目。

## 第 29 课

### 智能工具再体验

#### 学习目标

1. 初步了解生成式人工智能，能够用生成式人工智能工具解决问题或进行创作。
2. 通过用生成式人工智能工具描述算法与编写程序，感受人工智能给学习、生活带来的变化。

这一课我们一起来认识生成式人工智能工具，并利用它来生成自己所需的一些辅助学习资料，求解问题。

#### 学习活动 1

##### 用人工智能工具写诗歌

语文老师布置了一项作业，要求仿写课文，描写冬季初雪的景色。小夏同学完成仿写作业后，想尝试让计算机里的智能工具也完成这项作业。智能工具会写成什么样呢？小夏请爸爸帮忙用智能工具进行仿写。于是，爸爸将小夏的问题交给了人工智能工具。

输入要求：请为我写一首冬季初雪的诗

得到回答：好的，以下是一首冬季初雪的诗

##### 冬日初雪

初雪飘落时，世界变得纯净无瑕。  
银装素裹的大地，如同一幅美丽的画卷。  
树枝挂满了雪，仿佛披上了洁白的羽毛大衣。  
寒风吹过，雪花在空中翩翩起舞。

小夏看了很高兴，想要自己试一试，立即修改了问题进行尝试。

输入要求：请为我写一首冬季初雪的七言诗歌

教学重难点：

1. 用生成式人工智能工具解答问题；
2. 用生成式人工智能工具描述算法与编写程序。

这里通过仿写诗歌，让学生感受生成式人工智能工具的应用。教学时，结合学生的学习经历，通过其他学科的问题或者学生学习中的疑问等，引入人工智能技术的应用问题。也可以组织分组讨论，通过适当的问题，引导学生思考并获得答案。

介绍生成式人工智能的含义时，学生有所了解即可，不必过深地涉入。同时，可以强调背后是有很多算法支持的。

生成式人工智能已经在多个领域展示出应用潜力和价值。

文本生成，根据输入的文字要求，生成连贯并具有语法结构的相关内容；图像合成，能够分析图像的特征分布规律，生成与所输入图像类似的图像；视频生成，能够通过学习视频序列的特征，生成新的视频；音乐创作，能够学习音乐的节奏、旋律等，创作音乐作品。

生成式人工智能在自然语言处理、内容创作、聊天机器人、图像艺术创作、虚拟现实、特效生成、音乐创作等领域，都有着广泛的应用前景，而且相关技术还在不断被突破。

得到的回答：好的，以下是一首冬季初雪的七言诗歌

### 冬日初雪

初雪飘零落满山，  
银装素裹似仙境。  
寒风凛冽吹不散，  
冰清玉洁入梦来。

小夏感叹人工智能强大的功能，爸爸告诉他，这是生成式人工智能技术的实际应用。通过跟计算机中的智能系统进行对话，输入自己的问题要求，然后让智能系统帮助生成相关内容。

## 学习活动2

### 用人工智能工具解答问题

打开一种生成式人工智能工具，分别输入下面的问题，尝试让智能工具辅助自己求解问题。

问题1：我想知道交通出行中的算法应用，适合小学生阅读。

问题2：编写一个程序，输入两个数，计算这两个数的和并输出。

```
1 # 获取用户输入
2 num1 = float(input("请输入第一个数: "))
3 num2 = float(input("请输入第二个数: "))
4
5 # 计算和
6 sum_result = num1 + num2
7
8 # 输出结果
9 print("两个数的和是: ", sum_result)
```

你可以将这段代码复制粘贴到一个Python环境中运行，比如Python的交互式环境或者一个.py文件。当你运行这个程序时，它会提示你输入两个数，然后输出这两个数的和。

智能工具给出的一个程序

多数生成式人工智能工具都支持生成图像，教学时也可以让学生体验图像的生成，发挥自己的个性，创作图像作品。

在学生初步感受人工智能工具后，继续用它来解决问题。这里设计几道难度不同的问题供学生学习时选用。

先通过问题1，让学生熟悉利用人工智能工具求解问题的过程，阅读获得的内容。

问题2的难度有所增加，获得的内容包括相应的程序及其解释，可巩固、拓展学生前面学习的内容。

问题3：有两个沙漏，大沙漏能计时7分钟，小沙漏能计时4分钟，怎样才能准确地计时9分钟？

经过实践操作发现，智能工具多数时候能够给出正确答案，但有时并不能够完全解决问题。例如，输入上面问题3后，有时并不能给出准确的做法。

生成式人工智能系统通过研究大规模的历史数据来创造出新的、原创的内容，包括文本、图像、声音、音频、视频和程序代码等，但它并不是无所不能的，也不能完全代替人类的独立思考。

因此，作为小学生，需要认真学习，学会独立思考。平时可以通过这些技术工具辅助自己学习，拓宽知识面，探索未知的内容，但不能单纯地、过度地依赖这些工具。

### 学习活动3

#### 用人工智能工具实现算法

任务要求：体重指数又称BMI (body mass index) 指数，通常指用体重除以身高的平方得出的数值，是目前常用来衡量人体健康程度的标准之一。

1. 根据输入的数值，计算并输出BMI指数。
2. 描述计算BMI指数的算法并编写程序实现与验证。

主要的实践过程如下。

第1步：打开生成式人工智能工具，输入“描述计算BMI指数的算法”。

第2步：阅读生成的内容，判断其正确性。

第3步：输入“编写一个程序，计算BMI指数并输出”。

第4步：把生成的程序代码复制到Python环境中，调试运行，输入几位同学的体重和身高，查看计算结果，验证算法。

问题3的设计目的就是让学生通过实践发现：人工智能工具给出的答案不一定能解决问题，甚至无法找到正确答案。从而引导学生思考，在信息社会中，人工智能、大数据等新技术、新应用在学习过程中的作用。即使它们的功能很强大，也不能代替我们的独立思考，要学会利用技术工具发挥积极作用，但决不能过度依赖技术工具，更不能成为学习的阻碍。

必要时可以引导学生思考：你能写出计时的步骤吗？

1. 将两个沙漏都倒过来，等小沙漏漏完，时间用了4分钟。
2. 不动大沙漏，把小沙漏再倒过来重新计时，等大沙漏漏完，时间一共用了7分钟。
3. 不动小沙漏，把大沙漏再倒过来重新计时，等小沙漏漏完，时间一共用了8分钟。
4. 这时立即把大沙漏再倒过来，等待漏完，用时刚好是9分钟。

进一步学习：提出一个实际问题——计算BMI指数。

教师可以带着学生运行程序，进行算法验证。输入不同的数据比较分析，在此基础上还可以引导学生完善程序。

先通过人工智能工具生成算法，获得的内容通常是按步骤的文字描述，过程清晰，并把不同指数对应的健康状况也进行了说明。

接下来生成程序。除了实现计算功能外，应用分支结构判断不同的健康状况。

需要说明的是，把生成的程序复制到Python环境中后，有时需要适当调试处理，如文字格式、判断错误、符号等。

## 拓展与提升

1. 智能体在人工智能领域是一个很重要的概念，通常指能够自主活动、感知环境、做出决策并与环境交互的计算机系统或实体。目前，已经出现一些智能体平台，为人们提供智能体的相关服务。

依据自己的兴趣爱好，尝试创建自己的智能体。

2. 阅读下面的内容，思考“算法自主可控的重要性”。

算法的自主可控主要指在算法的设计、开发、应用过程中，能够实现自主权利和自主可控性，保护算法的知识产权，提高自研算法产品的推广和使用。算法自主可控的重要性体现在多个方面。

算法是信息科技领域的核心组成部分，拥有自主可控的算法意味着国家能够在算法设计和优化方面拥有独立自主的能力，不必过度依赖外部技术。这有助于提升国家的技术创新能力，在全球竞争中占据有利地位。

算法自主可控对于国家安全至关重要，如果算法依赖外部供应商，那么国家可能面临外部势力通过算法进行监控、渗透和攻击的风险。自主可控的算法能够确保国家关键信息基础设施的安全和稳定，防范外部威胁。

自主可控的算法能够更好地适应我国社会经济、文化的发展，促进公共利益最大化。

自主可控的算法能够更好地保障个人数据的处理和使用符合我国法律法规和道德标准，减少数据泄露和滥用的风险。这对于维护个人隐私权益、构建和谐社会具有重要意义。

因此，应该加强算法自主可控技术的研究和应用，提升国家算法技术的自主创新能力，确保算法的安全性和可靠性。

1. 智能体以云为基础，以人工智能为核心，是在一定的环境中体现出自治性、反应性、社会性、预动性、思辨性（慎思性）、认知性等一种或多种智能特征的软件或硬件实体。智能体能够根据自身目标和环境状态，自主地制订计划并执行行动，而不需要人类直接干预。能够与其他智能体或人类进行交互，通过通信和协作完成任务，还能实时感知并适应不同的环境变化。

2. 可以安排学生课下阅读延伸，也可以通过组织讨论交流的方式，引导学生学习。让学生知道算法具有知识产权，谁拥有越多具有知识产权的自主可控算法，谁在未来的竞争中就越有优势。自主可控与国家、社会、企业和个人密切相关。

## 第 30 课

### 生命游戏的算法

#### 学习目标

1. 了解生命游戏的演化过程，通过图示的方法，认识生命游戏中的算法规则。
2. 通过简单的游戏体验，初步领会简单规则也可能产生较复杂的模式。

这一课通过认识、体验生命游戏的规则，学习有关的算法知识。

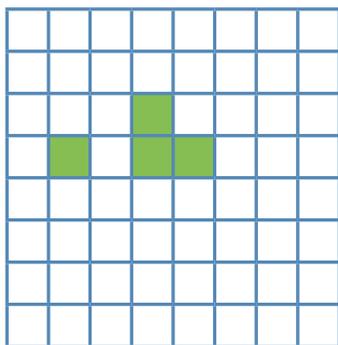
#### 学习活动 1

##### 生命游戏中的算法规则

最初的生命游戏模拟的是一种原始形态的“生命”如何按照特定的规则一代接一代地进行演化，并由此得名。

在生命游戏中，有很多小方块，每个小方块都有两种状态。我们可以把这些小方块看作是灯泡，两种状态分别对应着灯泡的“点亮”或“熄灭”。

游戏开始时，在一个平面网格中，可以设置若干个点亮的灯泡，表示原始状态的“初始构型”，下图所示就是一个初始构型的例子。接着设定表示进化的“代”数，这两个部分构成算法的输入。



一个初始构型

教学重难点：

1. 生命游戏中的算法规则；
2. 简单规则也可能产生较复杂的模式。

生命游戏是模拟生命演化过程的游戏，游戏中的每个生物都有一定的寿命和繁殖力，它们会在一定的空间中生存和繁衍。

生命游戏由数学家约翰·何顿·康威（John Horton Conway）于 20 世纪 60 年代末设计的一个计算机程序。1970 年趣味数学大师马丁·加德纳（Martin Gardner）在一份流行很广的科普杂志上将生命游戏介绍给读者，吸引了很多人关注和研究。

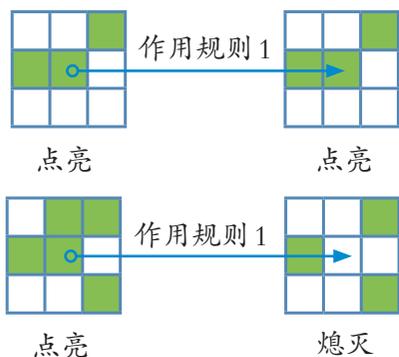
生命游戏对很多领域，特别是对计算机算法、计算机图形学、人工智能和计算机游戏设计等都产生了深远的影响，由生命游戏所衍生出来的各种问题至今仍然备受研究者的关注。

生命游戏是一种简单的生成式算法，即算法通过一定的规则作用到输入，从而生成新的事物。

生命游戏的算法规则（即进化规则）可以归纳为两条。

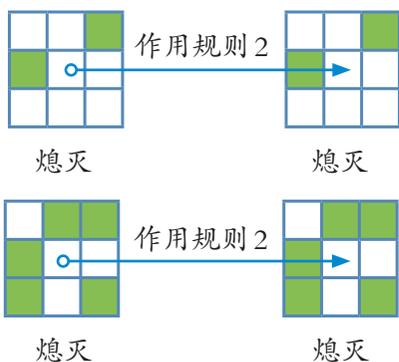
规则1：一个点亮的灯泡周围有2个或3个点亮的灯泡，就会继续点亮；它周围有少于2个或多于3个点亮的灯泡，就会熄灭。

例如，下图中心位置的灯泡，用规则1来检查它周围的状态，也就是作用规则1，发现它仍然可以保持点亮状态。



规则2：一个熄灭的灯泡周围刚好有3个点亮的灯泡，才会被点亮。否则，它周围有少于3个或多于3个点亮的灯泡，就会保持熄灭状态。

例如，下图中心位置的灯泡，作用规则2后保持熄灭状态。



作用规则到指定的代数后进化就会停止，停止之后的网格构型就可以认为是算法的输出。

生命游戏模拟了生物的生存和繁衍过程。在这个游戏中，有一个由许多方格组成的大网格，方格上会有一些生物。每个生物都有一定的寿命和繁殖力。而当一个生物死亡时，就会在周围的方格上留下一些生命种子，这些种子可能会在未来的进化中成长为新的生物。生物的生长需要食物，如果一个方格上没有食物，那么这个方格上的生物就会死亡，而食物的来源可以是周围生物死亡后留下的生命种子。

跨学科思考：生命游戏中的规则实际上是在模拟原始生命（如细菌）在进化中的一些规律，因此，每条规则大体对应着现实中某种真正的进化规则（当然是简化及理想化的）。

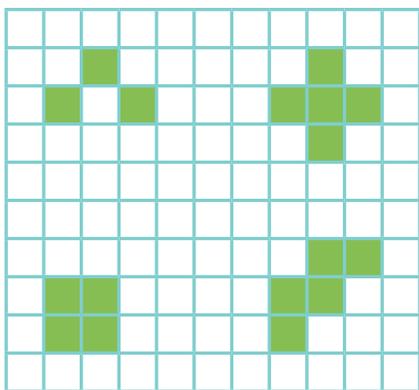
例如，规则1可以解释为：如果一个细菌的周边有2个或者3个其他细菌，在比较理想的环境中，这个细菌可以继续保持活力。而其周边的细菌过少（0个或1个）或过多（4个以上）都不适合生存。

教师可以指导学生，按照这个思路进一步解释其他几条规则。

## 学习活动2

### 体验生命游戏进化规则

以下图中的4个初始构型为例，进一步学习了解如何作用这些规则。



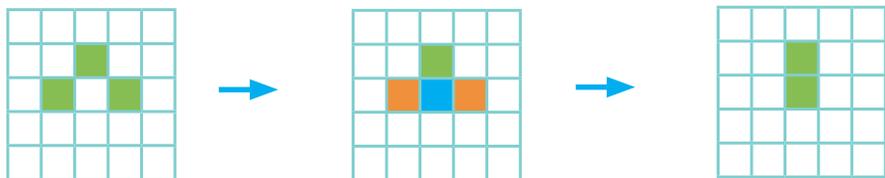
4个初始构型

图中的绿色方格代表点亮状态，空白代表熄灭状态。

第1步：把图中的4个构型作用于进化规则，完成第一代进化。

例如，左上角的构型有三个点亮的灯泡，作用规则1后，一个继续点亮、两个变成熄灭状态。同时作用规则2后中间一个被点亮。

图中的橙色和蓝色用来临时表示进化的过程，橙色方格表示该位置即将由点亮变为熄灭，蓝色表示该位置即将由熄灭变成点亮。



左上角的构型作用规则

教师可以带领学生一起来完成这个“进化”的体验过程。

对生命游戏的解释还有很多适用的情境或场合。

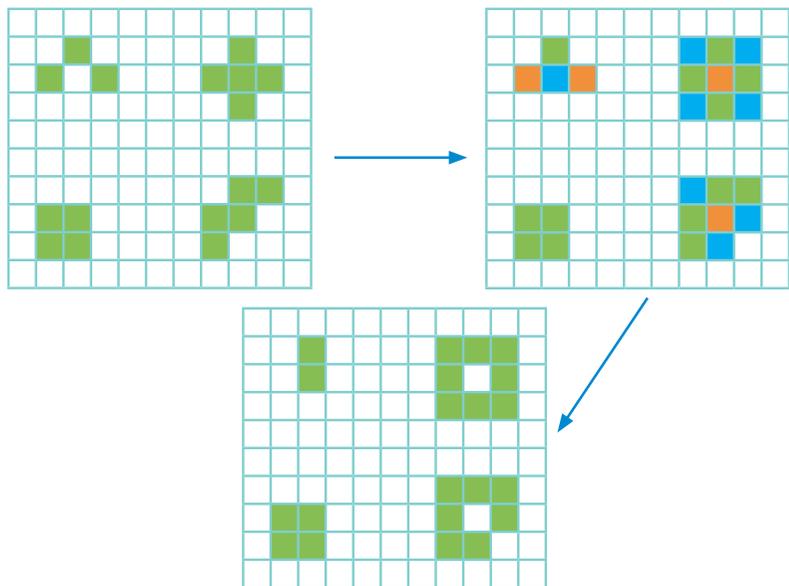
例如，可以将一个网格视为“熊猫乐园”，其中的方格中生活着一些熊猫。当一个方格点亮时表示生活在这个方格中的熊猫处于活跃状态，熄灭则表示处于安静状态。

这样，生命游戏的进化规则就会有另一种风格的解释。

例如，规则1可以解释为：如果一只处于活跃状态的熊猫周边有2只或者3只熊猫同样也活跃，它就会继续保持活跃状态。但如果它的周围活跃的熊猫太多（3个以上）或太少（少于3个），它就会变得安静。

教学时，教师可以基于各种不同的解释场景，设计多种能让学生实践参与的活动，从而在不用编程的前提下，加深对算法及算法中关键思想的认识。

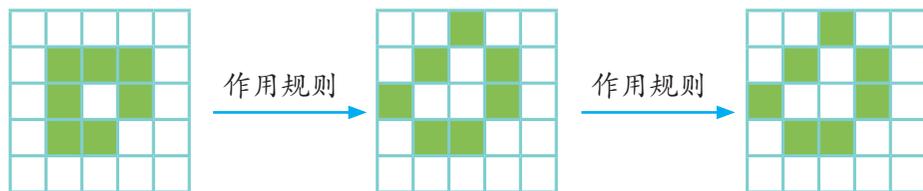
下图是对4个初始构型分别作用一轮进化规则后的状态。



4个初始构型分别作用一轮进化规则

第2步：把第一代进化的结果作为初始构型，继续作用进化规则，就会得到第二代、第三代，也会产生一些更有趣的现象。

例如，对第一代进化后右下角的构型继续作用规则，得到第二代、第三代的进化结果，如下图所示。



右下角的构型继续作用规则

可以发现，从第三代开始，这个构型就会保持不变，稳定下来。

第3步：继续用其他几个构型尝试，观察发生的变化。

五个构型都完成第一代进化后，会发现一些有趣的现象。

一是有三个构型发生了改变；

二是左下角的那个构型保持不变，这就意味着后续无论作用多少次规则、进化多少代，这个构型都将不会改变。

注意：要对网格中的每个方格都要作用一次规则。

生命游戏的算法看似简单，却能产生十分复杂的图案。教学时，要引导学生仔细观察、独立思考，发现规律，体验作用规则后构型是否产生变化，以及会变化成什么样的构型。

可以进一步启发学生联想：这个游戏模拟了自然界的生存法则——适者生存。

在游戏中，可以“看到”各种各样的生物群落是如何形成的，一些强大的生物是如何在竞争中逐渐占据优势的。同时，也会“看到”一些弱小生命在竞争中逐渐消失。

### 学习活动3

#### 试玩“生命游戏”程序

第1步：打开配套资源中名为“生命游戏”的Python程序。

第2步：运行这个程序，试着玩一玩。

第3步：观察、体会游戏中的算法及其进化规则。

### 拓展与提升

1. 利用更大的网格设置更多的构型，继续尝试探究生命游戏的规则，加深对算法进化规则的理解。

2. 使用生成式人工智能工具，输入下面的语句，阅读获得的内容，进一步了解生命游戏的规则。

为我解释生命游戏，要有简单的说明性实例，适合青少年阅读。

学习活动3可以依据教学实际选择学习或作为进一步的延展。

这个Python程序，给出了一些可以产生有趣输出的初始构型。

教学时，还可以带领学生进行一些简单的实验探究，让他们熟悉进化规则的作用过程，进一步领会“简单规则也可能产生看起来非常复杂的模式”，甚至自己发现一些更有趣的图案。

1. 教师可以准备更多的、具有不同难度的构型图案，供不同层次的学生拓展学习。

2. 引导学生利用前一课学习的生成式智能工具相关知识，进一步学习本课的内容。

此外，也可以建议学生回家后向家长介绍生命游戏，并与家长一起玩一玩这个游戏。