



普通高中教科书

XINXI
JISHU

信息技术

必修 1

数据与计算



XINXI
JISHU

信息技术

必修 1

数据与计算



教育科学出版社

前 言

子曰：“君子不器。”

事物是发展的，因此我们不能故步自封，要不断创新创造、追求完美，要以发展的眼光看待一切。物质世界所携带的信息可以用数字来描述，现实社会生活也可以用数字来记录……，可以说，恰是海量数据联系着现实世界和虚拟世界，将两者连为一体。数字之所以能够快速、高效地生成、加工与表达，是因为其背后有一种神奇的力量——计算机程序。而掌控这力量的，是由人设计的算法。因此，无论是用应用软件加工数据、用数字化技术处理信息，还是用程序代码编写软件，甚至是开发、运用人工智能，善于运用计算思维来解决问题，都是我们应当具备的基本能力。

本书根据《普通高中信息技术课程标准（2017年版）》编写，由初识数据与计算、编程计算、认识数据、计算与问题解决、数据分析与人工智能五个单元组成。本书内容主要以项目学习的方式组织，在给出明确的“学习目标”的基础上，设计生动、有趣的“任务”引领“活动”，设置必要的“拓展练习”供同学们自我检测，提供“拓展知识”帮助同学们开阔眼界，并以具体的测评要求和思维导图引导同学们进行单元学习评价与总结。

本书旨在引导同学们通过解决日常生活中遇到的各种问题，了解数据、信息和知识及其相互关系，了解数据在数字世界中的编码与结构，了解如何采集、存储、组织、处理、分析与可视化表达数据，认识数据对日常生活的影响，并形成数据的安全意识；知道通过计算解决问题的过程，了解算法的概念及其描述与基本特征，掌握算法的基本结构，能够使用程序设计语言编写代码实现简单算法并发布、分享程序代码，进而学会分析问题并根据问题选择合理的算法；了解人工智能技术的相关概念、发展趋势以及在信息社会中的重要作用。同学们通过参加丰富多彩的学习活动，可以形成良好的信息意识，初步形成计算思维，学会进行数字化学习与创新，明确信息社会责任。

希望同学们立足于本书，又不局限于本书，做到理论联系实际，在实践中练就技术本领，树立创新意识。期待大家积极分享学习中的各种收获！

目 录

第 1 单元	初识数据与计算	1
1.1	我们身边的数据	2
1.2	数据的计算	10
	单元学习评价	16
	单元学习总结	16
第 2 单元	编程计算	17
2.1	计算机解决问题的过程	18
2.2	做出判断的分支	26
2.3	周而复始的循环	34
2.4	可以复用的代码	40
	单元学习评价	47
	单元学习总结	47

第 3 单元	认识数据	48
	3.1 数据编码	49
	3.2 数据与结构	56
	3.3 数据与系统	66
	3.4 加密与解密	77
	单元学习评价	83
	单元学习总结	84
第 4 单元	计算与问题解决	85
	4.1 算法及其特征	86
	4.2 数值计算	93
	4.3 非数值计算	100
	4.4 综合问题的解决	107
	单元学习评价	115
	单元学习总结	116
第 5 单元	数据分析与人工智能	117
	5.1 走近数据分析	118
	5.2 探秘人工智能	126
	单元学习评价	135
	单元学习总结	136
后	记	137

第 1 单元 初识数据与计算

人类在原始社会就出现了“结绳记事”“刻契记事”等简单的计算行为。在信息化时代的今天，数据与计算对我们的生活、学习和工作的影响越来越大，已经和我们密不可分。

对现实世界的事物进行感知和测量，进而获得大量的数据，再依靠计算发现数据背后蕴含的规律是人们生产、研究的重要手段。天文学家通过计算分析太空脉冲，生物学家通过计算发现基因组的奥秘，交通部门通过计算实现公共车辆的合理调度，工厂通过计算确定生产材料的最佳配置方案……。人类的活动，时时刻刻都离不开数据和计算。

那么，数据是什么？计算是什么？这是本单元要讨论的问题，我们也将从这里开始“数据与计算”之旅。

本单元我们将从认识身边的数据开始，了解数据和信息的特征，理解数据、信息和知识的关系，感受数据在现代社会生活中的重要作用。通过比较多种计算方法在解决问题时的特点，体会计算机在处理数据上的优势和价值。



如果你“拷问”数据到一定程度，它自然会坦白的。

—— 罗纳德·哈里·科斯 (Ronald H. Coase)

1.1 我们身边的数据

当今时代被称为数字时代或信息时代，数据、信息和知识与人类的活动密切相关。加深对这些概念的认识，是本课程初期的重要任务。我们将通过实例，了解数据在现代生活中的重要性，了解数据和信息的特征，理解数据、信息和知识的相互关系，并在实际问题情境中，尝试选择恰当的方法收集数据，进行简单的数据分析。



学习目标

- ★ 了解数据与信息的特征。
- ★ 理解数据、信息和知识的相互关系。
- ★ 能从恰当的来源获取数据，并对数据进行简单的分析。
- ★ 体验数字化学习过程，感受利用数字化工具和资源的优势。

气象生活指数是气象部门根据对气象预测数据的分析得出的居民生活出行参考数据。气象生活指数所含指标众多，常见的如紫外线指数、雨伞指数、感冒指数、穿衣指数，也有很多新奇的指数，如放风筝指数、啤酒指数等。这些指数不仅可以给我们的日常生活提供帮助，也可以为许多行业决策提供参考。本节我们将围绕“气象生活指数的参考价值研究”项目展开学习。同学们将通过该项目活动，了解数据、信息和知识的相互关系，并知道如何通过收集和分析数据来佐证研究结论。

本项目主要包含“获取当地的气象生活指数”和“分析气象生活指数的参考价值”两个任务。

我们常会遇到“今天我该穿什么”“出门是不是需要带雨伞”“今天空气污染是否严重，能否长时间进行户外运动”等问题。很多应用

软件会给出气象生活指数建议，如“今天天气较舒适，建议穿薄外套或牛仔裤等服装”，这些贴心的建议是如何给出的呢？



任务一 获取当地的气象生活指数

※ 活动1 获取当地的气象生活指数及建议

气象部门对当地每天的气象因素如温度、湿度、风力、气压等数据进行再加工，结合各气象因素对人们生活的影响程度，把与气象生活指数相关的气象因素代入设定好的公式中进行计算，再将结果细分成不同等级，形成生活指数建议。请将今日当地的气象生活指数等级及生活建议填入表1.1.1。

表1.1.1 今日气象生活指数

	紫外线指数	感冒指数	穿衣指数	运动指数	空气污染扩散指数
指数等级					
生活建议					

※ 活动2 计算今日感冒指数

不同地域的气候条件差异较大，气象生活指数种类各不相同，气象生活指数的计算方法也不统一。气象部门公布的感冒指数是由当地24小时降温幅度、气温日较差、相对湿度、气压对感冒指数的贡献值（表1.1.2）相加而得。感冒指数与等级的对应关系如表1.1.3所示。

表1.1.2 各种相关气象因素对感冒指数的贡献值

24小时降温幅度		气温日较差		相对湿度		气压	
数值/℃	贡献值	数值/℃	贡献值	数值/%	贡献值	数值/hPa	贡献值
< 4	0	≤ 7.9	0	>50	0	<1030	0
4~7	5	8~9.9	5	30~50	3	≥1030	10
8~10	20	10~12.9	10	< 30	6		
> 10	30	≥13	15				

24小时降温幅度一般指前一天最低气温和当天的最低气温的差。气温日较差是一天中气温最高值与最低值的差。

表1.1.3 感冒指数等级

等级	指数范围	含义
1级	≤ 6	少发
2级	7~19	较易发
3级	20~30	易发
4级	≥ 31	极易发

请上网查找当地的天气数据，依据上述方法求出今日的感冒指数等级，并将结果写在横线上。

昨日最低气温 _____ 今日最高气温 _____

今日最低气温 _____ 今日相对湿度 _____

今日气压 _____ 感冒指数等级 _____

将求得的感冒指数等级与表1.1.1的相应结果比较。如果不同，原因是什么？

在完成任务的过程中，我们实际上经历了获取数据、提取信息和利用知识的过程。那么什么是数据、信息和知识呢？它们在任务中是如何呈现的？彼此又存在着怎样的联系呢？

● 数据

在我们身边存在着各种数据，如刷公交卡时刷卡机显示的本次扣款金额和卡内余额、电子手表上显示的日期和时间、教科书封底下端的国际标准书号（ISBN）、体育课上测量的各项运动成绩等。这些数字都是数据，我们在活动2中填写的数字也是数据。

我们的名字、诗人创作的诗句、奥运会的主题曲、记载学校运动会的视频等都不是数字，那么它们是数据吗？实际上，数据的表现形式多种多样，除了数字以外，还有文字、图形、图像、声音和视频等形式。

在日常生活中，有些数据是固定不变的。例如，不管圆有多大，它的周长与直径的比值总是一个固定的数，也就是我们常说的圆周率 π ；再如，一个标准大气压下，冰水混合物的温度为 0°C 。

有些数据是不断变化的，例如家庭每天的用电量、国际黄金价格、气温等。这些数据随着时间而改变，因此在运用此类数据时要注意时效性，否则有可能造成计算无效。

有些数据是随机出现的。比如抛硬币，是正面向上还是反面向上是随机的。但若抛几百次、几千次，正面向上的可能性就稳定在50%左



右。这时，数据的出现往往又有一定的概率。

● 信息

单纯的数据不能表达具体的含义，例如我们并不清楚像28和90%这样的数值在生活中的具体意义。在对数据进行处理或赋予其具体的情境后，这些数据就成了有意义的信息。28和90%可以分别用于表达今日的最高气温28℃，相对湿度90%，此时数据就成了有意义的信息。

数据是信息的符号表示；信息是数据的内涵，是对数据的语义解释。数据处理就是把原始数据转换成人们所需要的信息的过程。完整的数据处理过程一般包括数据的采集、加工、传输、存储、检索和输出六个环节。数据处理的目的是把数据转化成有用的信息以满足用户的需求。

一般而言，不管是什么样的信息，它们通常都具有一些特征，如载体依附性、价值性、时效性、共享性等。

以天气预报信息为例，该信息不能独立存在，需要依附于一定的载体，它可以显示在网页上，也可以显示在报纸上，体现了信息的载体依附性；人们可以利用天气预报作为穿衣或出行的参考，体现了信息的价值性；天气预报只是预报了未来某一特定时段的天气状况，它会随着时间的推移而变化，体现了信息的时效性；作为一种资源，天气预报往往可以被多个信息接收者接收并且多次使用，这就体现了信息的共享性。

● 知识

知识是信息经过加工提炼后形成的抽象产物。它表述的是事物运动的状态和状态变化的规律。可以说，知识是一类高级的、抽象的，而且具有普遍适应性的信息。

知识是人类求知活动的结晶。例如，人们发现感冒的发生除了与病毒感染有关外，天气条件也是一个重要因素。冷空气的入侵，造成大幅度降温，或者冷空气经过后出现冷高压天气，特别是大气压高于1030百帕的晴好天气，就会有大批人感冒。感冒指数的计算方法，是经专家调查研究和演算归纳出来的。知识是认识世界的结果，同时也是改造世界的依据。

数据、信息和知识可以看作人类对客观事物感知的三个不同阶段。数据是对事物属性的客观记录，信息是经过组织的有结构的数据，知识是经过人的思维整理过的信息、数据、形象、价值标准以及社会的其他符号化产物。

由于身体条件不同，不同的人对温度、相对湿度等的感受也不同。有的人说气象生活指数很准确，有的人却认为这些指数参考价值不大。下面我们对此开展研究。



任务二 分析气象生活指数的参考价值

※ 活动1 调查本班同学的感冒和穿衣情况

请从感冒指数和穿衣指数等话题入手收集研究数据，探讨气象生活指数的参考价值。

设计一份调查问卷，了解今天本班全体同学的感冒和穿衣情况。要求调查问题有针对性，问卷格式符合一般设计规范。可以通过网络调查平台发布问卷并实施调查。

● 数据的来源

按照获取数据的渠道，可以将数据分为两类：直接数据和间接数据。统计调查或科学实验通常是数据的直接来源，由此得到的原始数据一般称为直接数据或一手数据；而通过查阅资料获得的数据或他人对原始数据处理过的数据，一般称为间接数据或二手数据。例如，我们用温度计实际测量得到的气温值是直接数据，而从网上搜索到的气温值是间接数据。

● 数据的收集方法

直接数据的收集方法有很多，如调查法、访谈法、观察法、实验法等。随着技术的发展，收集数据的手段日益丰富。例如，在农田中使用传感器可以实时收集农田土壤含水率，用以指导精确灌溉；利用车载尾气检测技术可以获取实时的动态尾气数据，用以协助调整交通策略及城市规划；利用中国嫦娥四号探测器收集月球表面成分的第一手资料。

间接数据一般都已经过加工处理，使用起来更加方便。收集间接数据的方法也有很多，如查阅文献资料、关注媒体资源、浏览权威网站、利用搜索引擎在网络中进行检索、从相关数据公司购买等。在收集这类数据时要综合考虑数据的时效性、可信度以及经济条件等方面的因素。

抽样调查时必须保证抽取样本的特征分布与调查对象总体的特征分布相一致。

问卷调查法是我们常用的数据收集方法之一。根据调查对象范围的不同，问卷调查法又分为全面调查和抽样调查。本活动要求设计一份调查问卷，以获取本班同学感冒和穿衣情况的直接数据。由于调查对象相对集中，而且人数有限，可以采用全面调查。调查问卷一般包

含标题、卷首语、调查内容和结束语。以下调查问卷可供参考。

您好！本次调查想了解您今天的身体状况和穿衣情况，目的是研究气象生活指数的参考价值。调查数据仅用于研究，调查结果不记名，不存在个人隐私泄露问题。谢谢合作！

气象生活指数的参考价值研究小组

请仔细阅读问题，在适当的选项上画“√”。所有问题均为单选题。

Q1. 您的性别:

1. 男

2. 女

Q2. 您近1-2天新发感冒了吗?

1. 是

2. 否

Q3. 您今天的穿衣情况: (如无匹配项, 请选择近似项)

1. 厚羽絨服

2. 棉衣加羊毛衫

3. 厚外套加毛衣

4. 薄外套或牛仔褲

5. 长袖衬衫、单裤

6. T恤、超薄外套

7. 短衫、短裤

衷心感谢您的配合。如您对本次调查还有其他意见和建议，请写在下面的横线上。

卷首语：一般需要明确调查的目的、意义和内容，以及回答问卷的方式，消除答卷者的后顾之忧。卷首语要简明扼要、谦虚诚恳。

调查内容通常包含：被调查者的基本状况，可作为对被调查者进行分类比较的依据；调查的具体项目，它是问卷的核心部分。

可在结束访谈部分对被调查者的合作表示真诚的感谢。

随着技术的发展，人们开始运用网络调查的方法收集数据。与传统的纸笔调查相比，网络调查可以节省成本，扩大调查范围，也便于后期的数据处理。可以使用专业的网络调查网站创建、发布调查问卷，如图1.1.1所示。

标题 问题 设置标题字体 输入图片说明 当前题型 单选题 转换题型

您近1~2天新发感冒了吗?

☒ 必答题
☐ 填写提示
☐ 无条件跳转
☐ 关联逻辑

选项文字	图片	说明	允许填空	跳转	默认	分组	操作
是			☐		☐	分组	
否			☐		☐	分组	

添加选项 批量增加

设置分组 设置排列

完成编辑

图1.1.1 使用网络调查平台创建问卷

图1.1.1 使用网络调查平台创建问卷

※ 活动2 分析气象生活指数的参考价值

依据调查结果计算新增感冒人数和各项穿衣情况占样本总数的百分比，将所得结果与当地同日的气象生活指数进行比较，你能得出哪些结论？

● 数据分析

交叉分析法通常指将两个有一定联系的变量及其值交叉排列在一张表格内，使各变量值成为不同变量的交叉节点，形成交叉表，从而分析交叉表中的变量之间的关系。

数据分析是指用适当的统计分析方法对收集来的大量数据进行分析，提取有用信息，并形成结论的过程。

针对调查问卷收集的数据，我们可以分析问卷中不同指标的分布、所占比例等情况，也可以进一步对问卷指标进行交叉分析，如性别不同，穿衣情况比例是不是也有所差异等，然后在分析的基础上得出结论。很多网络调查平台为用户提供了一些分析功能。

通过网络调查平台的统计分析功能，可以得到如图1.1.2所示的结果，比较当日的感冒和穿衣指数，你就可以得出相关的结论。



图1.1.2 网络调查平台的统计分析功能

本次调查因为样本量太小，且只有一天的数据，不能准确地反映感冒和穿衣指数的参考价值。为了使调查结果更有参考价值，应该怎么做呢？请写出你的方案。



拓展练习

在本节的项目活动中，我们借助了一些数字化工具来实现活动的要求，如搜索引擎和网上问卷发布工具。诸如此类的工具还有很多，如概念图和思维导图、RSS阅读器等。学习者借助数字化工具可以随时随地学习乃至进行终身学习。请设计一份调查问卷，调查同学们是如何利用数字化技术进行学习的。



拓展知识

关于数据、信息和知识的定义，不同的领域和组织有不同的认识。

数据是对象的表示。即事实、概念或指令按照适合于通信、解释或处理（借助人或自动装置）的方式所形成的表示。（摘自：《中国大百科全书》第二版）在计算机科学中，数据的定义是指所有能输入计算机中并被计算机程序处理的符号的介质的总称，是用于输入计算机进行处理，具有一定意义的数字、字母、符号和模拟量等的统称。

信息是事物的运动状态及其状态变化的方式。（摘自：钟义信，《信息科学原理》）凡是在一种情况下能减少不确定性的任何事物都叫作信息。（摘自：信息论的创始人香农和韦弗，《通信的数学理论》）物质存在的一种方式、形态或运动状态，也是事物的一种普遍属性，一般指数据、消息中所包含的意义，可以使消息中所描述事物的不确定性减少。（摘自：中华人民共和国国家标准GB 4894—85《情报与文献工作词汇基本术语》）

某一类事物的知识，就是主体关于该类事物运动状态及其变化规律的表述。（摘自：钟义信，《信息科学原理》）知识是人类认识的成果，是在实践的基础上产生，又经过实践检验的对客观实际的反映。（摘自：《中国大百科全书》第二版）

1.2 数据的计算

在人类发展的历史中，在较长的一段时间里，计算多依靠人工方式来完成。直至今日，我们也经常会“人工计算”。随着计算机技术的发展，以数据抽象和自动化处理为代表的计算方式在改变着人们的行为方式的同时，也在改变着人们的心理认知历程。本节我们将通过实例感受各种计算方式，并体会面对实际问题时应如何选择恰当的计算方式。



学习目标

- ★ 了解计算的基本方式和计算的发展过程。
- ★ 根据问题需求，选用恰当的计算方式。
- ★ 感受计算机在处理数据时的优势。

我国古代出现了多部数学著作，如《九章算术》《孙子算经》等，里面记录了很多经典的数学问题。本节以其中的“鸡兔同笼”问题为例，以“‘鸡兔同笼’问题的解决”项目为线索展开学习。同学们可以通过该项目，了解基本的计算方式及特点，领会如何根据需要选用恰当的计算方式。

本项目主要包含“探讨解决问题的计算方式”和“解决‘鸡兔同笼’问题的计算方式比较”两个任务。



任务一 探讨解决问题的计算方式

※ 活动1 人工方式解决“鸡兔同笼”问题

《孙子算经》记载：“今有雉（鸡）兔同笼，上有三十五头，下有九十四足，问雉兔各几何？”请思考该问题的计算方法。

● 计算

“数据”在“运算符”的操作下，按“规则”进行的数据变换，如“ $3 + 3 = 6$ ”“ $3 \times 2 = 6$ ”，这是简单计算中的算术运算。

通过建立分析模型和设计有效的步骤、方法，利用自动计算工具来进行规律预测和发现，这是计算机领域的计算。

人类分析问题常采用的方法是计算，因此计算可以看作是一种获得信息的过程。

请同学们上网搜索《孙子算经》中“鸡兔同笼”问题的计算方法，并与你所采用的方法进行对比，说明异同。

※ 活动2 借助电子表格软件解决“鸡兔同笼”问题

除了《孙子算经》记载的解决“鸡兔同笼”问题所使用的假设法外，还可以使用方程法、抬腿法、列表法等。列表法比较容易理解，它需要借助表格，把35头鸡和兔的脚的组合都列出来。

请用电子表格软件设计一张解决问题的表格。表格结构如表1.2.1所示，通过计算得出各单元格的值，并用红色标识出求得的鸡兔数量。

表1.2.1 “鸡兔同笼”问题表格

“鸡兔同笼”问题（35头94脚）				
兔的只数	鸡的只数	兔脚总数	鸡脚总数	共有的脚数
1				
2				
⋮				
33				
34				

● 电子表格软件

电子表格软件是面向大众的计算工具，它的图形用户界面及“所见即所得”的编辑技术等使对数据进行计算更加易于操作，从而更充分地实现人和计算机的优势互补，提高人机协作效率。下面我们以前“班级运动队的意向调查表”为例，了解电子表格软件中的一些常见操作。

● 数据的排序

WPS表格是一种常见的电子表格软件，利用排序功能可以依据关键字对表中的数据进行升序或降序排列，从而更清晰地了解各种分类信息。

例如，我们希望找到身高最高的几位同学，可以依据身高进行排序。如果需要按照多个条件排序，可以进行多个关键字的设定。如图 1.2.1 所示，通过设置主要关键字、次要关键字，能很容易地找出女生中身高最高的几位同学。

2	编号	姓名	班级	性别	身高	体重	爱好
3	010107	陈和	高一（1）女		1.73	65	其他
4	010133	陈凤灵	高一（1）女		1.72	58	羽毛球
5	010114	季潇潇	高一（1）女		1.70	60	羽毛球
6	010134	陆怡人	高一（1）女		1.68	53	篮球
7	010140	全美	高一（1）女		1.65	51	羽毛球
8	010142	李冰洁	高一（1）女		1.65	52	羽毛球
9	010146	简菲菲	高一（1）女		1.63	49	羽毛球
10	010123	张和梅	高一（1）女		1.62	51	羽毛球
11	010111	樊嘉诚	高一（1）女		1.61	65	羽毛球

图1.2.1 多条件排序的结果

● 数据的筛选

根据一定的规则成批筛选数据，有助于我们发现某类特定数据蕴含的信息。筛选会依据条件显示需要的数据，其余的内容都会被隐藏起来。同时，筛选不会以任何方式更改数据，取消筛选之后，所有数据都会重新出现。

例如，筛选性别为男、爱好为篮球的数据，筛选后的结果如图 1.2.2 所示。

2	编号	姓名	班级	性别	身高	体重	爱好
29	010113	许朝天	高一（1）	男	1.78	60	篮球
30	010135	鲁天羊	高一（1）	男	1.78	80	篮球
33	010110	杨建博	高一（1）	男	1.76	64	篮球
34	010120	李信哲	高一（1）	男	1.76	65	篮球
35	010103	洪恩凯	高一（1）	男	1.75	65	篮球
36	010117	罗先伟	高一（1）	男	1.74	65	篮球
38	010105	刘浩然	高一（1）	男	1.73	73	篮球
39	010101	李文惠	高一（1）	男	1.72	59	篮球
40	010115	唐含	高一（1）	男	1.72	60	篮球
44	010122	周武	高一（1）	男	1.71	62	篮球
46	010102	赵家石	高一（1）	男	1.70	51	篮球

图1.2.2 筛选性别为男、爱好为篮球的数据

● 公式和函数

在选择班级运动队队员时，体重是一个很重要的参考因素，我们可以计算出各位同学的体重指数，如图 1.2.3 所示。

SUM × ✓ fx =F3/(E3*E3)							
	A	B	C	D	E	F	G
1	班级运动队意向调查表						
2	编号	姓名	班级	性别	身高	体重	爱好
3	010101	李文惠	高一(1)	男	1.72	59	篮球
4	010102	赵家石	高一(1)	男	1.70	51	篮球

图1.2.3 用公式计算各位同学的体重指数

计算全班同学的平均体重时，我们可以使用电子表格软件提供的函数功能“=AVERAGE(F3:F49)”求得结果。常用的函数如表1.2.2所示。

表1.2.2 WPS常用函数表

函数	SUM	AVERAGE	COUNT	MAX	MIN
功能	求和	求平均值	计数	求最大值	求最小值



体重指数 (BMI) 是目前国际上常用的衡量人体胖瘦程度以及是否健康的标准之一。成年人体重指数的计算方法是：体重 ÷ (身高 × 身高)，这里体重的单位为千克，身高的单位为米。对于我们中国人来说，正常的体重指数范围是 18.5~23.9。

● 分类汇总

若要进行数据的分类汇总，首先要按照分类字段排序，以便将要进行分类汇总的行排列在一起，然后依据汇总方式计算选定汇总项。当插入分类汇总时，列表将分级显示，我们可以灵活选择显示或隐藏明细数据行。

先按性别、爱好排序，再设定分类字段为“爱好”、汇总方式为“计数”、选定汇总项为“爱好”，然后进行分类汇总，分类汇总的结果如图1.2.4所示。

1	2	3	A	B	C	D	E	F	G
	2		编号	姓名	班级	性别	身高	体重	爱好
	3		010101	李文惠	高一(1)	男	1.72	59	篮球
	4		010102	赵家石	高一(1)	男	1.70	51	篮球
	5		010103	洪恩凯	高一(1)	男	1.75	65	篮球
	6		010105	刘浩然	高一(1)	男	1.73	73	篮球
	7		010110	杨建博	高一(1)	男	1.76	64	篮球
	8		010113	许朝天	高一(1)	男	1.78	60	篮球
	9		010115	唐含	高一(1)	男	1.72	60	篮球
	10		010117	罗先伟	高一(1)	男	1.74	65	篮球
	11		010120	李信哲	高一(1)	男	1.76	65	篮球
	12		010122	周武	高一(1)	男	1.71	62	篮球
	13		010135	鲁天羊	高一(1)	男	1.78	80	篮球
	14							篮球 计数	11
	15		010126	周威波	高一(1)	男	2.00	93	排球
	16							排球 计数	1

图1.2.4 分类汇总的结果

借助WPS表格，我们列表得出了“鸡兔同笼”问题的鸡兔数量，如图1.2.5所示。

1	“鸡兔同笼”问题（35头94脚）				
2	兔的只数	鸡的只数	兔脚总数	鸡脚总数	共有的脚数
14	12	23	48	46	94

图1.2.5 鸡兔数量

※ 活动3 编程解决“鸡兔同笼”问题

打开教科书配套资源中的jitu.py程序，体验编程解决“鸡兔同笼”问题的过程。

运行程序，只需要输入总的头数和总的脚数，程序瞬间就给出了鸡和兔的只数，求解的速度很快。运行结果如图1.2.6所示。

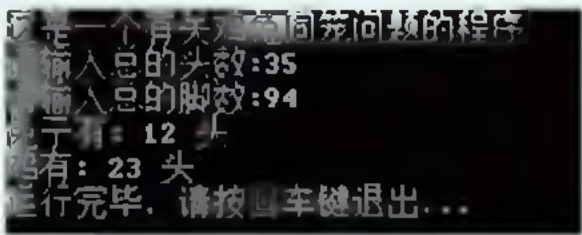


图1.2.6 “鸡兔同笼”问题的程序运行结果

 任务二 解决“鸡兔同笼”问题的计算方式比较

面对“鸡兔同笼”问题，与人工计算和借助电子表格软件计算相比较，编程计算进一步提高了效率。请比较人工计算、借助电子表格软件计算和编程计算三种计算方式的特点，填写表1.2.3。

表1.2.3 不同计算方式的比较

计算方式	考查的方面			
	使用成本	使用方便性	计算速度	其他
人工计算				
借助电子表格 软件计算				
编程计算				

三种计算方式都有其适用的问题类型，灵活使用这三种方式将有助于我们提高数据计算的效率。



拓展练习

“百鸡百钱”问题是一个有名的数学问题，出自《张丘建算经》。其内容是：公鸡5文钱1只，母鸡3文钱1只，小鸡3只1文钱，用100文钱买100只鸡，其中公鸡、母鸡和小鸡都必须要有，问公鸡、母鸡和小鸡各多少只？请尝试使用人工计算、借助电子表格软件计算和编程计算三种方式解决问题，编程计算的程序可以从教科书配套资源中获得。



拓展知识

超级计算机

全球超级计算机500强榜单由国际组织TOP500每半年公布一次，目的是促进国际超级计算机领域的交流和合作，促进超级计算机的推广应用。2018年11月12日，新一期全球超级计算机500强榜单公布，中国“神威·太湖之光”和“天河二号”分别位列第三、第四名。

“神威·太湖之光”这款由国家并行计算机工程技术研究中心研发的超级计算机，峰值速度达到12.5亿亿次每秒，常规速度为9.3亿亿次每秒。简单地说，这套系统1分钟的计算能力，相当于全球72亿人同时用计算器不间断计算32年。它也是全球首台运行速度超十亿亿次的超级计算机。更重要的是，“神威·太湖之光”里包括处理器在内的所有核心部件全部为国产，核心技术完全不依赖国外力量。

超级计算机是一个国家科技能力的象征，它可以解决交通工具制造、气候问题、生物信息、地震监测、地球科学、天体物理、公共健康、材料科学等诸多领域的挑战性问题。20世纪80年代初，我国决心自行研发超级计算机，于是历史上知名的“银河”系列超级计算机在国防科技大学诞生，此后中国科学院又研制了“曙光”系列，现在知名的还有“神威”系列、“天河”系列。2010年11月，“天河一号”曾以4.7千万亿次每秒的峰值速度首次夺得超级计算领域冠军。2013年6月，“天河二号”以5.49亿亿次每秒的峰值速度再次夺得超级计算领域冠军，并连续六次占据榜首，直至2016年6月，“神威·太湖之光”取得冠军。截至2017年6月，中国产的超级计算机已经连续霸榜九期。

单元学习评价

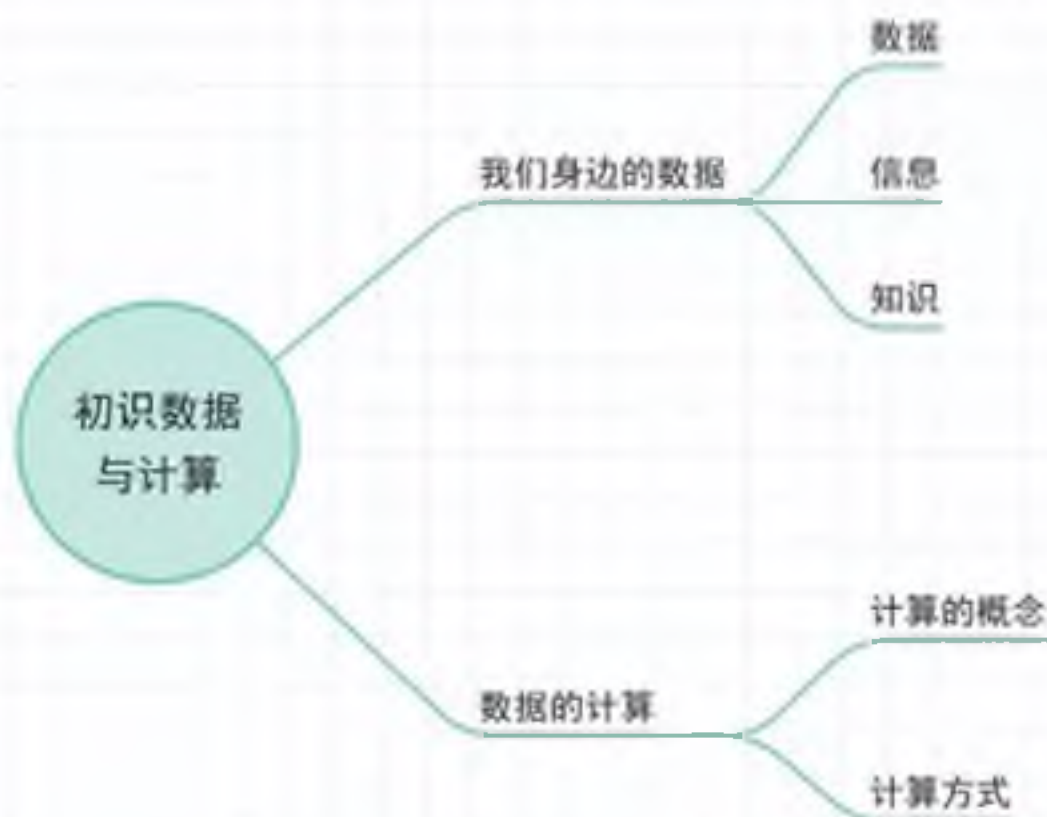
1. 有“成年人清晨安静状态下的口腔正常温度在 $36.3 \sim 37.2^{\circ}\text{C}$ ”“38”和“叶达的口腔温度是 38°C ”三种描述，这三种描述分别是_____、_____和_____（填写数据、信息或知识）。结合本例简要说明数据、信息与知识的关系。

2. （单选）利簋，西周早期青铜器，1976年出土于陕西省临潼县零口镇。器内底铸铭文4行，共33字，记载了甲子日清晨武王伐纣这一重大历史事件。信息记载于簋上体现的信息基本特征是（ ）。

A. 载体依附性 B. 价值性 C. 时效性 D. 共享性

3. 1972年图灵奖得主艾兹格·迪科斯彻（Edsger Dijkstra）说：“我们所使用的工具影响着我们的思维方式和思维习惯，从而也深刻影响着我们的思维。”请以“来自计算工具发展的启示”为题，寻找有代表性的计算工具，比较它们在数据表示、数据存储和计算规则方面的差异，从而了解计算工具的发展，并思考数据及计算工具的发展对我们日常生活的影响。

单元学习总结



第 2 单元 编程计算

“阿尔法围棋”（AlphaGo）是第一个战胜人类世界围棋冠军的人工智能程序。这是计算机的胜利，也是“计算机程序”的胜利，更是程序编写者——“人”的胜利。信息论的奠基人克劳德·埃尔伍德·香农（Claude Elwood Shannon）在1950年就说过：“研究下棋问题旨在开发更为实用的技术。”因此，“阿尔法围棋”存在的意义在于帮助人们研究如何最大限度地发挥计算机的优势，更好地为人类服务。如果我们学会与计算机沟通的语言，那么我们也可以编写计算机程序，让计算机帮我们解决问题。

通过计算机编程解决问题的过程是怎样的？程序又是如何执行的？这些是本单元需要讨论的问题。

本单元我们将通过编制有趣的程序，探究计算机解决问题的过程，初步了解算法的三种基本结构，即顺序结构、分支结构和循环结构，以及如何对算法进行描述，熟悉计算机程序设计语言——Python的开发环境，学会编写简单的程序解决问题，并尝试发布程序代码，与他人分享。



每个人都应该学习如何编程，因为它教会你如何思考。

——史蒂夫·乔布斯（Steve Jobs）

2.1 计算机解决问题的过程

要通过编程解决问题，首先需要通过分析找出解决问题的方法和步骤；然后将待处理的信息用计算机能识别的方式表示，使之成为计算机能处理的数据；最后编写并调试程序，使之无误并能顺利运行。本节我们将通过完成一个项目，体验这些过程。



学习目标

- ★ 了解计算机解决问题的过程。
- ★ 了解流程图的规范表示方法，学会使用自然语言和流程图描述算法。
- ★ 了解算法的三种基本结构。

软件工程师是将人类想法翻译成计算机语言的译者。他懂得让计算机完成任务的方法，能通过富有逻辑、清晰明了的语言，把方法提供给计算机。本节我们将围绕“智力游戏”项目展开学习，通过该项目的活动，初步体验使用计算机解决问题的过程。

本项目主要包含“表示游戏的算法”和“用Python语言实现游戏”两个任务。

“最强大脑”游戏的游戏规则为：用10秒记忆10个物品以及它们的编号，之后根据提问回答5个物品的编号，看谁记住的物品编号数量多。如何通过计算机编程实现这个游戏？其实，要用计算机解决问题，算法设计是最重要的环节。



任务一 表示游戏的算法

※ 活动1 使用自然语言描述算法

● 算法

简单地说，算法就是解决问题的方法和步骤。“最强大脑”游戏问题的求解过程就代表着一种算法。其实，我们在日常生活中也经常使用算法。例如，我们到超市购物，首先确定要购买的物品，然后进行挑选、比较，最后到收银台结账付款，这一系列活动流程与策略就是所谓的算法。解决问题的过程，就是实现算法的过程。

“最强大脑”游戏问题的算法分析：显示游戏规则，显示物品以及编号，屏幕停留10秒；随机出题，给出物品，回答编号；判断回答是否正确，如果正确，则答对的题数加1。重复以上步骤，直到答完5道题；显示答对的题数。

找到算法之后，不能只是自己心知肚明，还需要准确、具体地将它描述出来。要让计算机解决问题，必须明确地告诉它要处理的具体对象和每一步的准确处理过程，否则计算机就无法工作。因此，对算法的描述要求尽可能精确、详尽。

描述算法可以有多种方法，比如可以使用自然语言或流程图进行描述。

● 使用自然语言描述算法

自然语言是指人们日常生活中使用的语言，汉语、英语、德语等都是自然语言。用自然语言描述算法符合我们的表达习惯，并且容易理解。用自然语言描述算法，通常是我们理解算法的第一步。

思考实现“最强大脑”游戏问题的算法，用自然语言描述“最强大脑”游戏问题的算法，填写表2.1.1。

表2.1.1 实现“最强大脑”游戏问题的步骤

环节	解决的问题	解决的步骤
出题	显示物品和编号10秒	1.
		2.
		
答题		
输出结果		

用自然语言描述算法简单且通俗易懂，但容易产生歧义，表示的含义往往不大严格。我们也可以使用流程图来描述算法。

※ 活动2 使用流程图描述算法

试画出实现“最强大脑”游戏问题的流程图。

● 使用流程图描述算法

流程图也称为程序框图，它是算法的一种图形化表示方法。与使用自然语言描述算法相比，用流程图描述算法形象、直观、更容易理解。对于一些复杂的算法，直接编写代码很困难，我们可以先用流程图描述算法，然后根据流程图写出程序代码。

为了使算法的流程图看起来更加直观，需要对流程图的表示方法进行规范，目前最常用的流程图规范是由美国国家标准学会制定的一系列流程图符号，如图2.1.1所示。

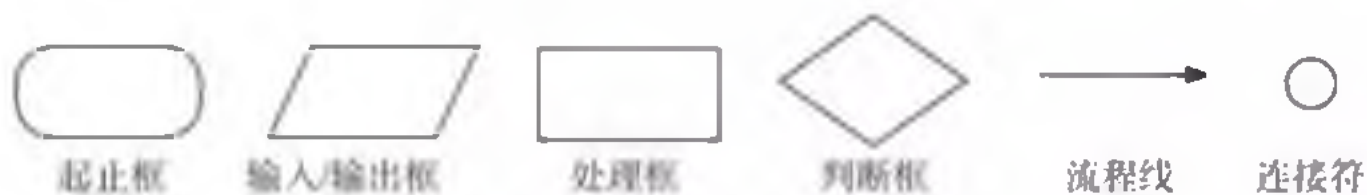


图2.1.1 流程图符号

- ◆ 起止框：表示一个算法的开始和结束。
- ◆ 输入/输出框：表示从外部输入数据到计算机内部或者从计算机内部输出数据到计算机外部。
- ◆ 处理框：表示操作的内容。
- ◆ 判断框：表示判断的条件。满足条件，执行标识为“是”的路径；不满足条件，则执行标识为“否”的路径。
- ◆ 流程线：指向算法运行的方向。
- ◆ 连接符：表示流程图的接续。在相互联系的流程图内，流程线将在具有相同数字或字母的另一连接符处继续下去。

● 三种基本结构的流程图

算法的三种基本结构是顺序结构、分支结构和循环结构。

顺序结构是一种最简单的基本结构，按照自上而下的先后顺序依次执行程序语句。在如图2.1.2所示的顺序结构示意图中，先执行处理框A，再执行处理框B。

分支结构也称为选择结构，是一种根据给定的条件进行判断的结构。如图2.1.2所示的分支结构中必定包括一个判断框，若满足条件则

执行处理框A，若不满足条件则执行处理框B。

循环结构是一种重复某一部分操作的结构。如图2.1.2所示的循环结构中，若满足条件则执行循环体（处理框A），若不满足条件则跳出循环，执行循环结构后面的代码。

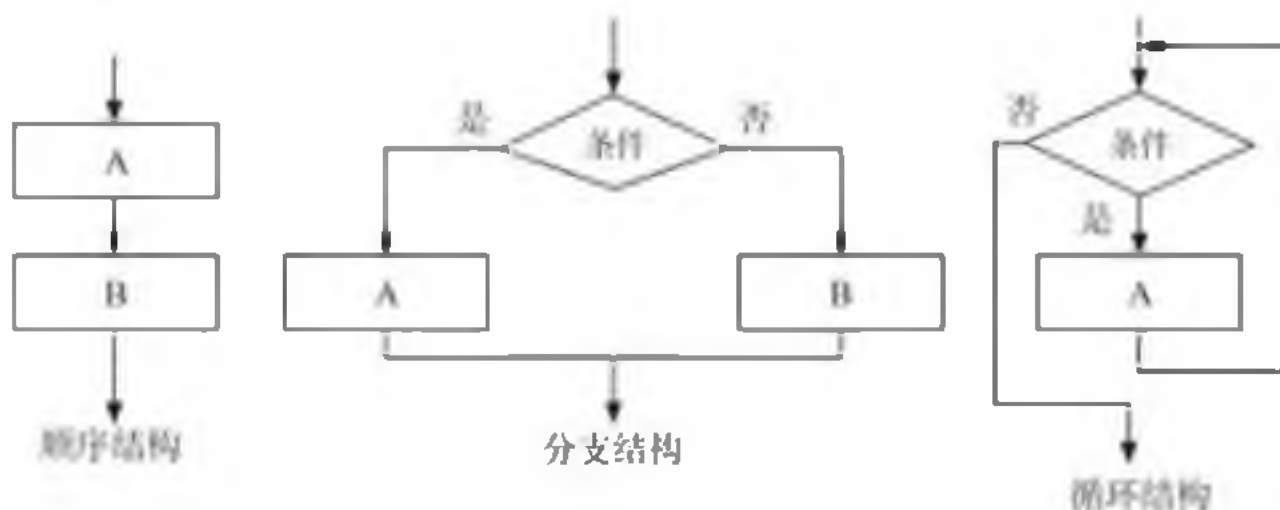


图2.1.2 三种基本结构的流程图

“最强大脑”游戏问题的算法可以用这三种基本结构的流程图来表示。如表2.1.2所示，请对照算法的自然语言描述，试画出每个环节的流程图。

表2.1.2 自然语言与流程图

环节	自然语言	流程图
出题	1. 显示游戏规则、物品及其编号10秒 2. 清屏	
答题	1. 给出一个物品名称 2. 回答编号 3. 如果回答正确，答对的题数加1 4. 重复以上步骤，直到答完5道题	
输出结果	输出答对的题数	

使用流程图可以对算法进行描述，但是流程图一般不能直接在计算机上运行，这时需要使用一种计算机能理解的程序设计语言将算法表达成程序。



程序设计语言
泛指一切用于书写
计算机程序的语言。



任务二 用Python语言实现游戏

※ 活动1 运行程序

请在Python语言的开发环境中，调试运行程序，观察运行结果。

以#开头的代码是注释语句，用于说明代码实现的功能、采用的算法等信息，可提高程序的可读性，不参与程序的执行。

```
import random
import time
import os
print("你好，现在你有10秒的时间记忆下列物品及其编号")
things=["苹果","香蕉","橙子","梨子","猕猴桃","柚子",
        "猴魁","铁观音","彩蛋","复活节"]
for i in range(10):
    print(i,":",things[i])           #在屏幕上显示物品及编号
time.sleep(10)                       #延时10秒
os.system("cls")                     #清屏
n=0                                  #记录答对的题数，初值为0
t2=random.sample(things,5)          #随机抽出5个物品
for i in t2:                          #出5题
    ans=int(input(i + "的编号是:"))  #输入编号答题
    if i==things[ans]:
        n=n+1                        #如果回答正确，答对的题数加1
print("\n你一共答对了",n,"次")      #屏幕显示答对的题数
input("\n按回车键结束程序")
```

● Python语言的开发环境

计算机程序是指示计算机解决问题或完成任务的一组可执行的指令。

Python是一种程序开发工具。下面我们用Python3.4.4一步一步尝试运行程序，实现“最强大脑”游戏。

启动Python3.4.4，在Python shell中打开多行代码编辑窗口。编写或直接粘贴程序并运行，运行中会要求保存一个.py的文件，运行结果如图2.1.3所示。也可双击.py文件直接运行，观察运行结果的差异。

```
你好，现在你有10秒的时间记忆下列物品及其编号
0: 苹果
1: 香蕉
2: 橙子
3: 梨子
4: 猕猴桃
5: 柚子
6: 猴魁
7: 铁观音
8: 彩蛋
9: 复活节
柚子的编号是: 5
猕猴桃的编号是: 4
```

苹果的编号是：0
 彩蛋的编号是：8
 假题的编号是：9

你一共答对了4次

按回车键结束程序

图2.1.3 运行结果

※ 活动2 借助流程图分析代码

借助“最强大脑”游戏问题的完整流程图，分析活动1的代码，尝试找出与图2.1.4中的处理框对应的代码。

 代码是指一系列计算机语言指令。

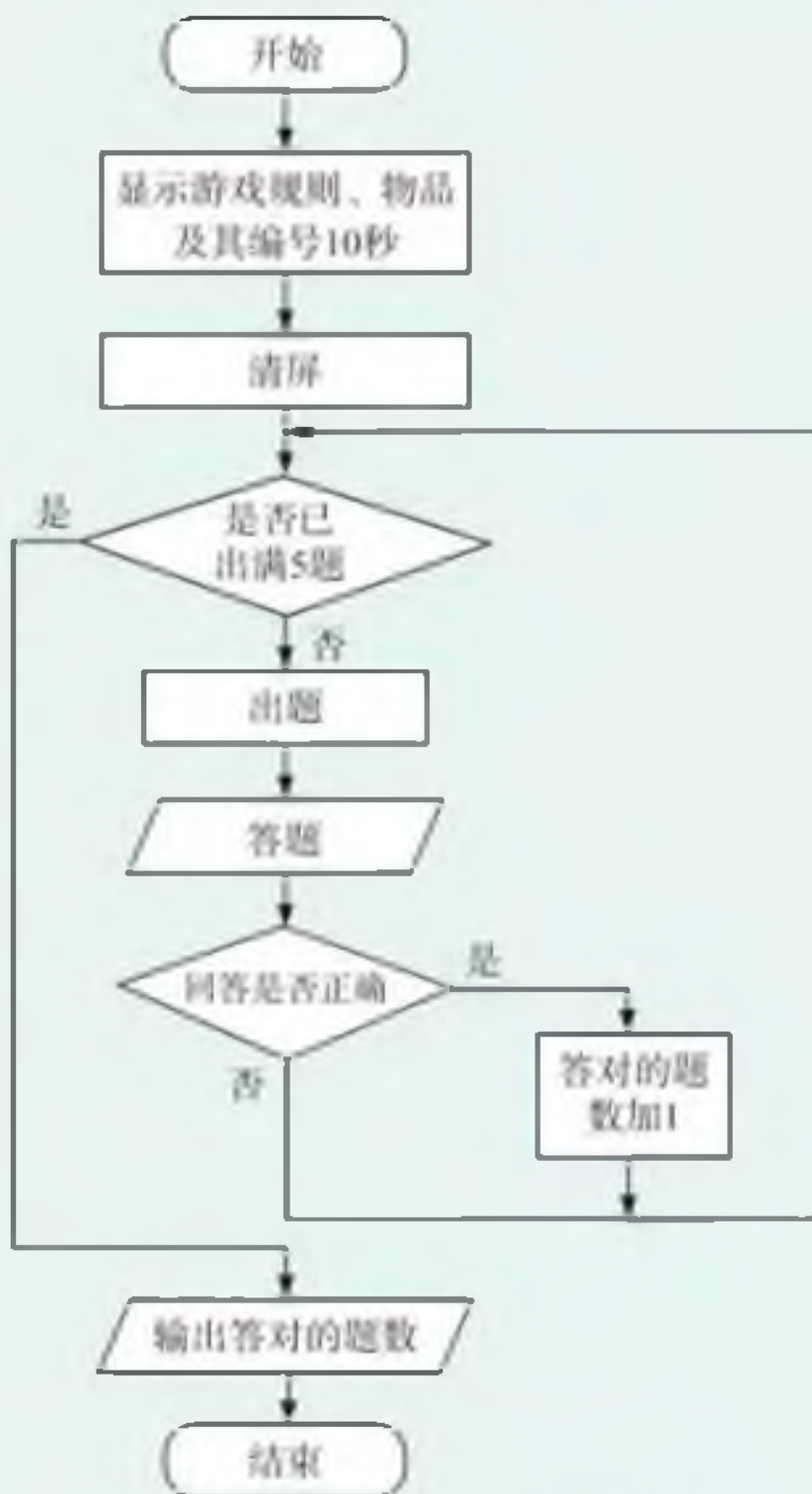


图2.1.4 “最强大脑”游戏问题流程图

● 程序设计语言的发展

除Python语言之外，还有很多其他程序设计语言。程序设计语言经历了从机器语言、汇编语言到高级语言的发展过程。

机器语言由二进制的 0、1 代码指令构成，能被计算机直接识别。但理解和记忆机器语言非常困难，并且容易出错，编程效率极低。

汇编语言是符号化的机器语言，采用英文助记符代替机器指令，比机器语言容易识别和记忆，从而提高了程序的可读性。但是汇编语言仍然是面向机器的语言，是为特定的计算机系统设计的，它要求软件工程师对相应的机器硬件非常熟悉，因而汇编语言属于低级语言。

高级语言更接近自然语言，并不特指某一种语言，也不依赖于特定的计算机系统，因而更容易掌握和使用，通用性也更好。比较流行的高级语言有Java、C/C++以及本书使用的Python等。用高级语言编写的程序可读性更强，也便于修改、维护。



拓展练习

“最强大脑”游戏程序实现了10秒记忆10件物品及其编号的任务。请尝试在原有的程序代码中进行修改，以增加“最强大脑”游戏的难度。观察程序运行的变化，填写表2.1.3。

表2.1.3 修改程序代码

原代码	修改过的代码	程序运行的变化



拓展知识

Python是一种面向对象的解释型计算机程序设计语言，发明于1989年。它在反映编程语言热门程度的TIOBE编程语言排行榜上曾多次名列前茅。目前，Python已经成为最受欢迎的程序设计语言之一，它具有简单、免费、开源和可移植性等特点。

简单：Python是一种代表简单主义思想的语言，它的这种伪代码本质是其最大的优点之一。它使人们能够专注于解决问题而不是去搞明白语言本身。

免费、开源：可以自由地发布这个软件、阅读它的源代码、对它做改动、把它的一部分用于新的自由软件中。

可移植性：由于它的开源本质，Python已经被移植在许多平台上，如Linux、Windows、Android等。

2.2 做出判断的分支

使用计算机解决问题时，必须准确地告诉计算机应该进行哪些操作以及进行这些操作的步骤。如果各个步骤是按照自上而下的顺序一步一步进行的，那么该流程是顺序结构；如果需要根据“条件判断”来决定下一步的走向，那么该流程是分支结构。通过本节的学习，我们将掌握如何运用顺序结构、分支结构控制程序流程，学会使用赋值语句、if语句编写简单的程序，初入程序设计的大门。



学习目标

- ★ 掌握顺序结构、分支结构的基本结构。
- ★ 学会赋值语句、if语句的使用方法。
- ★ 运用顺序结构、分支结构的语句编写简单的程序解决问题。

我们知道运动强度要因人而异，并可以通过监测运动时的心率掌握适当的运动量。本节我们将围绕“最适宜运动心率计算器”项目展开学习，通过该项目了解顺序结构、分支结构的基本结构，掌握Python中使用赋值语句及if语句实现顺序结构和分支结构的方法，并体验调试程序的过程。

本项目主要包含“利用赋值语句实现计算”和“利用分支语句实现分性别计算”两个任务。

某同学参加了学校的体检，体检医生告诉他体重超重，需要加强运动。他了解到，运动时心率如能控制在一定范围内，对增强心血管系统和呼吸系统的功能都有很好的效果；超出范围，对身体反而会有损伤。他想知道自己最适宜的运动心率。



任务一 利用赋值语句实现计算

※ 活动1 计算最适宜运动心率

该同学今年16岁，安静心率为71。

最适宜运动心率 = $(220 - \text{年龄} - \text{安静心率}) \times (60\% \sim 80\%) + \text{安静心率}$

直接把他的年龄与安静心率的数值代入最适宜运动心率的公式中就可以算出最适宜运动心率，请将如图2.2.1所示的流程图补充完整。

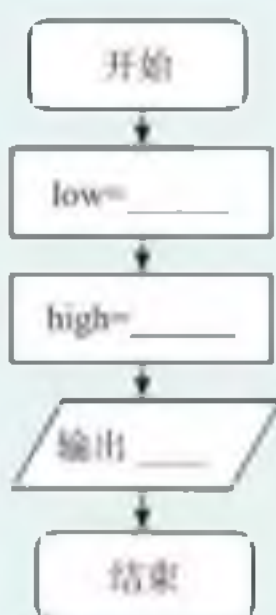


图2.2.1 活动1流程图

● 常量

在程序运行过程中值始终保持不变的量称为常量。所有的常数都是常量。例如 π 、活动1中出现的220、0.6等。

● 变量

在程序运行过程中值允许改变的量称为变量。例如，活动1中的low、high就是变量。计算机在处理数据时，首先要将数据调入内存单元，然后进行各种操作。变量是创建时在内存中开辟的一个存储空间，变量名是内存中存放数据的存储单元的标签，存放的数据叫变量的值。在Python中，变量的命名需遵循以下几条规则。

- ◆ 必须以字母或下划线开头，后面可以是字母、数字或下划线。
- ◆ 区分大小写。
- ◆ 不能使用保留字。

程序中的变量与数学中的变量含义略有不同。数学中的变量隐含着“未知”；在程序中，程序执行的每个瞬间，变量的值是确定的，但在程序执行的整个过程中，它的值是可以改变的。

● 运算符与表达式

表达式由一个或多个操作数通过运算符组合而成。例如“2 + 3”，它由运算符“+”和操作数“2”“3”两部分构成。运算符是进行某种运算的标识符号，它标明对操作数所进行的运算。常见的运算符如表 2.2.1 所示。

表2.2.1 运算符

运算符	+	-	•	**	/	//	%
功能	加	减	乘	幂	除	整除	取模

我们可以在Python shell的命令提示符（>>>）后，一行一行地输入命令，观察运行结果。

```
>>>4*2
8
>>>4/2
2.0
>>>4//3
1
>>>4%2
0
>>>4**3
64
>>>
```

● 赋值语句

赋值语句的基本格式如下：

变量 = 表达式

功能是将右边表达式的值赋值给左边的变量。

在Python中，通过给变量赋值来使用变量。我们可以在Python shell的命令提示符后输入命令，观察运行结果。

```
>>>age=17
>>>print(age)
17
>>>n=4
```

```
>>>print(n)
4
>>>n=n+1
>>>print(n)
5
>>>
```

赋值语句“ $n=n+1$ ”表达的含义是：先计算赋值号右边的表达式，也就是计算“ $n+1$ ”的值，然后将这个值“赋”给左边的变量“ n ”。

Python中的赋值语句较为灵活，例如：

<code>a=b=c=1</code>	#数值1赋值给变量a,b,c
<code>a,b,c=1,2,3</code>	#数值1,2,3分别赋值给变量a,b,c
<code>a,b=b,a</code>	#交换a,b两个变量的值
<code>c+=a</code>	#等效于 $c=c+a$

请将下列程序代码补充完整，并调试运行。注意避免常见错误，如变量名拼写错误、输入了中文的标点符号等。

<code>low=(220-__-__)*0.6+__</code>	#计算最适宜运动心率低值
<code>high=(220-__-__)*0.8+__</code>	#计算最适宜运动心率高值
<code>print(low,high)</code>	#显示最适宜运动心率的范围

该同学把计算器展示给其他同学，其他同学也想计算自己的最适宜心率，可是每次都要修改程序代码中的年龄和安静心率会很麻烦。那么可以怎么做呢？

※ 活动2 计算任意年龄与安静心率的最适宜运动心率

在程序运行时输入年龄和安静心率，计算出最适宜运动心率。请将如图2.2.2所示的流程图补充完整。

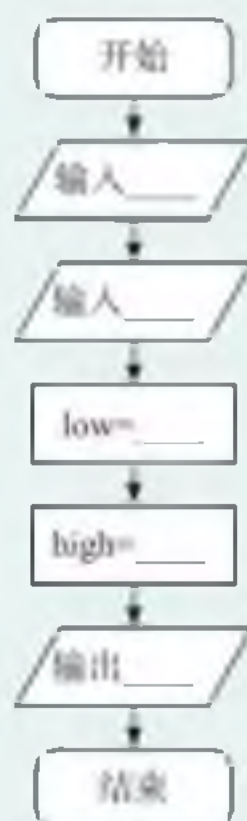


图2.2.2 活动2流程图

● 输入和输出语句

在Python中，使用“input()”从键盘输入数据，使用“print()”在屏幕上输出数据。如果有多个输出项，可以使用“,”分隔。下面先来看一个简单的例子，这个程序用来输入姓名，输出简单的问候。

```
myname=input("请输入myname:")
print("hello",myname)
```

运行结果：

```
请输入myname:Justin
hello Justin
```

代码中的“hello”表示字符串，字符串就是文本，可以是字母、数字或符号。字符串与数字的区别在于，字符串两端需加引号（'或"），输出时原样显示引号中的内容。我们可以改写最适宜运动心率的输出语句，使其显示更为友好。例如：

```
print ("最适宜的运动心率是：",low,"~",high)
```

程序运行的结果如图2.2.3所示。

```
请输入年龄=16
请输入安静心率=71
最适宜的运动心率是：150.8~177.4
```

图2.2.3 活动2运行结果

参考图2.2.3的运行结果，将下列程序代码补充完整，并调试运行。

age=float(input("请输入年龄="))	#输入年龄
_____	#输入安静心率
low=(220-age-HRrest)*0.6+HRrest	#计算最适宜运动心率低值
high=(220-age-HRrest)*0.8+HRrest	#计算最适宜运动心率高值
print(_____)	#显示最适宜运动心率的范围

程序运行时通过键盘输入的内容是字符串，它是怎样转化成数值再进行运算的呢？



HRrest 表示安静心率，指正常人安静状态下每分钟心跳的次数，一般为60~100次/分。

● 数据类型

为了便于数据的表示与处理，Python提供了整数、浮点数和字符串等常见的数据类型。下面给出一些函数，它们可以把数据从一种类型转换为另一种类型。

`float()`：根据一个字符串或整数创建一个新的浮点数（小数）。

`int()`：根据一个字符串或浮点数创建一个新的整数。

`str()`：根据一个数（也可以是任何其他类型）创建一个新的字符串。

男性最适宜运动心率 = $(220 - \text{年龄} - \text{安静心率}) \times (60\% \sim 80\%) + \text{安静心率}$

女性最适宜运动心率 = $(210 - \text{年龄} - \text{安静心率}) \times (60\% \sim 80\%) + \text{安静心率}$

以上计算最适宜运动心率的公式是有性别差异的，那么如何修改程序可以实现根据性别输出结果呢？



任务二 利用分支语句实现分性别计算

修改程序，要求根据性别进行正确计算。请将如图2.2.4所示的流程图补充完整。

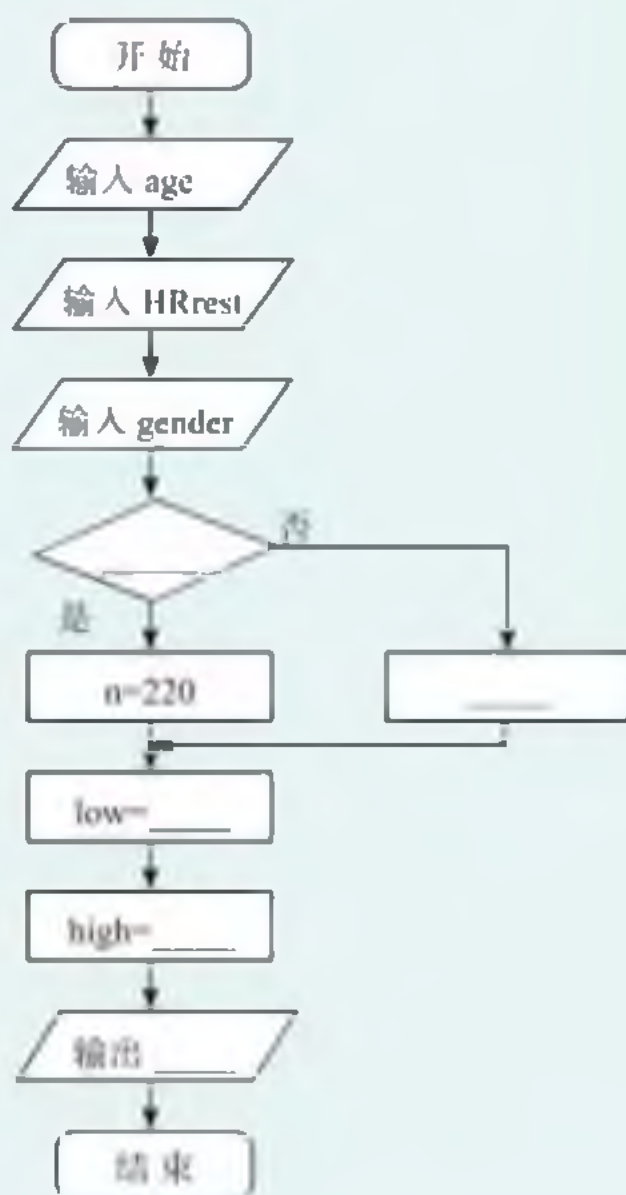


图2.2.4 任务二流程图

● 分支语句

当程序中需要根据不同的条件判断来决定程序执行的走向时，可以使用if分支语句实现，其基本格式如下。

```
单分支
if 条件:
    语句或语句组
双分支
if 条件:
    语句或语句组A
else:
    语句或语句组B
```

下面先来看一个简单的例子，这个分支程序用来判断飞船是否能够升空。当飞船速度“v>=7.91”时，输出“飞船成功飞行！”；否则输出“飞船不能升空！”。

```
v=float(input("请输入速度(千米/秒):"))
if v>=7.91:
    print("飞船成功飞行!")
else:
    print("飞船不能升空!")
```

分支语句的条件“v>=7.91”表达式，一般称为关系表达式。关系表达式是用关系运算符连接起来的式子。常用的关系运算符如表2.2.2所示。

表2.2.2 关系运算符

运算符	==	>	>=	<	<=	!=
含义	等于	大于	大于等于	小于	小于等于	不等于

当关系表达式成立时值为真（True），不成立时值为假（False）。对于数值，是按照数值的大小来比较它们的关系的。例如：9>=7.9，其值为True；23==24，其值为False。

```
将下列程序代码补充完整，并调试运行。
age=float(input("请输入年龄=")) #输入年龄
```

```

HRrest=float(input("请输入安静心率="))           #输入安静心率
gender=input("请输入male 或 female:")           #输入性别
if _____:                                     #根据输入的性别判断
    _____
else:
    _____
low=(n-age-HRrest)*0.6+HRrest                     #计算最适宜运动心率低值
high=(n-age-HRrest)*0.8+HRrest                   #计算最适宜运动心率高值
#输出最适宜运动心率的范围
print ("最适宜的运动心率是：",low,"~",high)

```



拓展练习

对以上程序进行修改，使其可以输入年龄、安静心率和运动后心率，并能依据最适宜运动心率公式输出反馈结果，如“您的运动心率太低，请适当提高”“您的运动心率正好，请保持”“您的运动心率太高，请适当降低”。



拓展知识

对于多分支语句的情况，Python 语言通常使用 if-elif-else 语句来处理。参照表1.1.3所示的感冒指数，根据输入的感冒指数，输出该指数所对应的含义。实现程序的代码如下。

```

num=int(input("请输入感冒指数："))
if 0<=num<=6 :
    print("少发")
elif 7<=num<=19:
    print("较易发")
elif 20<=num<=30:
    print("易发")
elif 31<=num<=61:
    print("极易发")
else:
    print("指数值不正确")

```

2.3 周而复始的循环

对人来说，长时间重复做同一件事情，通常会觉得不愉快，且效率会有所下降。但计算机往往不知疲倦，非常适宜用重复的方式完成任务。我们将计算机程序中某些代码的反复执行称为循环。Python中既有重复一定次数的计数循环，也有重复到某种情况结束的条件循环。



学习目标

- ★ 了解循环的基本结构。
- ★ 学会计数循环的使用方法。
- ★ 学会条件循环的使用方法。
- ★ 熟练使用循环解决实际问题。

随着“理财时代”的到来，“你不理财，财不理你”成为人们的一句口头禅。我国金融市场日益繁荣，为个人投资者提供的投资理财产品和渠道越来越多，如储蓄存款、股票、债券、保险等。因此，理财规划的重要性日益凸显。本节我们将围绕“学习投资理财”项目展开学习，通过该项目活动，了解循环的作用，学习如何根据问题的需要选择恰当的循环类型，如何准确设置循环条件和编写循环语句块。

本项目主要包含“计算5年储蓄存款收益”和“计算理财产品收益”两个任务。



任务一 计算5年储蓄存款收益

※ 活动1 计算利率不变情况下的到期存款总额

如果将5万元存入银行，选择“1年定期、自动转存”的存款方式，假设5年内存款利率没有发生变化，均为3.25%，5年后到期存款

总额是多少？在程序中，暂不考虑存款分位以下的四舍五入。

※ 活动2 计算利率变动情况下的到期存款总额

如果首次存款时的利率是3.25%，自动转存时利率分别是3%、3%、2%、1.75%，那么5年后到期存款总额是多少？

由于每年的利率不同，如果都存储在变量中，那么需要5个变量，这时可以借助列表来实现。



自动转存：存款到期后，客户如不办理转存手续，银行可自动将到期的存款本息按相同存期一并转存，不受次数限制，续存期利息按前期到期日利率计算。

● 列表

列表在Python中的作用非常强大，它可以包含多个数据元素，同时它是一个有序的集合。下面通过一个简单的例子来熟悉列表的使用方法。

```
#创建列表，各元素用逗号隔开，放在方括号内
#列表可以存储混合类型的数据
object=["石榴",1,"香蕉","橙子","梨子"]
#修改列表第1个元素的值，注意列表索引号从0开始
object[0]="苹果"
del object[1]      #删除列表第2个元素
#在列表尾部添加一个数据元素"猕猴桃"
object.append("猕猴桃")
print (object)     #输出列表
```

运行结果：

```
['苹果','香蕉','橙子','梨子','猕猴桃']
```

用列表表示各年的利率，即`rate=[0.0325,0.03,0.03,0.02,0.0175]`，那么存款总额的计算公式是：本金 * (1 + rate [0]) * (1 + rate [1]) * ... * (1 + rate [4])。存款期限越长，公式就越复杂。有没有其他解决方法呢？

请分析每年的存款变化，从中寻找规律。

第1年 存款总额=本金*(1+第1年利率)

第2年 存款总额=上一年存款总额*(1+第2年利率)

第3年 存款总额=上一年存款总额*(1+第3年利率)

.....

第n年 存款总额=_____

你发现其中的规律了吗？

在计算每一年的存款总额时，计算公式都很相似。在解决这类反复执行某些代码的问题时，采用循环语句是常用的方式之一。

● 计数循环

循环重复次数一定时，我们通常称为计数循环。在Python中，使用for语句来创建这类循环。for语句的基本格式如下。

```
for 循环变量 in 列表:
    语句或语句组
```

下面先来看一个简单的例子，该例子通过for语句输出列表中的每个元素。

```
for i in ["hello","world"]:
    print (i)                #循环体
```

运行结果：

```
hello
world
```

通常我们将一组重复执行的语句称为循环体。Python对循环体约定的格式是：循环体必须相对关键词for有缩进。在本例中，循环体只有一条语句，即print(i)。循环执行时，依次将列表中的元素赋给i，并且执行循环体（输出i的值），直到整个列表的元素被取完，循环结束。

在for语句中，in后也可以是range()函数，它可以生成某个范围内的数字列表。例如：range(1,6)就会生成[1,2,3,4,5]这样一个列表，而range(8)会生成[0,1,2,3,4,5,6,7]这样一个列表。下面这个循环可以输出0到100的自然数。

```
for i in range(101):          #想想为什么是101
    print (i)
```

根据前面分析得出的规律，可以使用计数循环来实现每年存款总额的重复计算。循环的次数即为存款的年限。因为银行支付利息计至分位，分位以下四舍五入，所以可以调用round()函数实现保留两位小数。例如，round(1.428,2)的运算结果为1.43。

遵循for语句的格式，完善下列程序代码，实现存款总额的计算。

```

money=_____ #本金50000元
rate=_____ #利率列表
for _____
    _____ #计算每年存款总额
print("5年以后存款总额：",money,"元") #输出结果

```

很多银行人民币理财产品的收益高于同期定期存款，但是高收益往往伴随高风险，因此很多普通投资者会选择收益相对稳定、风险较低的保证收益型理财产品。



任务二 计算理财产品收益

某投资者购买了10万元一年期收益率3.7%的银行保证收益型理财产品。每年理财赎回后，他会提取2万元用作生活所需，余下资金仍购买此种理财。在收益率不变的情况下，多少年后资金被全部取出？

理财一年到期赎回，投资者取出2万元。即 $money = \text{round}(money * (1 + 0.037), 2) - 20000$ 。重复执行该语句，就可以计算连续若干年后的资金余额。但投资多少年是未知的，即循环的次数无法确定，因此使用计数循环不太合适。这时可以考虑使用条件循环来解决该问题。

● 条件循环

条件循环一般用于循环次数未知的情况。只有当循环条件成立时，才执行循环体。在Python中，使用while语句来创建这类循环。while语句的基本格式如下。

```

while 关系表达式:
    语句或语句组

```

例如：

```

s=0
while s<=30:
    s=s+10          #循环体
    print(s)        #循环体

```

运行结果：

```

10
20
30
40

```

在本例中，循环体是两条语句，因此这两条语句相对于关键词while均要缩进。程序执行过程是：变量s初值为0，判断“s≤30”是否成立；如果成立则执行循环体（s值增加10，输出s的值），不断重复直到“s≤30”不成立，退出循环。

在此我们需要特别关注语句“s=s+10”。在编程时，形如“s=s+x”的语句可以起到累加的作用，如果x=1，即s=s+1可以起到计数的作用。一般在使用累加器或计数器时将变量赋初值为0。

也有一类特殊的关系表达式，如数字0表示假，非0数字表示真。如果循环是以“while 1:”开头，那么在不加干预的情况下，程序就会“不知疲倦”地一直执行下去。如果你想停止这个程序，按快捷键Ctrl+C即可。通常情况下，我们不会让程序陷入这种“死循环”的境地。

在计算新的理财金额时，“本金未被全部取出”可以作为循环的条件。遵循while语句的格式，完善下列程序代码，实现理财年数和金额的计算。

```
money=100000                                #本金100000元
year=_____                                #理财年数赋初值为0
while _____
    _____                                #计算新的理财金额
    _____                                #理财年数加1
print(year,"年后资金被全部取出")            #输出结果
```

● 调试程序

编写程序时难免会出现错误，这时可以借助Python的调试器pdb。例如在程序开头加入语句“import pdb”，然后，在适当的位置放一个“pdb.set_trace()”，就可以设置一个断点，让程序暂停运行，进入pdb调试环境。这时我们可以用命令“p”查看变量，用命令“c”继续运行，用命令“q”结束调试，退出程序。

在程序的第一句增加“import pdb”，while循环的语句组最后一句增加“pdb.set_trace()”，观察调试结果是否如下所示。

```
> d:\理财.py(4)<module>()
-> while money>=0:
(Pdb) p money,year
(83700.0, 1)
```

```
(Pdb) c
> d:\理财.py(4)<module>()
-> while money>=0:
(Pdb) p money,year
(66796.9, 2)
(Pdb) q
```



拓展练习

1. 在任务二中，投资人投入多少钱可以恰好在第10年取完最后一次2万元？
2. 某一年期理财产品实际收益率为4.25%，由于产品一直在持续运作，投资者不按时赎回视为自动再投资。在收益率不变的情况下，至少多少年后投资可以翻倍？



拓展知识

如果在一个循环中包含另一个循环，则称为循环嵌套。下面的双重循环可实现输出九九乘法表。

```
for i in range(1,10):
    for j in range(1,i+1):
        print("{0:1}*{1:1}={2:2} ".format(j,i,j*i),end="")
    print() #换行
```

上面的程序中通过 `format()` 函数来控制格式化的输出。其中，“{0:1}”表示第0个参数，即 `j` 输出长度为1个字符；“`end=""`”表示输出结束后不换行。

2.4 可以复用的代码

我们编写的程序代码中有许多功能是相似甚至完全相同的，这时可以考虑用代码复用的方法提高程序的效率。使用“函数”和“模块”是代码复用的常见方法。Python 包含了丰富的内置函数和模块，你也可以自己编写一些函数或模块，还可以到代码共享社区去下载。善于利用这些方法既可以减少重复编写代码的工作量，也可以使程序的结构更加清晰。



学习目标

- ★ 了解函数的作用。
- ★ 学会函数的定义和调用方法，并能熟练使用函数解决问题。
- ★ 学会发布共享代码的常见方法，并能熟练使用模块解决问题。

当我们面对一些复杂问题时，可以采取“自顶向下、逐步细化”的方法，即将一个完整的问题分成若干个小问题，每个小问题实现一个功能，再将每个小问题逐步细化、分解为一系列具体可实现的步骤。

本节我们将围绕“解决火柴棒摆数字问题”项目展开学习，通过该项目活动，体验复用和共享代码为编程带来的便利。

本项目主要包含“用自定义函数实现火柴棒摆数字问题”和“发布共享代码实现火柴棒摆数字问题”两个任务。



任务一 用自定义函数实现火柴棒摆数字问题

※ 活动1 用流程图表述问题解决的思路

有6根火柴棒，列出所有能摆出的自然数，要求火柴棒正好用完。火柴数字如图2.4.1所示。



图2.4.1 火柴数字

要找出这些数，可以从0到111依次判断这些数是不是恰巧需要6根火柴棒。为什么是0到111呢？观察图2.4.1可知，在0~9这十个数字中，摆出“1”只需要2根火柴棒，是需要火柴棒数最少的数字。因此6根火柴棒能摆放的最大数就是111。将自己的想法用流程图表示出来，如图2.4.2所示。

如何计算一个数需要多少根火柴棒呢？可以将这个数的每个数位上的数字所需要的火柴棒数加起来。实现方法是将这个数除以10取余，求得个位数；然后将这个个位数需要的火柴棒数累加；接着将这个数除以10取整，去掉个位数；重复以上步骤直到这个数为0。这种方法可行吗？请将其转化为流程图填入图2.4.3的框中。

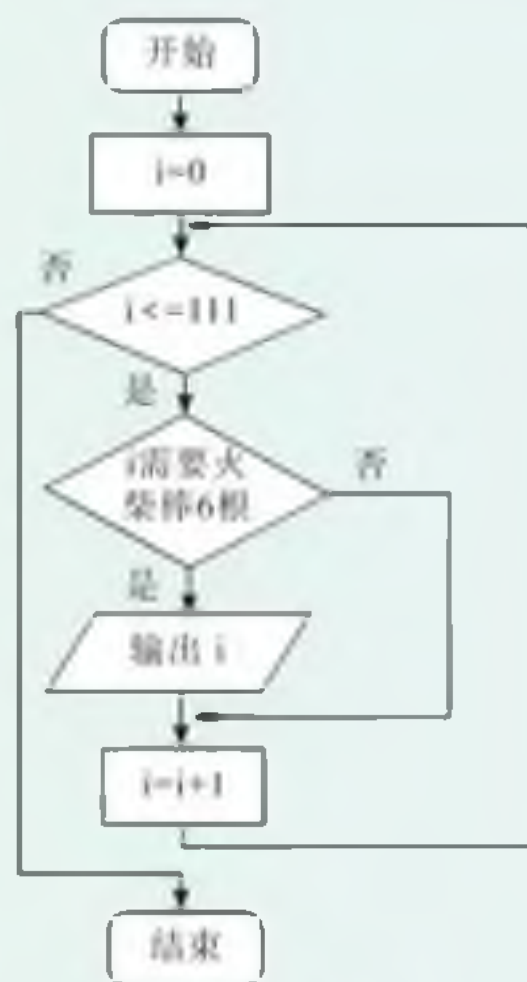


图2.4.2 流程图1



图2.4.3 流程图2

※ 活动2 完善程序实现问题解决

如果遇到一个需求多次出现，可以先思考，有没有内置函数可以提供帮助。如果没有，则可以考虑寻找或自己编写一个自定义函数。

“一个数字需要多少根火柴棒”的问题可以通过函数计算来实现。那么，如何编写函数呢？

● 函数

常见的系统函数有以下几种：

- 数学运算类函数：
- 数据转换类函数：
- 字符串操作类函数：
- 输入输出函数：
- 文件操作类函数：
- 其他类函数。

函数可分为系统函数和用户自定义函数两种类型。系统函数即函数库中的标准函数，是程序设计语言或操作系统提供给用户的一系列已经编制好的程序。在前面单元使用过的`print()`和`input()`函数都是系统函数。前者用于将字符串按指定格式输出到屏幕上；后者用于接收键盘的输入，并存储于相应变量中。用户自定义函数则是用户自己编写的一段程序。

函数一般包括函数名、参数、返回值和函数体等四部分。其中，函数名和函数体是必不可少的，参数和返回值可根据需要进行定义。定义函数的基本格式如下。

```
def 函数名(参数):
    #函数说明
    语句或语句组
    return 返回值
```

参数是用来向函数传递值的，当有多个参数时，各个参数之间用逗号分隔开。函数执行完成后，由`return`语句将表达式值返回给调用者，结束函数。函数的调用采用函数名(参数)的方式。

下面先来看一个简单的例子，这个函数用来求 $n!$ 。

```
def factorial(n):
    #求n!
    s=1
    for i in range(2,n+1):
        s=s*i
    return s
#调用factorial函数
total=factorial(4)
print(total)
```

上例中，程序执行时由于`def`块中的代码不是主程序的一部分，因此会跳过该段代码，从主程序的第一行`total = factorial(4)`开始运行。在遇到函数的调用语句`factorial(4)`时，转到函数中的第一行代码开始执行。函数运行时，变量`n`会被赋值为4。函数执行完成返回变量`s`的值即24，然后回到语句`total = factorial(4)`，将返回值赋给变量`total`，最后输出。

请完善下列程序，列出所有能摆出的自然数。

```
def match_num(num):
    #该自定义函数实现计算一个数字需要多少根火柴棒

#以下为主程序
snum=6                                #6根火柴棒
print("你可以拼出这些数字：")
for i in range(112):
    if match_num(i)==snum:            #如果i需要的火柴棒数等于现有火柴棒数
        print(i)
```



任务二 发布共享代码实现火柴棒摆数字问题

※ 活动1 发布共享代码

任何人编写的代码都有可能帮到其他人，那么如何共享这些有价值的成果呢？常见的方法是发布代码。请将“计算一个数字需要多少根火柴棒”的函数match_num转换为模块，然后发布。

● 发布代码

通过模块发布代码可以让很多人受益于你的工作。下面就以任务一为例介绍发布代码的方法。

首先做好发布前的准备工作。在D:\创建一个文件夹mymodule，其中包含mymodule.py和setup.py两个文件。

mymodule.py是一个模块文件，其内容就是match_num函数。

```
def match_num(num):
    f=[6,2,5,5,4,5,6,3,7,6]          #0~9分别需要多少根火柴棒
    if num==0:                        #将火柴棒总数变量赋初值
        total=f[0]
    else:
        total=0
    while (num>0):
        x=num % 10                   #取num除以10的余数，即num的个位数
        total=total+f[x]              #所需火柴棒数累加
        num=num//10                   #num整除10，即去掉num的个位数
    return total                      #返回需要多少根火柴棒数
```

setup.py文件包含有关发布的元数据，代码如下。

```
from distutils.core import setup
setup(
    name='mymodule',      #参数一定要与mymodule.py文件名相同
    version='1.0.0',      #版本号
    #参数一定要与mymodule.py文件名相同
    py_modules=['mymodule'],
    author='yd',           #作者
    author_email='yd@mail.com', #作者邮箱
    #代码功能描述
    description = '计算一个数字需要多少根火柴棒',
)
```

然后打开mymodule所在的目录，构建发布文件。最后将发布文件安装到你的Python本地副本中。安装完成后，mymodule文件夹包含了如图2.4.4所示的文件。如果你想让全世界都分享你的成果，你可以在互联网上发布你的代码。

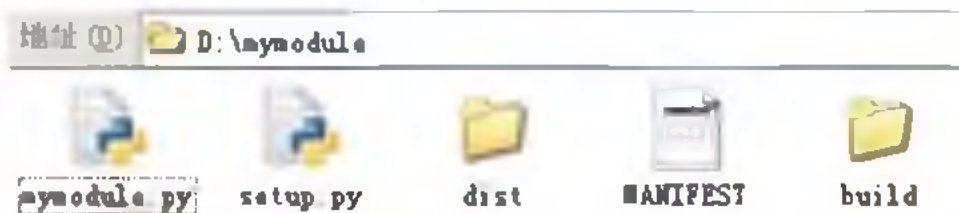


图2.4.4 mymodule文件夹内容

※ 活动2 导入并使用模块

在本机发布模块mymodule后，现在需要使用该模块来完成火柴棒摆数字的问题，应该如何调用呢？

● 导入模块并使用

已经构建发布的模块必须先导入程序才能使用。导入时使用语句“import mymodule”，调用模块内的函数时用“mymodule.match_num()”来实现。

完善下列程序，导入安装发布好的模块mymodule，实现火柴棒摆数字问题。

```
import _____ #导入模块
snum=6
print("你可以拼出这些数字：")
for i in range(112):
    if _____==snum: #调用模块内的match_num()函数
        print(i)
```



拓展练习

请编写程序，实现输入一个人的出生年份（ ≥ 1900 ），输出他的生肖，直到输入0结束，运行结果如图2.4.5所示。

```
请输入出生年份，输入0结束：2004
你的生肖是：猴
请输入出生年份，输入0结束：
```

图2.4.5 运行结果

完善下列代码，实现上述功能。

```
def shengxiao(year):
    #求生肖
    zodiac=["鼠","牛","虎","兔","龙","蛇","马","羊","猴","鸡",
            "狗","猪"]

#以下为主程序
myyear=int(input("请输入出生年份,输入0结束:"))
while myyear!=0:
    print("你的生肖是:",_____)
    myyear=int(input("请输入出生年份，输入0结束:"))
```



拓展知识

Python中提供了大量标准模块，亦称标准库，例如在“最强大脑”游戏的程序中使用的time模块和random模块。

利用time模块可以获取计算机的时钟信息，包括日期和时间。也可以利用它进行延时，参见下面的程序。

```
import time
print(time.time())           #时间戳，即1970纪元后经过的浮点秒数
time.sleep(2)                #延时2秒
print(time.asctime())        #格式化显示时间
```

利用random模块可以生成随机数，参见下面的程序。

```
import random
num=random.randint(0,1)      #产生一个[0, 1]之间的随机整数
print(num)
```

单元学习评价

1. 算法的三种基本结构是顺序结构、分支结构和循环结构，其基本思想来源于生活实践。请辨析以下生活事例所蕴含的结构思想，并尝试用流程图表达。

(1) 智能百叶窗可以根据室外光照强度自动调节叶片角度。控制程序根据采集数据（0~255）分为三级控制。数据为0~80时，室外光照弱，叶片角度调为 0° ；数据为81~175，叶片角度调整为 45° ；数据为176~255时，光照过强，叶片角度调整为 90° ，百叶窗呈关闭状态。

(2) 十进制正整数转换为二进制的方法是“除2取余，逆序排列”。即用十进制数整除2，得到商和余数，再用商整除2，以此类推，直到商等于0为止。然后把先得到的余数作为二进制数的低位，后得到的余数作为二进制数的高位，依次排列起来。

(3) “嫦娥四号”探测器经历了点火准备、主减速段、快速调整段、接近段、悬停段、避障段以及缓速下降段等动力下降过程，实现从距离月面15千米高度安全下降至月球表面的软着陆。

2. 请编程实现“猜数字”游戏：由计算机随机产生一个[1, 100]的整数，竞猜者输入猜测的数字，如果输入不正确则要求再次输入，直到输入正确时显示“你猜对了！”及猜测的次数。

单元学习总结

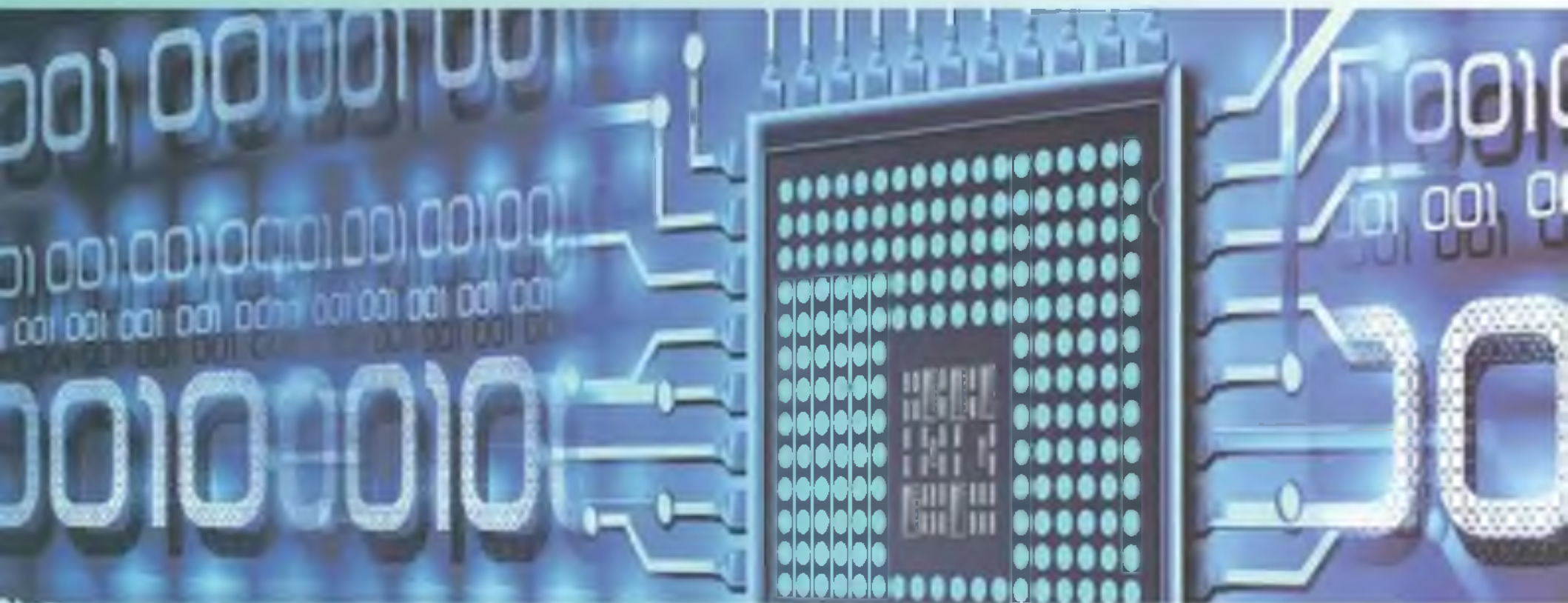


第 3 单元 认识数据

通过前两个单元的学习，我们已经对数据及计算有了初步的了解，知道了简单的编程计算的方法。在此基础上，我们还需要进一步理解：数据是无处不在的，我们可以通过数据来把握对象、描述问题并以此为基础解决问题。

本单元我们将进一步认识数据，了解数据编码的知识，学会分析数据间的关系；形成数据安全的意识，掌握设置密码保护数据的方法；根据解决问题的需要，选择合适的结构组织数据，让数据在系统中体现其价值。

对于信息时代的公民来说，学会发现并合理利用身边的数据解决实际问题，是应对生活、学习与工作的基本需要。



世界的本质是数据。

——维克托·迈尔-舍恩伯格（Viktor Mayer-Schönberger）

3.1 数据编码

在当今信息时代，计算机已经成为数据处理的主要工具。那么，数据是如何进入计算机中的？它们在计算机内部是如何存储的？

理解这些问题是将来进一步学习数据加工与处理的重要基础。本节我们将在解决问题的活动中，学习数据编码的基本知识，熟悉数据编码的思想和方法，认识数据编码的重要性。

编码是信息从一种形式或格式转换为另一种形式或格式的过程。这里的编码是指用预先规定的方法将文字、数字或其他对象编成可以存储在计算机里的数值。



学习目标

- ★ 了解各类数据采集的基本方法。
- ★ 能够解释文本、音频等数据的编码原理。
- ★ 理解数据编码的意义和作用。

智能公交系统是智慧城市的重要组成部分，它的应用提高了城市公交的运营效率，方便了人们的出行。在本节中，我们将围绕“智能公交系统中的数据采集”项目展开学习。通过该项目，我们可以进一步认识生活中存在不同类型的数据，了解数据采集和编码的一般方法，理解其基本原理，为应用数据解决问题打基础。

本项目主要包含“认识智能公交系统中的数据”和“查看存储在计算机中的数据”两个任务。

许多城市开通了智能公交，使用市民卡可以方便出行。在办理市民卡的过程中，有很多数据需要采集与编码。



任务一 认识智能公交系统中的数据

※ 活动1 办理市民卡

市民在办理市民卡时，一般要到指定地点办理录入信息、拍摄照片等手续。想一想，在办理市民卡的流程中，管理系统都收集了哪些

数据？生活中还有哪些数据可以采集以及如何采集？请尝试分析这些问题，并将相关内容填入表3.1.1。

表3.1.1 计算机处理的数据

表现形式	实例	采集这类数据的设备
文本	姓名、	键盘、
数字	年龄、	
图像		
声音	讲话的声音、	
视频	监控视频、	摄像机、

● 模拟信号与数字信号

按照取值特征，可以将信号分为模拟信号和数字信号。用传感器直接获得的信号一般为模拟信号，如声音、温度、压强等。模拟信号的值是随时间连续变化的，波形光滑。模拟信号的值称为模拟数据。以声音为例，通过话筒的声音传感器获得的声音信号取值情况如图3.1.1所示。

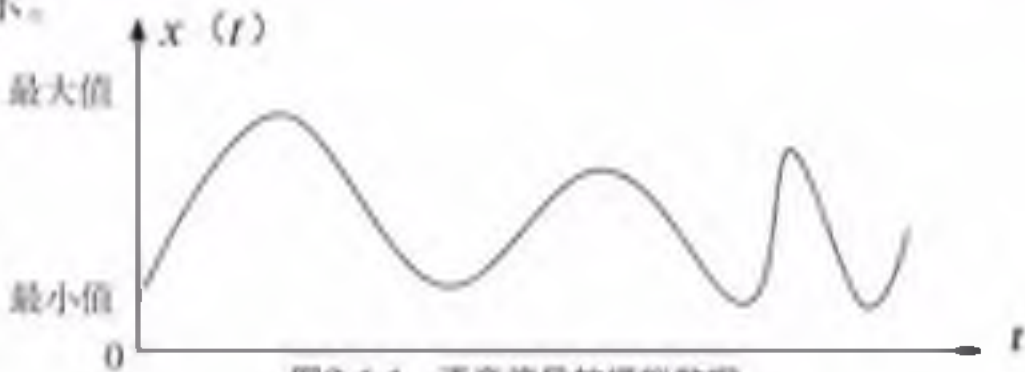


图3.1.1 语音信号的模拟数据

数字信号随时间的变化是非连续的，如图3.1.2所示。数字信号可以由模拟信号转换得来，数字信号的值即数字数据，直接用计算机所能理解的二进制表示，以方便计算机对其进行处理。

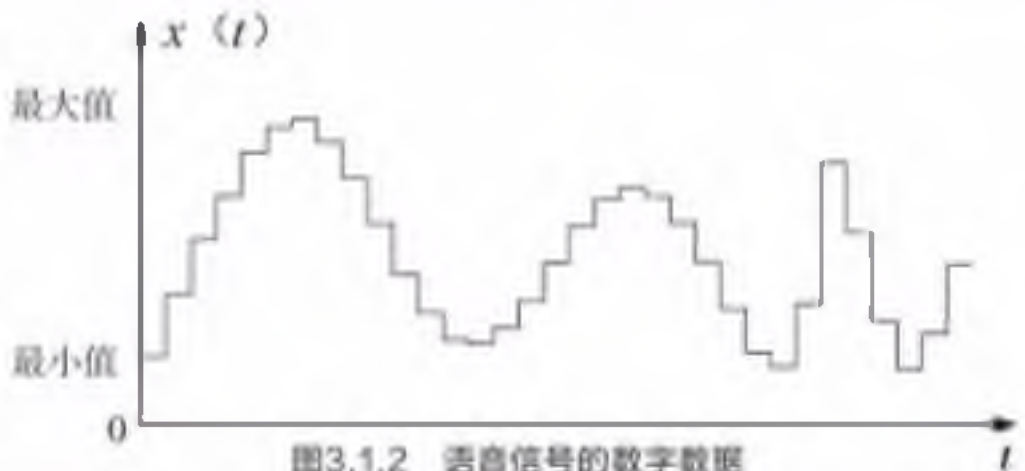


图3.1.2 语音信号的数字数据

※ 活动2 将语音模拟数据转换为数字数据

我们有时在公交车上能听到司机使用智能车载对讲系统与相关人

员通话。现在的智能车载对讲系统都采用了数字模式的对讲机。与传统模拟对讲机相比,数字模式可以更顺畅、更清晰地传输语音,这些语音数据是通过对模拟数据的处理得来的。

图3.1.3为待转换的模拟声音波形,其横轴为时间 t ,纵轴为声波的振幅。



图3.1.3 模拟声音信号

第一步,采样。在时间轴上按照一定的时间间隔取一系列时刻,得到每个时刻对应的声波信号的瞬时振幅值。如图3.1.4所示,在横轴上取 n 个时刻采集样本,纵坐标 x_n 即为每个样本所对应的声波振幅的离散值,当前 $n=10$ 。

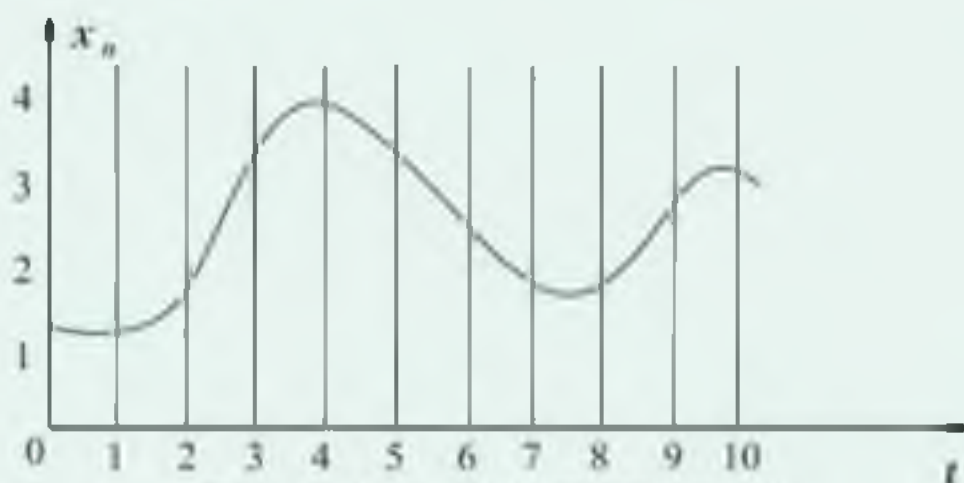


图3.1.4 按照一定的时间间隔进行采样

第二步,量化。将第一步所得的振幅瞬时值往最接近的整数取整,每个样本所对应的整数值得即为纵坐标,在图中标出这些整数值得,如图3.1.5所示。

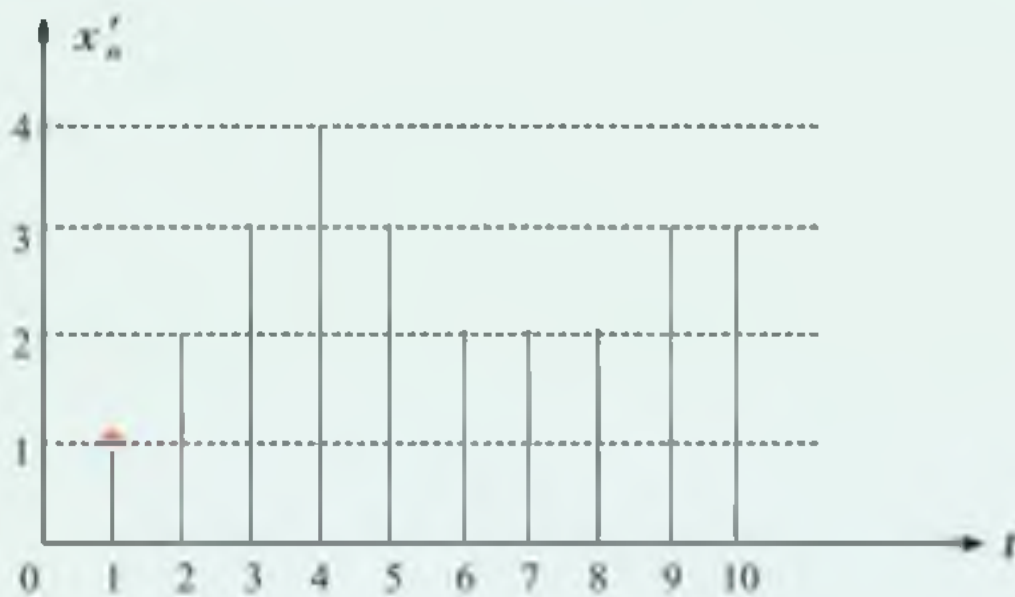


图3.1.5 量化后的声音信号

声波的振幅是反映波形从波峰到波谷压力变化的物理量。音量与振幅成正比。

在横轴上按一定的时间间隔从信号中取一个瞬时值,这个过程称为采样。录音设备在1秒内对声音信号的采样次数称为采样频率,常用的CD音质的采样频率是44.1kHz。在一定的时间内,采集的信号样本越多,对纵轴的刻度划分越细密,对信号波形的表示就越精确。

将信号波形的纵轴划分为若干区间,落到某一区间的采样点按四舍五入的规则取整数值,这个过程称为量化。

将样本值用计算机能够存储和处理的二进制数值表示的过程称为编码。

十进制与二进制数对应表	
十进制数	二进制数
1	001
2	010
3	011
4	100

第三步，编码。请参照边栏中的“十进制与二进制数对应表”将上一步标出的整数值用二进制数值表示，即可得到一串二进制数，将结果填入表3.1.2。我们可以将这些二进制数值理解为该声音信号的数字数据。

表3.1.2 数字化的声音信号

时刻	1	2	3	4	5
二进制数值	001				
时刻	6	7	8	9	10
二进制数值					

● 声音的数字化

我们不能把连续的音频信号存放在计算机中，必须将模拟声音数据转换为数字数据。将模拟声音数据转换为数字数据的这一过程，称为声音的数字化。通常先对模拟声音信号按给定的时间间隔进行采样，然后对采样数据进行量化，最后将量化后的数据用计算机能够存储和处理的二进制数值表示，即编码。

任务二 查看存储在计算机中的数据

※ 活动1 卡片组合游戏

无论是数字、文字，还是声音、视频等，编码后都以二进制的形式存储在计算机中。我们可以通过下面这个有趣的了解二进制。如图3.1.6（a）所示，桌面上摆放了5张卡片，将卡片正面的点数用十进制数表示，从右到左分别为1、2、4、8、16。现在翻动3张卡片，使其反面朝上，如图3.1.6（b）所示，这时卡片正面的点数和为1+8=9。若我们将所有卡片的正面读为1，将卡片的反面读为0，则得出十进制数9对应的二进制数就是01001。



图3.1.6（a） 标记点数的卡片



图3.1.6（b） 翻动3张卡片

● 二进制与数制转换

计算机外部的各种形式的数据经过编码后，都以二进制数的形式存储在计算机中。二进制数中只有0和1两个数字符号，其计数特点是“逢二进一”，即1+1=10。计算机中存储数据的最小单位是二进制位，用比

特 (bit) 表示, 1 比特能表示两种 (2^1) 状态 (即 0 或 1), 2 比特可以表示 00、01、10、11 四种 (2^2) 状态, 3 比特可表示八种 (2^3) 状态……。计算机存储数据的基本单位是字节 (Byte, 简称为 B), 8 比特构成 1 字节, 1 字节二进制数可表示 2^8 (256) 种状态。

1KB=1024B
1MB=1024KB
1GB=1024MB

在二进制数中, 每一个数字在不同的位置上具有不同的权值, 各位上的权值是基数 2 的若干次幂。比如二进制数 10010, 5 个位置上的权值从右往左可以表示为: 2^0 、 2^1 、 2^2 、 2^3 、 2^4 。将二进制数转换为十进制数时, 只要将该二进制数的每个位置上的数字与其对应位置的权值相乘, 然后求和即可。比如, $10010_{(2)} = 1 \times 2^4 + 0 \times 2^3 + 0 \times 2^2 + 1 \times 2^1 + 0 \times 2^0 = 18$ 。

把十进制非负整数转换为二进制数, 可使用短除法, 即“除二取余”法。例如, 把十进制整数 18 转换为二进制数的过程如图 3.1.7 所示, 结果为 $18_{(10)} = 10010_{(2)}$ 。

2	18	余数	
2	9	0	低位
2	4	1	
2	2	0	
2	1	1	
	0	1	高位

图 3.1.7 十进制整数转为二进制

由于二进制数位数较多, 书写不便, 故常改用十六进制数表示。我国古代曾经在重量单位上使用过十六进制, 如规定 16 两为 1 斤。十六进制需要使用 16 个不同的基本数字, 除了阿拉伯数字 0 到 9 外, 一般用字母 A 到 F (或 a~f) 依次表示后续数字 (相当于十进制数中的 10~15)。十六进制的基本计数规则是“逢十六进 1”, 即 $F+1=10$ 。

一般地, 十进制非负整数转换成 R 进制数 ($R \geq 2$) 采用“除 R 取余法”, 把 R 进制非负整数转换成十进制数采用“按权展开求和法”。

※ 活动2 查看数据的编码

人们在申请表里填写的姓名、性别和家庭住址等数据都是文本数据。请你尝试用 Python 语言实践下面的程序, 了解文本数据所对应的编码。

```
>>> ord('男')           #显示字符“男”的编码值
30007
>>> ord('1')           #字符“1”的编码值为49
49
>>> ord('A')           #字符“A”的编码值为65
65
```

Python 语言中的“ord”是将字符转换为对应存储编码的函数, 编码值以十进制数形式显示; “chr”是将编码值转换为对应字符的函数; “bin”是将一个十进制数转换为对应二进制数的函数。

Python 语言用“0b”作为前缀，表示一个二进制数值。

```
>>> bin(65)                #将十进制数65转换为二进制数
'0b1000001'
>>> chr(0b1000001)         #将编码值转换为对应的字符
'A'
```

● 文本数据的编码

文本数据是用来表示一定意思的一系列字符，包括字母、数字、标点符号、汉字等。

文本数据的编码通过对其中每一个字符进行编码实现。用于文本数据字符编码的方案有多种类型，其中ASCII码和Unicode是最为典型的两种编码方案。

1. ASCII码

ASCII码（American Standard Code for Information Interchange，美国信息交换标准码）用7位（标准ASCII码）或8位（扩展ASCII码）二进制数表示一个字符。标准ASCII码字符集中定义了大小写英文字母、标点符号和数字等字符和符号，共128个。计算机内部用一个字节来存放一个ASCII码字符，最高位用0表示。例如，“A”的ASCII码为01000001，即65。常用的ASCII码如表3.1.3所示。

表3.1.3 常用的ASCII码

ASCII码（十进制）	字符
0~31	控制字符或通信专用字符（不可见）
32	空格
48~57	数字0~9
65~90	大写英文字母A~Z
97~122	小写英文字母a~z
其他	英文标点符号、运算符号、括号等

2. Unicode码

Unicode是为了解决传统字符编码方案的局限性而产生的。例如，它可以用两个字节来表示一个符号，这样就有2¹⁶（65536）种不同的二进制编码，能表示很多国家的常用字。目前，Unicode还可以用更多字节来对字符进行编码。为了简化ASCII与Unicode之间的转换，Unicode的设计者还使其兼容ASCII码。原来用ASCII码能表示的字符，其对应的

Unicode 码只是在原来的ASCII码前加上8个0。比如，“a”的ASCII码是01100001，而它的Unicode码是0000000001100001。

计算机要处理汉字，必须对汉字进行编码，显然每个汉字至少需要两个字节。我国先后制定了多个汉字编码方案：（1）GB 2312—1980字符集共收录6763个汉字，其中一级汉字3755个、二级汉字3008个；同时收录了包括拉丁字母、希腊字母等在内的682个字符；（2）GBK字符集，兼容GB 2312—1980标准，收入21003个汉字、883个符号，共计21886个字符；（3）GB 18030—2000字符集，包含GBK字符集和CJK统一汉字扩充A的汉字，共计27533个汉字；（4）GB 18030—2005字符集，在GB 18030—2000的基础上，增加了CJK统一汉字扩充B的汉字及其他汉字，共计70244个汉字。

Unicode字符集，是全球可以共享的编码字符集，涵盖了世界上主要文字的字符，其中包括简繁体汉字，共计74686个汉字。如汉字“男”的Unicode编码为30007（十进制），用十六进制表示为7537，用二进制表示为0111010100110111。



拓展练习

1. 图3.1.8所示为一个条形码示例。请你利用网络搜索条形码数据编码的原理，并跟同学们分享你的收获。
2. 在实际生活中，是否还有其他“编码”的例子？请尝试列举生活中有关编码的例子。



6 943337 900015

图3.1.8 条形码示例图



拓展知识

除了文本数据、声音数据外，其他类型的数据也都有各自的编码方式。与声音数据类似，图像数据（如颜色）的数字化，也需要先进行采样，但不同的是，图像数据是在空间上变化，而不是在时间上变化。视频是图像（称为帧）在时间上的表示，一段视频就是一系列的帧连续播放而形成的，同时，视频往往也有伴音。由于视频常常数据量很大，往往要压缩存储，以便传播。

3.2 数据与结构

各种类型的数据被编码表示成二进制数据，存储到计算机中。在利用计算机解决问题的过程中，这些数据将是最基本的元素。但是，零散孤立的数据是很难被有效利用的。根据所要解决的问题的不同，我们还需要依据数据关系建立合适的结构。采用这些结构将数据组织起来，才能有利于操作和管理，进而更高效地解决实际问题。

本节我们将学习表、队列、树、图等数据结构，了解结构中数据间的关系，在一定的结构上完成算法设计；学会在生活中根据实际问题，建立合适的数据结构，进而运用所学的知识解决问题。



学习目标

- ★ 熟悉队列结构的概念和特点，能够使用 Python 语言对队列进行操作。
- ★ 了解树、图结构的基本概念及特点。
- ★ 能够比较不同数据结构的特点，会选用合适的数据结构组织数据解决简单问题。

数据经过采集和数字化后存储在计算机中，是为了便于应用和解决问题。本节我们将围绕“网络购物”项目展开学习，通过项目活动，认识相关数据的组织方法，了解数据之间的关系，理解几种典型的数据结构，为利用数据、实现数据的价值做准备。

本项目主要包含“探究网购订单处理”和“探究快递配送过程”两个任务。



任务一 探究网购订单处理

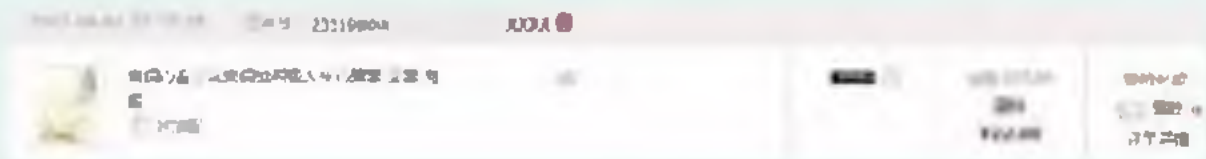
※ 活动1 了解订单数据

在网上购物时，在我们提交订单后，网页上就会显示订单数据。请

你参照图3.2.1所示的订单数据或者你自己的购物订单数据填写表3.2.1。



(a) A网站商品的订单数据



(b) B网站商品的订单数据

图3.2.1 网络订单

表3.2.1 网购中的订单数据

网站名称	订单中的数据	Python中对应的数据类型
	商品名称	字符串

利用计算机解决问题的过程，就是将问题中的已知数据输入计算机进行计算，然后输出结果数据的过程。比如，当我们利用网络购买商品时，计算机解决问题的过程就是对订单数据、商品数据等相关数据进行计算的过程。为了方便对数据进行处理，我们可以选择合适的软件工具，根据问题的需要为数据抽象出合适的数据类型，然后对数据进行组织和计算。

● 数据类型

数据类型用来定义一系列值及应用于这些值的一系列操作。比如，在Python语言中，有整数、浮点数、字符串、布尔等数据类型。整数类型的范围几乎仅受内存限制，能够进行加、减、乘、除等多种计算操作。

大多数程序设计语言都定义了两类数据类型：简单数据类型和复合数据类型。简单数据类型不能分解成更小的数据类型，复合数据类型则由简单数据类型或者复合数据类型组成。在Python语言中，整数、浮点数、字符串、布尔属于简单数据类型，列表、字典等属于复合数据类型。

订单数据中的商品名称可以抽象为字符串类型的数据，是一个基本

数据项。商品数量可以抽象为整数类型的数据，也是一个基本数据项。每个订单数据包括商品名称、单价、数量、金额、收货地址等基本数据项，所以订单数据需要抽象为复合数据类型。如图3.2.1(a)所示的订单数据用Python的列表存储，列表名称为OrderList。

```
OrderList=["2374761814130XXX","语文：生命的，文学的，美学的",34.66]
```

列表中前两个数据是字符串类型，最后一项是浮点数类型。我们还可以把很多订单数据排列在一起，形成订单表，用更复杂的列表存储。

※ 活动2 编制订单数据处理程序

网店接受了大量的订单，如何安排发货呢？实际上，网店在处理订单时，一般采取“先下单，先发货”的原则。因此，所有的订单将按照下单的时间顺序放进一个列表中，先放进去的先发货，所有订单排列在一起，像是一群人在排队。

下面的Python程序可以实现以下功能：提供“添加订单”“发货”“查看订单列表”“退出”四个操作选项。当我们选择“1”后输入订单数据，程序将订单数据添加到订单数据表中；选择“2”后，程序将当前订单列表中最先进入的数据删除（表示已安排发货处理）；选择“3”可以显示当前订单列表中所有的订单数据；选择“4”将结束运行。

请你完善下列Python程序，模拟添加订单和发货的过程，了解订单列表的操作过程。

```
listque=[]                #定义列表listque存储订单
x=0
while(x!=4):              #当x!=4时，执行循环
    print('1. 添加订单')
    print('2. 发货')
    print('3. 查看订单列表')
    print('4. 退出')
    x=int(input("输入你的选择:"))    #输入选择项
    if x==1:
        y=input("输入订单编号:")    #输入订单编号
        _____    #在列表listque 中添加订单号
    elif x==2:
        if len(listque)==0:
            print("订单列表为空")
        else:
            _____    #删除列表listque的首元素，表示发货
    elif x==3:
        print(listque)    #查询列表listque中的订单号
```

● 数据结构

数据结构是存在特定关系的数据元素的集合。在解决有些问题时，一些相关联的数据将集中在一起，形成一个数据的集合，这种集合能够单独或作为一个整体被访问和处理。

这里的
数据结构
也称为
逻辑结
构，主
要有集
合结构
、线性
结构、
树结构
和图结
构（又
称网状
结构）
四种
类型。

● 线性数据结构

线性数据结构又称为线性表。在线性数据结构中，除首元素没有前趋元素、尾元素没有后继元素外，其他元素都只有一个前趋元素和一个后继元素，如图3.2.2所示。线性表中数据元素之间是一一对应的关系。

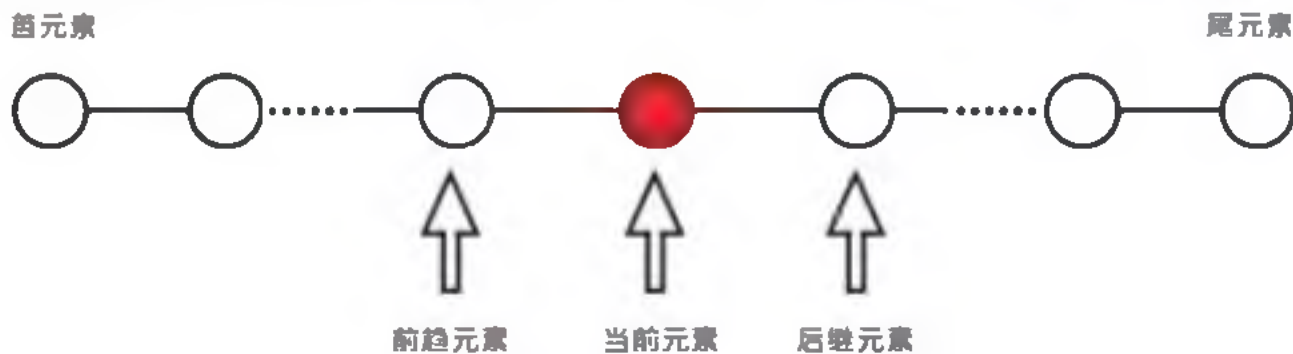


图3.2.2 线性数据结构

● 队列

队列是一种有限制的线性结构，它的数据元素只能在一端依次添加（进队），在另一端依次删除（出队）。典型的例子如超市里排队付款的队伍。

许多程序设计语言定义了复杂数据类型，以实现对数据结构更高层级的抽象。复杂数据类型可以封装并隐藏数据结构中的操作细节，让程序设计者更多地关注数据结构能做什么，便于利用数据结构解决问题。

Python中的列表数据类型，可以实现线性结构组织的数据元素的存储和操作。列表的使用者只需要知道列表上有哪些可用的操作，而不需要知道这些操作是如何进行的。

Python 中的
列表还
有 insert、
count、
len 等多
种操作
方法，
可以用
于实现
在中间
插入元
素、统
计元素
出现的
次数和
获取列
表的长
度等功
能。

比如在上述代码中，listque是列表类型的数据，存放了一组字符串类型的数据，表示订单编号。我们可以通过对应的方法对列表进行操作：pop(0)方法可以删除列表的首元素，append方法可以在列表尾部添加一个数据元素。利用列表，我们可以模拟队列中数据元素进队和出队的操作。



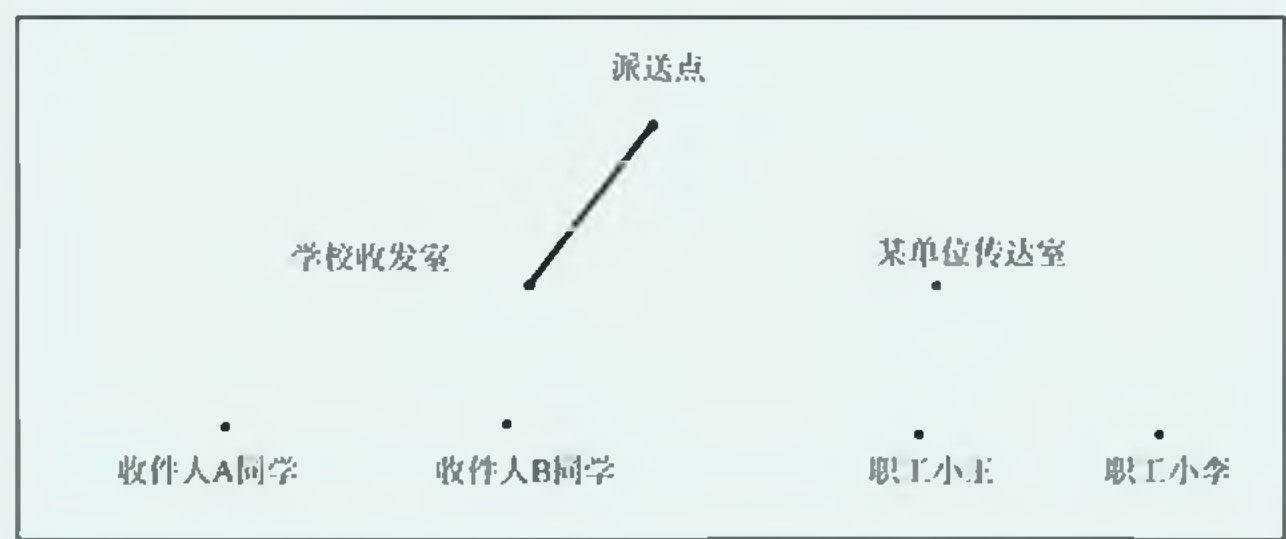
任务二 探究快递配送过程

※ 活动1 了解快递派送线路

每个快递员只负责固定的派送范围，他们从快件派送点领取快件后，

分别送往各自负责的快件领取点（比如小区门卫处、单位门卫处）或者具体用户。学校的快递由快递员送件上门后，收发室老师将快件按工作人员部门、学生班级分类摆放，由各班级指定专人取件。

现将派送点、学校收发室和收件人用点表示，派送的线路用线段表示，请你尝试在下面框中画出多个快件从派送点到不同收件人所经过的线路。



● 树结构

树结构是一种具有层次关系的非线性结构。树是由 n ($n \geq 0$) 个节点组成的有限集合。若 $n = 0$ ，则称为空树。任何一个非空树均满足以下两个条件：（1）仅有一个称为根的节点；（2）当 $n > 0$ 时，其余节点可分为 m ($m \geq 0$) 个互不相交的有限集合，其中每个集合又是一棵树，并称为根的子树。在图3.2.3中，节点A为根节点，B、C、D为A的子树的根节点。同理，E、F、G是B的子树的根节点，B是E、F、G的父节点。在树结构中，数据元素之间是一对多的关系。

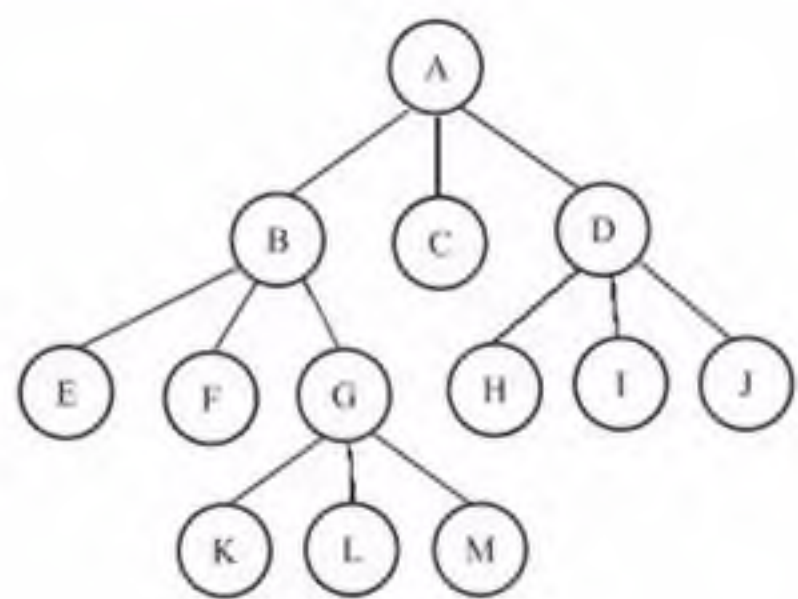


图3.2.3 树结构

快递到达目的地城市后，物流图的结构呈树状，如图3.2.4所示。



图3.2.4 城市物流配送线路

※ 活动2 了解物流网络

由于需要综合考虑运营成本，商品在城市间运输的路线是需要计算和规划的。请你查看图3.2.5中的物流过程，尝试用圆圈表示城市，用线段表示城市之间的送达关系，将图3.2.6补充完整，了解商品配送的路线特点。

包裹正在等待揽收

湖南省岳阳市汨罗市公司 已收件

[岳阳市]A快递湖南省岳阳市汨罗市收件员已揽件

[岳阳市]湖南省岳阳市汨罗市 已发出

[长沙市]快件已到达 长沙转运中心

长沙转运中心公司 已打包

[长沙市]长沙转运中心 已发出

[南通市]快件已到达 南通转运中心

[南通市]快件已到达 南通转运中心

[南通市]南通转运中心 已发出

[泰州市]快件已到达 泰州转运中心

[泰州市]泰州转运中心 已发出

[扬州市]江苏省扬州市 已发出

江苏省扬州市邗江区泰安镇 已收入

包裹正在等待揽收

[岳阳市]B快递 湖南汨罗市公司派件员 已揽件

[长沙市]快件已到达 湖南长沙转运中心

[长沙市]湖南长沙转运中心 已发出

[南京市]快件已到达 江苏南京转运中心

[扬州市]快件已到达 江苏扬州转运中心

[扬州市]江苏扬州转运中心 已发出

[扬州市]江苏主城区公司扬州广陵区湾头镇服务部派件员：因区代理点XXXXXXXX正在为您派件

[扬州市]在江苏主城区公司扬州广陵区湾头镇服务部进行签收扫描，快件已被 签收，感谢使用B快递，期待再次为您服务

[扬州市]江苏省扬州市 已发出江苏省扬州市邗江区泰安镇派件员：XXX正在为您派件

[扬州市]江苏省扬州市 已发出江苏省扬州市邗江区泰安镇公司 已签收

签收人：XX，感谢使用A快递，期待再次为您服务。

(a)

[扬州市]您的快递已由扬州声谷园区点XX驿站代收，请凭取货码及时取件（查询方式：短信/物流详情页）

[XX驿站]您已在扬州声谷园区点完成取件，感谢使用XX驿站，期待再次为您服务。

(b)

图3.2.5 网购订单物流信息



图3.2.6 物流数据图形化示意图

● 图结构

图结构是由一组节点（称为顶点）和一组节点间的连线（称为边或弧）构成的一种数据结构。图结构中的每个顶点都可以与其他顶点有边相连，图结构中数据元素之间是多对多的关系。

图3.2.6表示的是商品从供货点到收货点的派送过程的图结构。图3.2.7也是一个图结构，其中，标为“1”的顶点与两条边相连，顶点“4”与“2”“8”“9”相连。

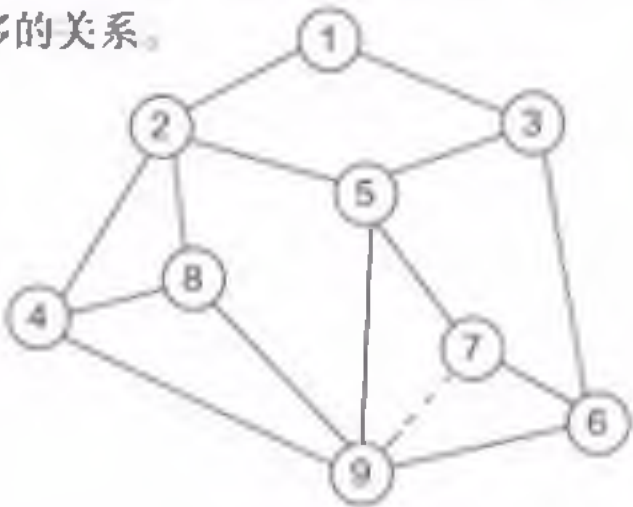


图3.2.7 图结构

在物流网络中，分拨中心、配送中心、货物需求点等可以抽象为图的顶点，城市道路、各级铁路等可以抽象为图的边，如城市以及城市之间的运输道路就是图结构。利用图结构，我们还可以解决物流中的许多问题，如道路网络分析、车辆运营安排等。

※ 活动3 规划取快递最快路线

某同学网购的书已经到达家附近的快递门店，需要他自己去取。不巧的是，这次购买的三本书是三个不同的物流公司派送的，他家与

各快递门店的位置如图3.2.8所示。



图3.2.8 快递门店位置分布图

该同学估算了在这些地点之间步行需要的时间，详见表3.2.2。

表3.2.2 该同学家及快递门店间步行所需时间表

地点—地点	时间/分
家—快递门店A	2
家—快递门店B	5
家—快递门店C	10
快递门店A—快递门店B	4
快递门店A—快递门店C	6
快递门店B—快递门店C	4

请你帮他规划最省时的路线，然后设计算法解决问题并在下框中描述你的算法。

我们可以将该同学家和各个快递门店的位置抽象成顶点，两个位置间的步行线路抽象为边，边上的值表示步行时间，如图3.2.9所示。利用

在图3.2.9中，每条边上的数值表示两个位置间的步行时间，称为边的“权”，这样的图称为“加权图”。

该图我们可以更清晰地知道数据间的关系，从而解决问题。

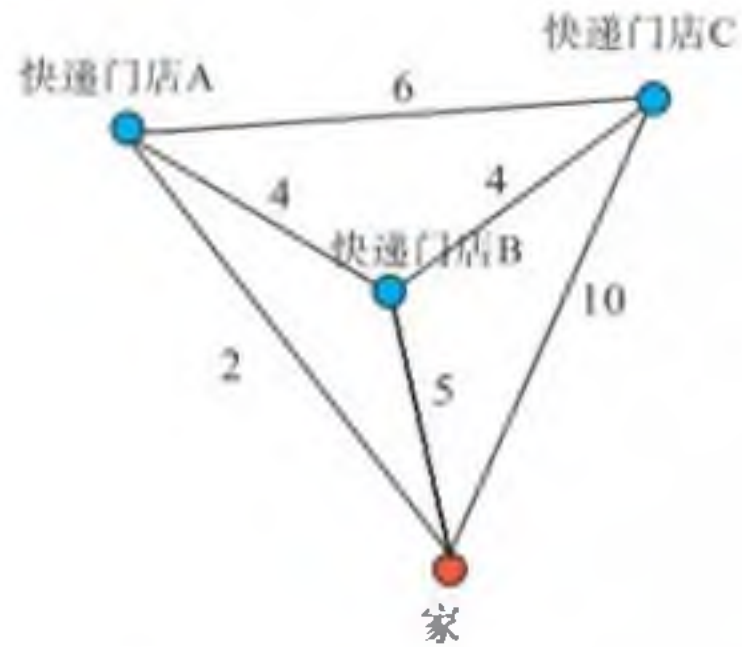


图3.2.9 该同学家及附近快递门店之间步行所需时间数据（单位：分）

从起点出发，把当前可以到达的下一个位置列举出来，再从列举出的新位置出发，继续列举下一步可以到达的位置，以此类推，直到返回起点。我们把所有可能的做法用图形描述，如图3.2.10所示，图下方圆圈中的数值是该走法的总用时。我们发现，分析过程的图形是树结构，树中的节点表示当前所在的位置，边表示选择的线路。利用树结构，我们能够更清晰地实现不重复、不遗漏地列举所有做法，更利于通过比较得到最优解。

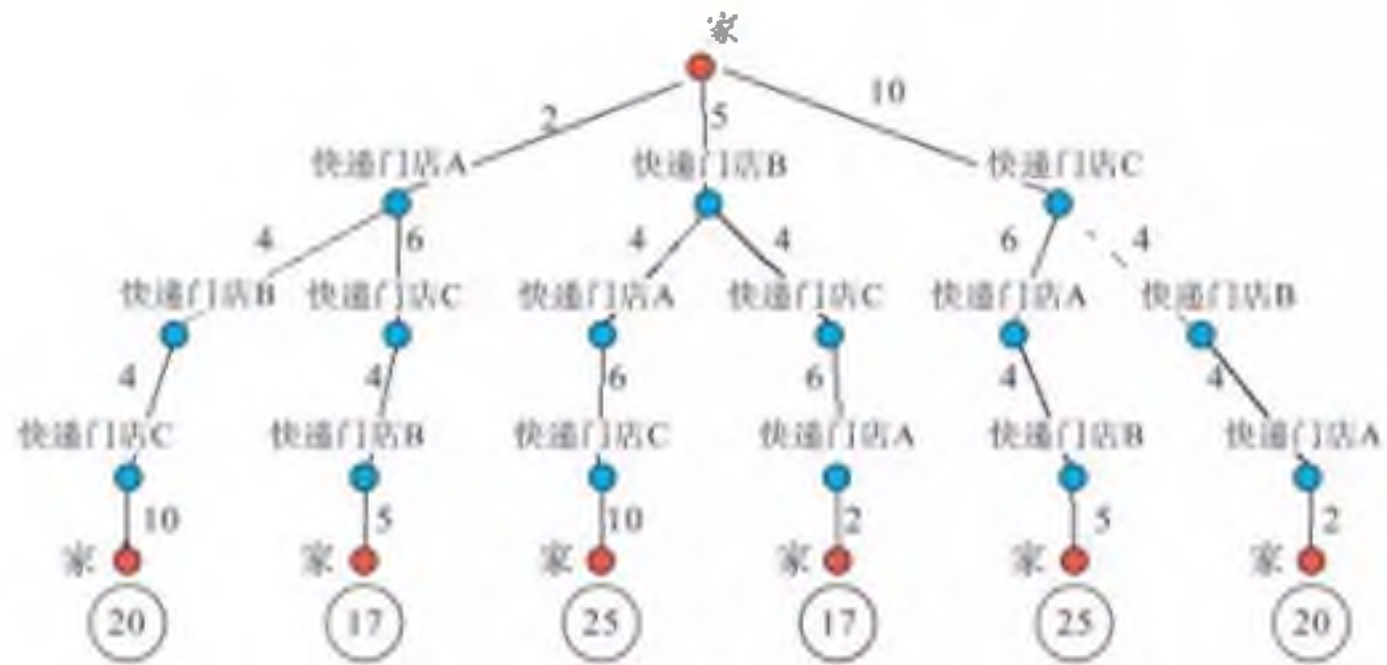


图3.2.10 求解最短用时分析树（单位：分）



拓展练习

请分析队列、树、图三种结构的区别，并将结果填在表3.2.3中。

表3.2.3 数据结构的比较

结构类型	数据（节点）之间的关系	生活中相应结构应用举例
队列 (线性)		
树		
图		



拓展知识

数据结构可分为逻辑结构和物理结构两种类型。本节学习的结构属于逻辑结构，强调数据与数据之间的逻辑关系。在逻辑关系中，除了线性结构、树结构和图结构之外，还有集合结构，指结构中的元素之间仅“属于同一集合”，没有其他关系。逻辑结构主要是应解决问题的需要而建立的数据组织形式。物理结构指数据在计算机中的表示形式，即数据的逻辑结构在计算机存储空间的存放形式，又称为存储结构。数据的存储结构有顺序存储结构、链式存储结构、索引存储结构和散列存储结构等。

3.3 数据与系统

在人们的日常生活、学习和工作中，大多数问题是包含多种数据间关系的复杂问题。要解决这些复杂问题，就需要把数据存储在系统中，以使其发挥更大的作用。在本节中，我们将通过分析数据库系统的功能，研究系统中的数据、数据的属性、数据间的关系以及数据的结构，理解数据与系统的关系，并通过电子表格、Access和Python等软件管理和分析系统中的数据，以更好地应用数据。



学习目标

- ★ 了解文件和数据库文件的概念。
- ★ 了解 GUI 的基本概念，学会设计简单的 GUI。
- ★ 学会根据任务需求，选用恰当的软件处理数据。
- ★ 体验使用不同工具管理数据的过程，简单了解数据库管理系统和数据库系统的基本概念。

利用数据结构组织起来的数据必须在系统中才能“活”起来。本节我们将围绕“旅行小助手系统”项目展开学习，通过项目活动，认识数据与系统的关系，做到根据系统功能选择所需数据，让系统充分发挥数据的潜能。

本项目主要包含“探究旅行线路”“走近火车票订票系统”“探究订票系统功能”和“设计‘旅行小助手系统’”四个任务。

某同学经常和亲友们一起旅游。结合信息技术课程的学习，他想自己开发一个旅行小助手系统，用它来解决自己遇到的旅行线路管理、交通乘车指导、以往旅游相关记录管理等各种问题。我们将通过下面的活动了解这个系统的构成并熟悉系统，进而尝试帮助他全面地设计、规划这个系统。



任务一 探究旅行线路

※ 活动1 初看旅行线路

“旅行线路”是该同学的旅行小助手系统中的一项功能，可以记录计划出游的旅行线路。他希望在这个系统中能够添加新的线路，并可以查询已经输入系统的线路。

下面的Python程序代码可以生成“旅行线路”程序界面，我们一起来帮他设计界面吧。请修改程序，使得程序运行后，可以在“添加线路”按钮下方添加“查询线路”按钮。

```
from tkinter import *          #导入tkinter模块
root = Tk()                   #创建一个窗口
root.title("旅行线路")        #设置窗口标题
root.geometry('300x120')      #设置窗口大小
root.resizable(0,0)           #禁止调整窗口大小
#在窗口上建一个“添加线路”按钮
Button(root, text="添加线路", relief="solid",width=10,height=2).
pack()
#下面的代码实现在窗口上添加一个“查询线路”按钮的功能
```

● 图形用户界面

图形用户界面（Graphical User Interface，GUI）又称图形用户接口，是指采用图形方式显示的计算机操作用户界面。在Python语言中，tkinter是实现图形用户界面的模块，可以快速创建GUI应用程序。我们需要先导入模块，上面所示的代码用“from tkinter import *”导入tkinter模块，然后就可以运用相应代码创建矩形容器（Frame）、文本框（Entry）、命令按钮（Button）、文本标签（Label）等交互对象。

在现代程序设计中，我们用“对象”建立计算机世界的实体与客观世界的实体相一致的映射关系，问题的解决方案可以用一些彼此交互的对象来形象化表示。比如在“旅行线路”例子中，我们利用Python提供的按钮类创建按钮对象，通过单击按钮的交互方式完成代码的执行过程。

在面向过程的程序设计中，我们关注的是各种类型的数据及其操作；而在面向对象的程序设计中，我们将数据及其操作隐藏细节后封装为类，函数调用式的计算演变为对象之间的交互。

※ 活动2 更新旅行线路

该同学对一条新的旅行线路感兴趣，希望将它添加到系统中。线路情况如下：

桂林，5日游，桂林市区—龙脊梯田—漓江漂流—骑行之旅—遇龙河漂流，人均1200元。

在教科书配套资源中找到并打开该同学已收集的旅行线路文件“旅行线路.txt”，了解文件内容。

配套资源中的“添加线路.py”程序用来实现将数据添加到“旅行线路.txt”文件中的功能，你能否按照提示运行该程序并把新的旅行线路情况添加到文件中呢？完成后，再次打开“旅行线路.txt”，查看文件内容。对照程序想一想数据存储需要通过哪些代码来实现。

Python代码：

```
from tkinter import *
root = Tk()                                #创建一个窗口
root.title("添加线路")                    #设置窗口标题
root.geometry('600x100')                  #设置窗口大小
root.resizable(0,0)                        #禁止调整窗口大小
var=StringVar()                            #定义StringVar()类型

def intomap():                             #Button按钮激发函数
    c=open("旅行线路.txt",'a+')             #以追加模式打开文件
    #在txt文件末尾添加text里的内容
    c.write(var.get()+"\n")
    c.close                                #关闭文件
#在窗口上建一个文本标签
Label(root, text='请输入线路', font=('Arial', 10)).pack()
#在窗口上建一个文本框
Entry(root, textvariable=var, width=550).pack()
#在窗口上建一个Button按钮
Button(root, text="添加线路", command=intomap, relief="solid",
width=10).pack()
root.mainloop()                            #进入事件（消息）循环
```

 intomap 是一个按钮激发函数。Python 程序运行后，当我们单击按钮时，计算机便转入执行函数中的代码。

● 文件

在解决问题的过程中，我们可以根据需要将数据组织在一起形成队列、树或图等逻辑结构。这时的数据一般存储在内存中，当退出程序时，相关数据也将丢失。如果需要更持久地存储数据，我们可以将数据

以文件的形式存储到外存储设备中。在活动2中，“线路数据”是用文件的形式存储在计算机的磁盘里的。

文件是存储在外存储设备中的相关数据的集合。

● Python读写文本文件

在Python语言中，用open函数把外存储器里的文本文件（.txt）打开，用write方法将内存中的数据存储到外存储器的文件中，文件操作结束时一定要用close方法关闭文件。

Python还提供了读写电子文档和电子表格等多种类型文件的模块，可以很方便地建立起不同计算工具间的联系。

在数据管理技术的发展历程中，文件的使用有助于长期保存数据，并且实现由专门的软件对数据进行管理，从而减轻人工管理数据的负担。

数据的存储传统上是使用单独的没有关联的文件。比如在活动2中，程序“添加线路.py”使用文件“旅行线路.txt”。现在我们也可以将一些文件建立关联，形成数据库文件。

计算机的存储设备按用途可分为主存储器（内存）和辅助存储器（外存）两类。内存指主板上的存储部件，用来存放当前正在执行的程序及数据，程序关闭后，数据消失。外存通常是磁性介质或光盘等，能长期保存信息。

该同学想乘火车从厦门去往成都，他登录12306网站准备购买车票。

任务二 走近火车票订票系统

※ 活动1 查找车次

该同学在网站上查找车次，发现厦门并没有直达成都的火车，而网站提供了多种接续换乘方案，可以任选一个中转站，比如武汉市（包括武昌、汉口等多个火车站），自己探寻具体的换乘信息。该同学在网站上查到了一些信息，如图3.3.1和图3.3.2所示。由于需要在页面间来回切换，他想利用电子表格软件将有用的信息整理到一个表中，以方便选择换乘方案。

中国铁路客户服务中心网站（12306网站）是铁路服务客户的重要窗口，客户通过登录该网站，可以查询列车时刻表、票价、列车正晚点、车票余票等信息并办理购票等业务。

车次	出发站 到达站	出发时间 到达时间	历时	商务座 特等座	一等座	二等座	商务座 软卧	一等座 硬卧	二等座 硬卧	硬座	站席	无座
K904	厦门 武昌	06:44 22:15	15:31 当日到达	-	-	-	-	-	-	-	-	-
							¥409.0	¥281.5	¥152.0	¥152.0		
Z126	厦门 武昌	07:06 18:15	09:09 当日到达	-	-	-	-	-	-	-	-	-
							¥372.0	¥238.0	¥138.0	¥138.0		
G2046	厦门北 武昌	07:07 15:03	07:56 当日到达	9	4	0	-	-	-	-	-	-
				¥1804.0	¥978.0	¥588.0						

图3.3.1 厦门至武汉部分车次信息

车次	出发站 到达站	出发时间 到达时间	历时	商务座 一等座	二等座	硬座	软座	硬座	软座	硬座	软座	正座	其他
K532	武昌 成都东	01:05 21:18	20:13 今日到站										
								¥475.5	¥300.5	¥172.5	¥172.5		
T128	武昌 成都	01:20 17:50	16:30 明日到站										
								¥627.5	¥271.5	¥183.5	¥183.5		
K392	武昌 成都	06:02 04:39	22:37 次日到站										
								¥464.5	¥208.5	¥173.5	¥173.5		

图3.3.2 武汉至成都部分车次信息

包含车次相关数据的WPS表格如图3.3.3所示。请你完成所需数据的采集和存储，然后提出一些车次选择的参考意见。为方便比较和选择，将“历时”的计量单位统一换算成分，票价选取该车次的最低列车席位票价。

	A	B	C	D	E	F	G
1	车次编号	出发站	到达站	出发时间	到达时间	历时	最低票价
2	K904	厦门	武昌	6:44	22:15	931	¥152.5
3							
4							
5							
6							
7							

图3.3.3 用WPS表格存储的车次相关数据

※ 活动2 整理购票信息

用WPS表格整理数据有时还是不太方便，我们尝试用数据库管理软件Access管理数据。

● 数据库

数据库文件中数据之间的关联有多种模型，如层次模型、网状模型和关系模型。其中，基于二维表的关系模型是最常用的。

在日常生活中，我们通常用电子表格软件来管理数据。但电子表格软件中的表是有行数限制的，而且在多人共享和联表查询方面也存在不足。因此当数据量比较大，且管理比较复杂时便需要采用更科学的数据管理方法。在活动中，我们发现12306网站的各种服务功能背后需要有大量数据，这些数据聚集成“库”，即所谓“数据库”。数据库是以一定的组织方式存储在计算机中的相互关联的数据集合。

该同学用Access软件为火车换乘数据建立了数据库，如图3.3.4所示。在数据库中，包含“厦门至武汉车次”和“武汉至成都车次”两个表。

车次编号	出发站	到达站	出发时间	到达时间	里程	票价
02101	汉口	成都	17:39	11:53	424	2316.5
02102	成都	汉口	17:44	22:01	424	2316.5

图3.3.4 Access管理数据库界面

请你在教科书配套资源中找到并打开“厦门至成都.mdb”数据库文件，将“厦门至武汉车次”和“武汉至成都车次”这两个表的数据补充完整，所需数据可以通过12306网站查询。

● 数据表

在数据库中，数据是以二维表的形式组织存储的，称为表。以如图3.3.4所示的“武汉至成都车次”表为例，表中的一列称为一个字段，一个字段对应车次实体的一个属性，每个字段都有一个名字，称为字段名，如车次编号、出发站等。表中的一行称为一条记录，一条记录对应于一个车次实体，保存有关实体属性的数据。可以唯一确定一条记录的字段称为主键，如车次编号。若干个车次数据的集合就组成了“武汉至成都车次”表。我们可以对数据表进行插入、删除、更新等操作。

需要指出的是，数据库中的表与表之间也可以是相互关联的。为不同表中的两个字段建立关联后，两表中的记录就可以通过这个关系联系在一起。比如，在图3.3.4所示的两个表中，换乘站既是前段车程的到达站，又是下段车程的出发站，可以通过这一点建立关系。

● 数据库管理系统

在数据库技术中，数据是如何被科学地组织和存储的？又是如何被高效地获取和维护的？完成这些任务的是一个系统软件——数据库管理系统。

数据库管理系统是定义、创建、维护数据库的一种工具。它允许用户管理和控制数据库中的数据。常见的数据库管理系统有Sybase、ORACLE、MySQL、Access等。



任务三 探究订票系统功能

※ 活动1 查询换乘方案

建立了数据库之后，为了能更方便地查询到换乘方案，可尝试在Access中建立表之间的“关系”，利用“关系”解决问题。

在查找换乘方案时，“厦门至武汉车次”表中的字段“到达站”与“武汉至成都车次”表中的“出发站”是有对应关系的。请你打开教科书配套资源中的“厦门至成都.mdb”文件，利用“关系”菜单创建如图3.3.5所示的表间关系，并查询可能的换乘方案。

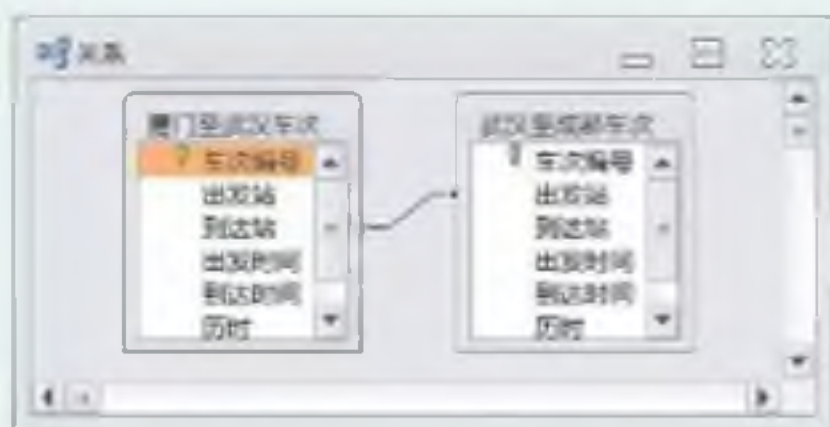


图3.3.5 表间关系

关系就像绳子将两个“表”联结在一起，我们可以利用“查询”面板中的“查询设计”选项按照提示创建选择查询，得到查询结果。但查询结果并不一定理想，比如没有考虑换乘时间的合理性等。武汉三个火车站之间利用地铁换乘，大约用时半小时。为此，修改查询表达式为：

```
SELECT 厦门至武汉车次.*, 武汉至成都车次.*
```

```
FROM 厦门至武汉车次 INNER JOIN 武汉至成都车次 ON 厦门至武汉车次.到达时间+0.05<武汉至成都车次.出发时间
```

式中0.05的单位为天，即 $0.05 \times 24 \text{小时} = 1.2 \text{小时}$ ，预留了武汉三站之间的换乘时间及在火车站的出站、进站时间。

执行查询，结果如图3.3.6所示。在图中，“厦武”为“厦门至武汉”的简称，“武成”为“武汉至成都”的简称。这个结果可以作为换乘方案。

厦武车次	厦武出发时间	厦武到达站	厦武到达时间	武成车次	武成出发站	武成出发时间
D3274	8:17	武汉	18:12	Z48	汉口	22:20
D3288	10:28	武汉	17:55	K422	武昌	23:12
D3288	10:28	武汉	17:55	T246	武昌	19:25

图3.3.6 查询结果

※ 活动2 满足个性化需求

在12306网站中有出发时间段和车型等选择项，可以满足不同旅客的出行要求。同样，也可以在Access中设置查询条件，解决个性化需求的问题。在查找换乘方案时，可以在“查询设计器”窗口的“条件”栏中设置查询的准则。

● 查询准则

如果只是简单地查找某个字段为特定值的记录，只要在该字段对应的“条件”栏中输入该值即可；如果不仅是查找某个特定值，在准则中可以使用表达式。在表达式中可通过操作符设置查询范围。



主要的操作符包括 AND、OR、BETWEEN AND、LIKE 等。

某同学想乘坐7:00~8:00的高铁或动车出行。为此，需在查询表达式中追加条件：

WHERE (厦门至武汉车次.车次编号 LIKE "G%" OR 厦门至武汉车次.车次编号 LIKE "D*") AND (厦门至武汉车次.出发时间 BETWEEN #7:00# AND #8:00#);

查询结果如图3.3.7所示。

车次编号	出发时间	出发站	到达时间	车次	出发站	到达时间
D2232	7:39	武汉	14:53	K422	武昌	23:12
D2232	7:39	武汉	14:53	T246	武昌	19:25
D2232	7:39	武汉	14:53	Z49	汉口	22:20

图3.3.7 体验个性化出行

数据库的建立、使用和维护等工作仅靠数据库管理系统还远远不够，还要有专门的人员来完成，这些人被称为数据库管理员。数据库管理员通过数据库管理系统可以对数据库进行操作和维护。但对于广大普通用户来说，很难像专业的数据库管理员那样操作数据库，而具有良好交互性的数据库应用系统可以满足人们简单、方便地使用数据的需求。

● 数据库系统

数据库系统是由数据库、数据库管理系统（及其应用开发工具）、数据库应用系统、数据库管理员和用户组成的存储、管理、处理和维护数据的系统。数据库主要用于存储数据，需要足够大的内存储器 and 外存储器等硬件平台的支持。软件部分除了操作系统和数据库管理系统外，还包括支持特定应用环境开发的软件工具以及开发完成的数据库应用系统。12306网站是一个大型数据库应用系统，其他如各类财务管理系统、人事管理系统、图书管理系统等软件也都是数据库应用系统。



任务四 设计“旅行小助手系统”

※ 活动1 设计系统功能和应用程序界面

该同学想通过开发“旅行小助手系统”记录以往出行的信息，以及个人的旅游经历、爱好等，作为今后安排旅行时的参考。

请你帮他设想一下，这个旅行小助手系统应该具备哪些功能？请在小组内展开讨论，并用思维导图的形式将你们设计的系统功能表示出来。



为使用户可以便捷地调用系统功能，还需要设计一个友好的界面。请你跟他一起来设计，并将结果填在表3.3.1中。

表3.3.1 系统界面首页中的对象及对应的功能

对象类别	对应的功能
“成员管理”按钮	单击按钮可以打开成员管理界面

设计数据库应用系统时，我们一般自顶向下进行。首先设计总体结构，然后再逐层深入，直至进行每一个模块的设计。我们需要先大体地勾画出系统以及每个模块大致的计算机处理流程，将头脑中构想的处理过程描述下来。比如，首先单击“成员管理”按钮，然后出现“成员管理”子窗口，利用该子窗口可以添加成员信息、修改成员信息和删除成员信息……“旅行小助手系统”还可能包含“旅行记录管理”“旅行线路管理”等模块。根据处理过程，可以设计系统功能和相关界面。

※ 活动2 设计系统数据库，了解Python访问Access数据库的步骤

该同学的“旅行小助手系统”界面设计已经完成，但系统功能的实现需要相关数据的支撑。他想利用系统制订一个乘坐飞机去一座大家都没有游玩过的城市旅行的计划，这就需要旅客姓名、以往旅行的城市名称等数据。因此，他创建了“旅行小助手”数据库，如图3.3.8所示。他希望该系统能把所有成员的交通工具要求和以往的旅行城市名称都显示出来。请你打开教科书配套资源中的“旅行小助手.mdb”文件，分析表内数据和表间数据的关系，看看创建的数据库是否合理，并且根据小组设计的系统功能修改数据库，以使其更合理。

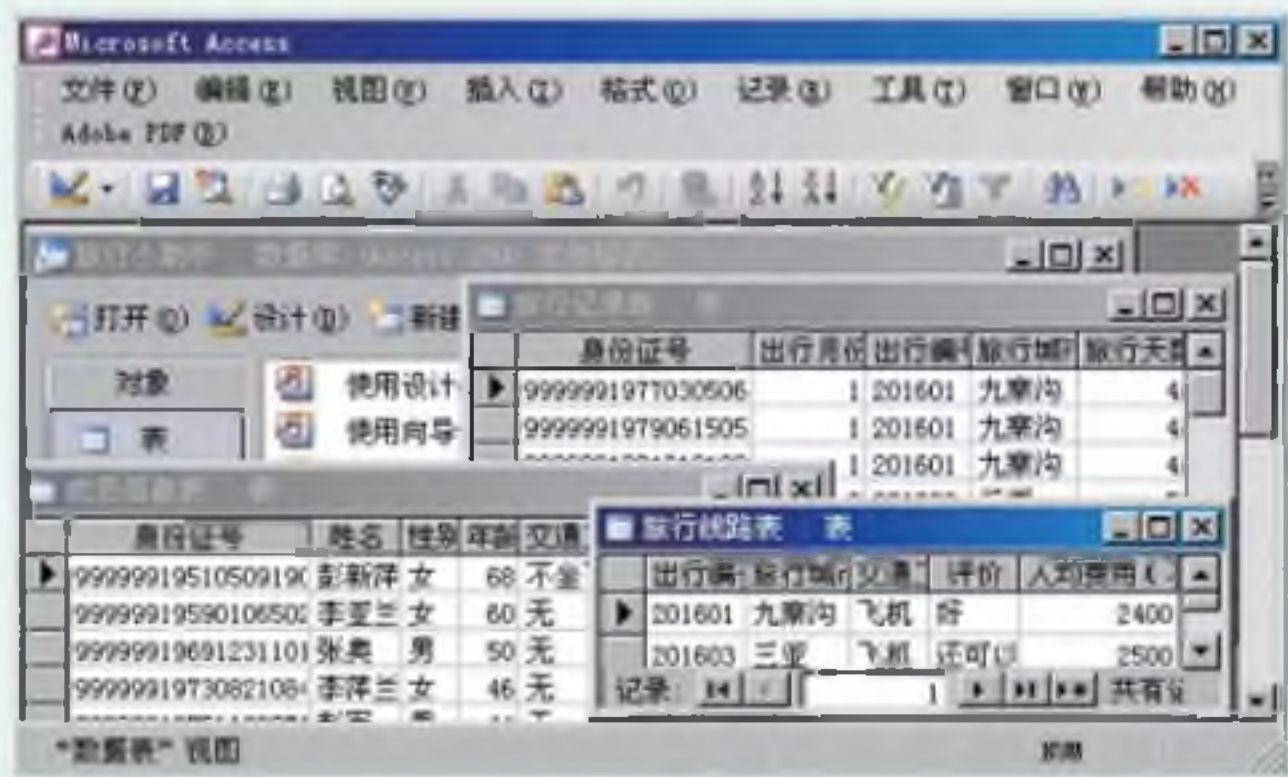


图3.3.8 “旅行小助手”数据库

利用Python访问Access数据库一般可分为以下5个步骤：

- (1) 建立与数据库的连接；
- (2) 根据需要产生记录集；
- (3) 访问记录集，处理数据；
- (4) 根据需要把处理好的数据更新到数据库中；
- (5) 断开与数据库的连接。

请打开教科书配套资源中的“利用Python访问mdb.py”文件，运行程序，观察运行结果；阅读程序代码，体会利用Python访问mdb数据库的基本步骤。这个程序利用win32com.client模块访问Access的mdb文件，因此，需要先下载并运行pywin32安装包。

数据库应用系统的设计一般包括系统总体结构设计、代码设计、数据库（文件）设计、输入/输出（I/O）设计、模块功能与处理过程设计等步骤。“旅行小助手系统”在完成了系统界面设计和数据库设计之后，还需要经过编写应用程序、调试等环节才能成为真正的应用系统。



拓展练习

请打开教科书配套资源中的“旅行小助手系统.py”，体验该程序的功能，并思考该程序还可以增加哪些功能。



拓展知识

我们通过功能设计、应用程序界面设计、采集数据和建立数据库，体验了“旅行小助手系统”的部分设计过程。实际上，数据库的设计绝非易事，需要一步步完成。首先通常需要与数据库潜在用户进行沟通，收集需要存储的数据和对数据的存取需求，然后建立一个实体关系模型，之后建立基于模型的关系并规范化这些关系。而开发数据库应用系统可以采用MVC模式，MVC为Model、View、Controller的简称。此模式中，Model（模型）是应用程序中用于处理应用程序数据逻辑的部分，通常模型对象负责在数据库中存取数据；View（视图）是应用程序中处理数据显示的部分，通常视图是依据模型数据创建的；Controller（控制器）是应用程序中处理用户交互的部分，通常控制器负责从视图中读取数据，控制用户输入，并向模型发送数据。

3.4 加密与解密

数据应用既为我们带来便利，同时也存在许多安全隐患，如未经许可数据被复制、篡改、暴露或破坏。因此，在数据传输、交换、存储、处理的过程中，我们必须采取相应的策略，进行数据保护，如设置密码和验证码、对数据加密、验证数据完整性、安装防火墙及安全软件、对数据进行备份等，确保信息的完整、可用、保密和可靠。在本节中，我们将了解如何使用密码保护数据。



学习目标

- ★ 理解数据加密和解密的概念及原理。
- ★ 了解数据加密的历史。
- ★ 会用 Python 实现简单的加密算法。
- ★ 会分析生活中由密码引起的数据安全问题，理解对数据进行保护的意义。

数据聚集和共享为人们充分应用数据创造便捷的生活提供了条件，但在享受数据带来的便利的同时，也伴随着数据安全问题。本节我们将围绕“神秘密码之谜”项目展开学习，通过项目活动，学习数据加密和解密的知识，树立保护数据安全的意识。

本项目主要包含“解开网站账号被盗之谜”和“让数据‘隐身’”两个任务。



任务一 解开网站账号被盗之谜

※ 活动1 防范“盗窃”行为

不少人有密码被盗的经历，这往往会导致个人信息泄露，甚至是财产损失。你是否有使用账号、密码的经验？请你与同学交流、

讨论并上网搜索表3.4.1中列出的密码盗窃方式，思考相应的防盗措施，填入表中。

表3.4.1 密码盗窃与防盗分析表

盗窃方式	防盗措施
肩窥方式	我们可以在输入密码时遮挡自己的操作过程，防止别人偷看密码信息；或者确定环境安全后再进行操作
字典破解	
暴力破解	

● 密码

密码是指用来核对用户ID以验证用户就是本人的一组字符。

※ 活动2 设置安全密码

请你打开教科书配套资源中的“数据安全性测试.py”，运行并测试穷举搜索一个5位、7位和9位的数字密码，分别需要多长时间，然后填写表3.4.2。

表3.4.2 穷举搜索密码需要的时间

密码长度	密码值	破解时间
5位	12567	1.0毫秒
	98302	15.6毫秒
7位		
9位		

具体实验结果会根据计算机性能不同而有所不同。

varin 和 varout 是程序中的两个文本框对象，分别用来接收输入的密码和显示解密所需时间。get 和 set 是文本框的两个方法，分别用于接收文本框中的数据和在文本框中显示输出数据。

根据你实践的结果研究下面的Python代码，想一想，如果密码中还有字母，破解的时间会变长吗？请与同学讨论创建安全密码的方法。

```
def jiemi():
    d1=datetime.datetime.now()           #获取当前系统时间d1
    p=int(varin.get())                   #获取输入文本框的数字密码
    for i in range(0,p+1):               #从0循环到正确密码数值
        if i==p:                         #如果密码相同
```

```
d2=datetime.datetime.now() #获取当前系统时间d2
d=d2-d1                    #取得时间差
#在输出文本框中显示解密用时
varout.set(str(d.seconds)+"秒"+str(d.microseconds/1000)+
"毫秒")
```

● 创建安全密码的一般技巧

- 使用长度不少于8个字符的密码。密码长度越长越不容易被破解。
- 在可能的情况下，尽量使用字母、数字和特殊字符（如\$、#）相结合的密码。
- 不要使用电话号码、身份证号码或生日等信息作为密码。
- 不要使用整个用户ID或用户ID的一部分作为密码。
- 不要使用字典中能找到的词语作为密码，即使是字母次序颠倒过来的常用词语也不可以。

为保障数据安全，还可采取限制登录尝试次数、使用验证码、使用用户的具有排他性的自然特征（如指纹、人脸等）作为身份验证依据等措施。

 任务二 让数据“隐身”

※ 活动1 古老的“隐身术”

在古代，人们是怎样秘密传送数据的呢？请你利用网络搜索相关信息，了解数据加密的发展历史，并完成表3.4.3。

表3.4.3 数据加密技术的发展

时间	加密方式
683年	拆字法，将明文中的文字进行组合生成新的字，即为密文。比如“十二月”合起来为“青”

● 加密

自古以来，在管理国家、指挥战斗或是经济来往中，都需要高效、安全的数据通信系统。古人就已会用很多办法将数据伪装起来，使得只有联络好的数据接收者才能读懂。比如，将数据刻在木板上，然后再覆上一层蜡，使其成为一个看上去很普通的刻写板；使用一套尺寸不等、形状各异的“阴符”，每支符都表示特定的含义等。这些都是数据的加密。

加密就是将原始信息（数据）隐匿起来，使之在缺少特殊信息（数据）时不可读。原始信息（数据）称为明文，加密后的信息（数据）称为密文。将密文还原成明文的过程称为解密（或解码）。

※ 活动2 揭秘“隐身术”

恺撒在征服高卢、袭击日耳曼和不列颠的多次战斗中频繁使用加密技术。苏托尼厄斯在公元2世纪写的《恺撒传》中对恺撒用过的一种加密技术进行了详细的介绍。恺撒只是简单地将明文中的每一个字母用字母表中该字母后的第3个字母替换。例如，将明文中的a用d替换，b用e替换，……，z用c替换，这就是恺撒密码。

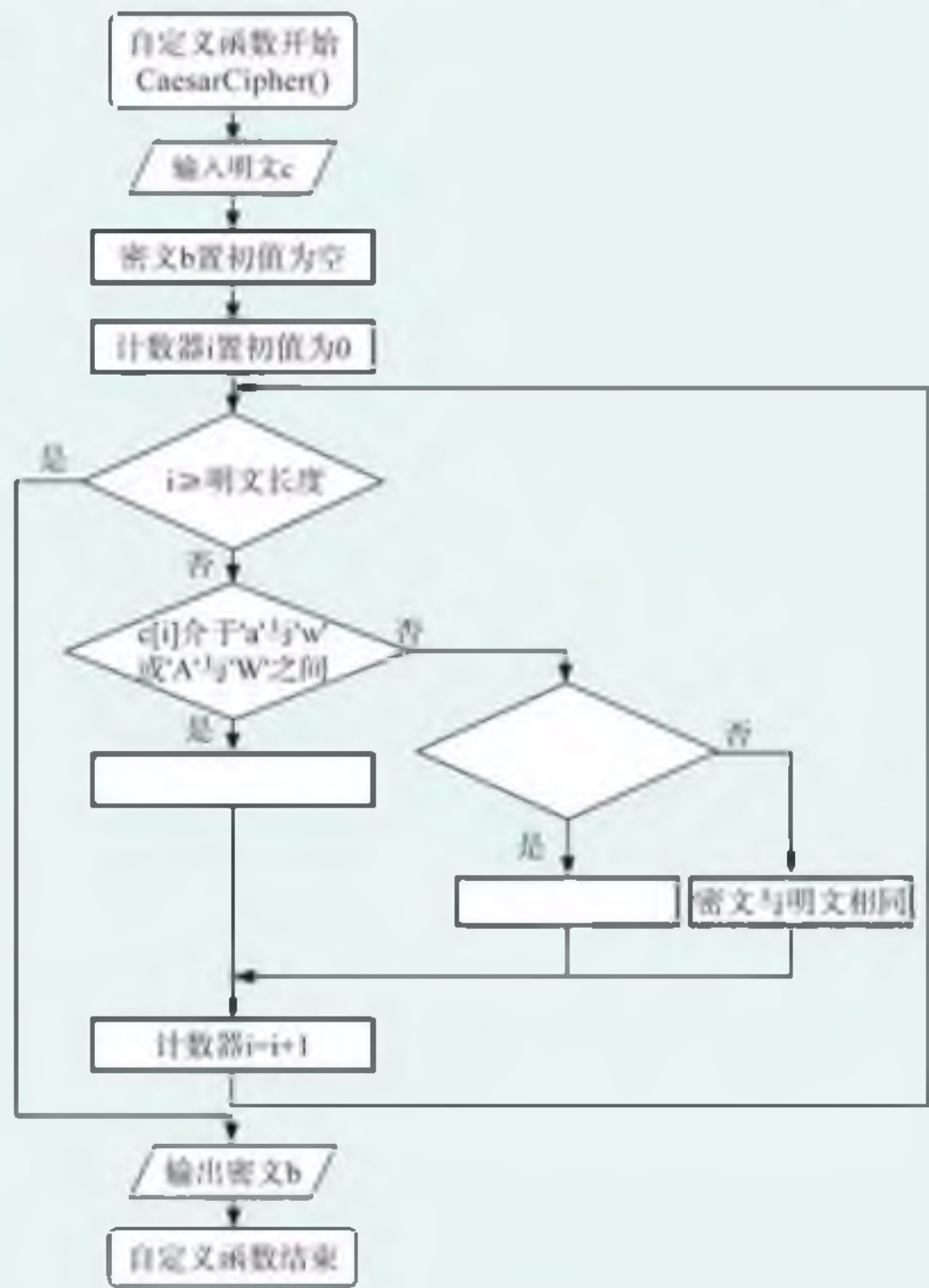


图3.4.1 恺撒加密算法流程图

请将流程图中的空白部分补充完整。

打开教科书配套资源中的“恺撒密码.py”，根据流程图，将程序

补充完整并运行程序，体验数据加密的过程。

部分核心算法代码：

```
def CaesarCipher():                                #Button按钮激发函数
    c=mingwen.get("0.0", "end")[:-1]               #获取Text组件mingwen的明文内容
    b=""
    miwen.delete("0.0", "end")                     #清空Text组件miwen的内容
    for i in range(len(c)):                         #获取明文内容的每一个字符，并加密
        #判断a~w或A~W间的字母
        if 'a'<=c[i]<='w' or 'A'<=c[i]<='W':
            b=b+chr(ord(c[i])+3)                   #生成密文
        #判断x~z或X~Z间的字母
        elif 'x'<=c[i]<='z' or 'X'<=c[i]<='Z':
            b=b+chr(ord(c[i])-26+3)                #生成密文
        else:
            b=b+c[i]                                #字母以外的明文内容不变
    miwen.insert("0.0",b)                          #在Text组件miwen中显示结果
```



mingwen 和 miwen 是两个文本框对象，用于接收待加密的文本和显示加密后的密文。get 方法用于接收文本框中的数据，delete 方法用于删除文本框中的内容，insert 方法用于将数据插入文本框内。

● 加法密码

加法密码又被称为移位密码。在加法密码算法中，明文中的所有字母都在字母表上向后（或向前）按照一个固定数目进行偏移后被替换成密文。例如，当偏移量是3的时候，所有的字母A将被替换成D、B变成E、以此类推，X将变成A、Y变成B、Z变成C。

恺撒密码作为一种最为古老的对称加密体制，在古罗马的时候已经很流行，它是加法密码的典型代表。

用Python语言实现加密算法时，明文和密文都被抽象为字符串类型的数据。Python中的字符串由一串字符组成，用引号引起来表示，比如“abCEd”。类似于列表的操作，我们可以指向或操作字符串中的某一个字符，比如c[0]表示字符串中的第一个字符。我们还可以用Python系统函数对数据进行特定的操作，len函数可以获取字符串中字符的个数，ord函数可以将字符类型的参数转换为对应的整数值，chr函数可以将整数类型的参数转换为对应的字符。

了解密码安全与加密、解密的技术能够帮助我们保护个人隐私。同时，我们也要从自己做起，遵守相关法律和必要的道德规范，共同维护健康、良好的数据环境。



加密算法操作中的一组数字被称为密钥。比如，恺撒密码的密钥为3，加密和解密使用同一个密钥，被称为对称加密体制。



拓展练习

请参照恺撒加密算法，尝试设计并实现自己的加法密码算法。



拓展知识

当今数字时代，数据已经广泛应用于国家安全、公共治理、企业发展、民生改善等诸多领域，大大提高了人们工作、生活的便利程度，但各种数据也存在安全风险。为此，国家出台多部法律，大力实施数据保护，维护国家安全和网络秩序，保障社会、公民利益，促进经济社会信息化健康发展。如2017年6月1日起施行的《中华人民共和国网络安全法》，明确要求网络运营者应当按照网络安全等级保护制度的要求，履行安全保护义务，保障网络免受干扰、破坏或者未经授权的访问，防止网络数据泄露或者被窃取、篡改。《中华人民共和国刑法》第二百八十五条至第二百八十七条，对非法侵入计算机信息系统，对计算机信息系统功能进行删除、修改、增加、干扰，造成计算机信息系统不能正常运行，利用计算机实施金融诈骗、盗窃、贪污、挪用公款、窃取国家秘密等行为进行了量刑定罪。

单元学习评价

1. 进制转换。

$$10110100_{(2)} = (\quad)_{(10)}$$

$$3D_{(16)} = (\quad)_{(10)}$$

$$255_{(10)} = (\quad)_{(16)}$$

$$100_{(10)} = (\quad)_{(2)}$$

$$E2_{(16)} = (\quad)_{(2)}$$

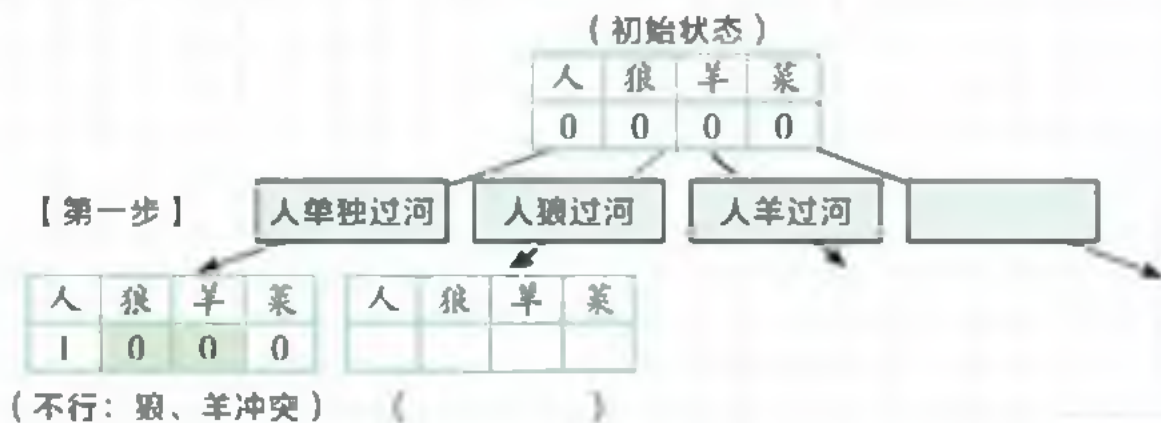
$$11100101_{(2)} = (\quad)_{(16)}$$

任意R进制数（R为大于等于2且不等于10的正整数，下同）转换为十进制数时，都可采用_____方法。十进制数转换为R进制数时，都可采用_____方法。

2. 一段时长为1分钟，采样频率为44.1kHz，量化位数为16位，双声道立体声的无压缩音频（如基于PCM编码的wav格式），占用的存储空间是_____MB（精确到0.1）。

3. 人、狼、羊、菜过河问题：有一个人带着一只狼、一只羊和一捆白菜，来到一条河边，河边只有一条小船，人每次过河最多只能带一样，如果人不在现场，狼就要吃羊，羊就要吃菜。他应该怎样安排过河呢？请完成下面的“树”结构分析图，帮他找到可行的过河方案。

提示：可约定对象在左岸用0表示，在右岸用1表示。



4. 当今信息技术发展迅速，人们的网络交流日益频繁，各种信息终端及网络中存储的个人信息越来越多，不恰当的使用行为往往会造成数据泄露，带来安全隐患。通过本单元的学习，请思考大家可能会有哪些不安全的信息技术使用习惯，并提出相应的数据保护方案。

不安全的信息技术使用习惯	相应的数据保护方案

★ 单元学习总结



第 4 单元 计算与问题解决

计算科学是关于计算的学问，它主要研究哪些问题是可计算的以及怎样去计算。计算是获得信息的一种过程，所以计算是动态的，信息的获得是计算的延伸。可以说，问题解决的过程，实质上是描述和变换信息的过程。

本单元我们将对“计算”进行更多探究。我们将进一步了解计算与算法的关系，明确算法的基本特征，尝试用枚举的思想实现简单算法；掌握数值类及非数值类问题求解中的常用算法思想；学会分析问题，针对问题的解决选择合理的算法，运用已学的计算知识解决实际问题。



人类必将生活在一种程序设计的世界里。在这个世界里，人类文化与程序设计不仅并行存在，而且会互相联系，融合为一种全新的人类思想。

—— 叶尔肖夫 (Yershow)

4.1 算法及其特征

算法并不遥远，它就在我们身边。日常生活中，很多看似平常的做法都蕴含着一定的道理，如果将这些做法抽象成数学描述可能就是一些非常有效的算法。算法可能是一个计算公式，可能是一个赢得游戏的策略，也可能是一个解决综合问题的复杂方案。



学习目标

- ★ 熟悉将解决问题的方法归结为一系列清晰、准确的步骤的过程。
- ★ 了解算法的基本要素和重要特征。
- ★ 运用恰当的方式描述算法。
- ★ 运用 Python 语言实现简单算法，解决问题。

软件设计师是业务领域和技术领域的专家。面对客户需求，软件设计师要对问题进行整体分析以找到解决方案，通过恰当的方式与客户沟通；面对技术问题，软件设计师需要借助计算机编程解决实际问题。我们将通过“软件开发社招新”项目讨论计算机编程解决问题的过程。

本项目主要包含“探讨面试题的解决方案”和“求解‘谁是冠军’”两个任务。



任务一 探讨面试题的解决方案

※ 活动1 寻找“开关对应关系”

软件开发社团要招募新成员，报名的同学要经过面试才能加入。第一关面试题是：“一个房间有3盏灯，房间外有3个开关分别控制这3盏灯，如图4.1.1所示。在只允许进房间一次的情况下，如何判断哪个开关控制哪盏灯？”



图4.1.1 开关对应关系

解决问题的起点是对问题的描述，而算法则是对解决问题过程的形式化描述。

设计算法是解决问题的核心，它的基本任务是对问题进行定性分析和定量分析，遵循算法的特征和约定，寻求计算的方法和规则，明确解决问题的途径。

从表面上看，灯只有亮、灭两种状态，但是灯又具有一种特殊性，即开灯的同时会伴随发光发热，因此灯被触摸时还有冷、热两种状态。综上所述，一盏灯可能有4种不同的状态。而在房间内共有3盏灯，完全可以保证每盏灯的状态都是唯一的。由于题目中并没有限制开关按动次数，所以3个开关的闭合状态是可以随意改变的。如何能使3盏灯处于不同的状态？

请在下框中写下你的步骤，在小组中比比谁的方法更快捷、更合理。

第一步：

第二步：

.....

归纳有效解决问题的具体步骤，对问题进行定性分析和定量分析，就能得出答案。

首先开1号、2号两个开关，2分钟后关闭1号开关，然后进房间，显然亮着的灯由2号开关控制。接下来摸一下另外两盏不亮的灯，发热的灯肯定由1号开关控制。最后确定3号开关控制的灯。请同学们一起来完善如图4.1.2所示的“开关对应关系”流程图。

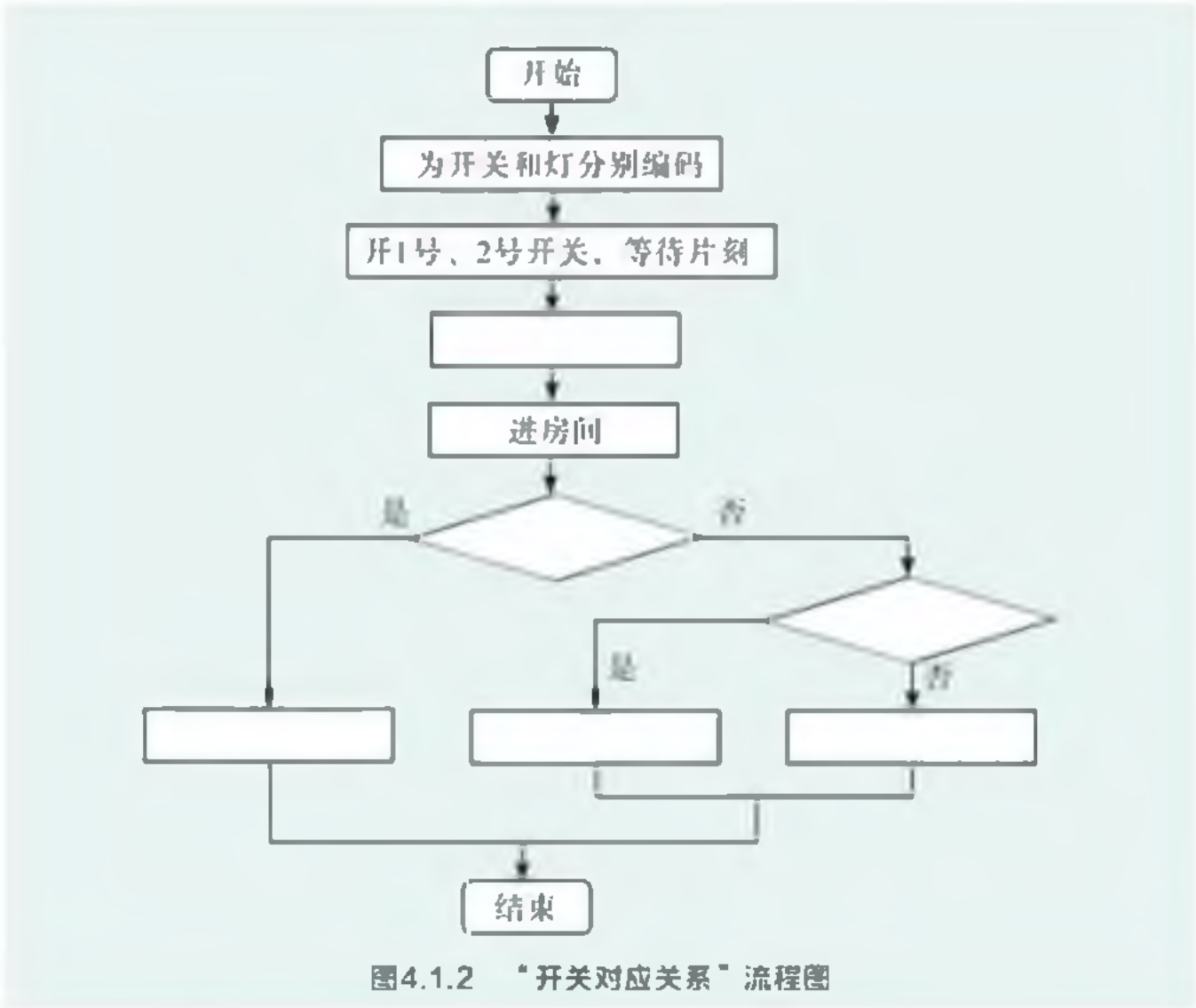


图4.1.2 “开关对应关系”流程图

● 算法的重要特征

对算法的评价主要从时间复杂度和空间复杂度来考虑。

- 有穷性。算法必须能在执行有限个步骤之后终止。
 - 确切性。算法中的每一次运算都有明确的定义，具有无二义性，并且可以通过计算得到唯一的结果。
 - 输入项。一个算法有0个或多个输入，以刻画运算对象的初始情况，所谓0个输入是指算法本身给出了初始条件。
 - 输出项。算法一定要有输出。任何算法都不能“无功而返”。
 - 可行性。算法中执行的任何计算都可以在有限时间内完成（也称为有效性）。算法中的运算都必须是可以实现的。
- 从某种意义上说，算法也是一种数学模型。一般而言，问题求解的第一步是数学建模。用数学语言描述实际现象，将现实世界的问题抽象成数学模型，就可能发现问题的本质并判定其能否求解，继而找到求解该问题的方法和算法。

※ 活动2 定量分析，寻找“被污染的药丸”

面试的第二关是实验题。如图4.1.3所示，有4个分别装了4种药丸的药瓶，里面每颗药丸都有单颗标准质量，其中有一个药瓶中的所有

药丸都被污染了。每颗被污染的药丸比正常药丸增重1克。请在只允许称量一次的情况下，判断出哪个药瓶中的药丸被污染了。



图4.1.3 4瓶药丸

如果从每个药瓶中取出1颗药丸分别进行称重，肯定可以判断出哪颗药丸被污染了，但是这种做法显然不符合“只能称量一次”的要求。你能改进判断方法吗？

考虑1颗药丸的重量变化，如果药丸被污染，则增重_____克，否则增重_____克。

从某一个药瓶中取出 n 颗药丸，如果被污染，则增重_____克，否则增重_____克。

如果我们从不同的药瓶中取出不同颗数的药丸，你能根据增重情况找出被污染的药丸吗？

从第1个药瓶中取出1颗药丸，从第2个药瓶中取出2颗药丸，从第3个药瓶中取出3颗药丸，从第4个药瓶中取出4颗药丸，共10颗药丸。如果增重_____克，则_____号药瓶中的药丸被污染。

回顾算法的特点，思考一下，在这个问题中，哪些信息属于输入、哪些信息属于输出呢？

请设计程序并运行，使输入10颗药丸的总重量及4种药丸的单颗标准质量就可以看到结果，找到被污染的药丸。

※ 活动3 巧用运算，寻找“误删的ID号”

面试的第三关是程序设计题。学校历届校友的数据存储在学校网络中心服务器中（共10000条，无重复数据），某管理员由于误操作删除了一位校友的ID号（8位整数）。恰好在备份文件中保存了所有人员的ID号（无重复数据，无序）。怎样快速找出被误删的ID号以便恢复数据？

仔细分析问题，我们发现实际需要参与分析及处理的只有ID号，而且这些ID号的特征也很明显。请归纳ID号的特征，写在下面的横线上。

算法的每一步都是一个准确表达的步骤或指令，旨在通过这一系列步骤在有限的时间内解决实际问题。

ID号的特征

- ①数据类型及大小范围：_____
- ②数据在两个文件中出现的次数：_____
- ③备份文件中ID号总和与故障文件中的ID号总和的差值为：_____

通过上面的分析，我们发现计算备份文件ID号总和与故障文件ID号总和的差值就是误删的ID号。

充分利用数值及计算的特点，可以将复杂的查找过程转换成简单的计算。

程序是我们和计算机沟通的主要途径。编写程序就是使用程序设计语言，将算法用程序代码的形式输入，并最终得到结果的过程。让我们一起来完成下面这个程序，并回顾文件读取的方法。

```
target=_____ #设置初始值
f1=open('copy.txt', 'r') #打开备份文件
list1=f1.readlines() #按行读取备份文件
for line in _____: #依次处理列表list1中的数据
    target= target+int(line) #将读取的数据做加运算
f1._____ #关闭备份文件
f2=open('trouble.txt', 'r') #打开故障文件
list2=_____ #按行读取故障文件
for _____: #依次处理列表list2中的数据
    target= _____ #将读取的数据做减运算
_____ #关闭故障文件
print("被误删的ID号是:", _____) #输出被误删的ID号
```

请设计copy.txt文件（10个数，每行一个）和trouble.txt文件（9个在copy.txt文件中的数，每行一个）的信息，看看程序运行结果是否是被误删的ID号。



任务二 求解“谁是冠军”

※ 活动 尝试枚举

这次面试的冠军在A、B、C、D四位同学中。A说：“不是我。” B说：“是C。” C说：“是D。” D说：“C说的不对。”已知四人中有一人说了假话。你能判断出到底谁是冠军吗？说出你的

结论和判断过程。

结论：_____是冠军。

判断过程：_____

● 枚举

我们常利用计算机运算速度快、精确度高的特点解决实际问题。在设计算法时，最简单的方法就是“直译”我们的思维过程。有一种算法是把所有可能的答案一一列举，合适就保留，不合适就丢弃。这种方法称作“枚举”或“穷举”。

 枚举法解决问题的一般结构：循环 + 判断。其优势在于正确性容易证明。

在不知道谁说真话、谁说假话的情况下，最简单的方法就是把所有可能都枚举出来。因为只有一位冠军，所以可以枚举选手的编号，并对A、B、C、D四个人的话进行判断。

计算的本质是完成一系列算术运算和逻辑运算。因此在进行计算时，首先要将各种类型的数值问题转化为计算机能够执行的基本运算。在本任务中，我们需要把每个人说的话转化成计算机能够执行的表达式。如A说：“不是我。”可以表示为“i!=‘A’”，其中i为枚举的冠军选手编号。

 i!=‘A’的运算结果是一个逻辑值，在数值计算中会自动转换成1（真）或0（假）。

请分析以下代码的含义，理解解题思路，并在横线上填写语句的功能。

```
champion=['A','B','C','D']    #设置选手列表
for i in champion:    #_____
    cond=(i!='A')+(i=='C')+(i=='D')+(i!='D')    #_____
    if cond==3:
        print("冠军是:",i)
```

请在Python中输入代码并运行程序，看看谁是冠军。

算法来源于生活，服务于生活。程序同样如此。

 **%** 是 Python 中的求余运算符，如 $9\%4=1$ 。



拓展练习

假设有20颗糖果，两人轮流取糖果，每次可以取1至2颗，拿到最后一颗糖果的人获胜；你可以自由选择先取还是后取糖果。请设计一个必赢的算法并用流程图表示。
提示：必赢方案与求余运算及数字3有关。



拓展知识

用PAD描述算法

PAD图（问题分析图）是一种算法描述工具，它所描述的程序结构十分清晰。如图4.1.4所示，最左边的竖线是程序的主线，即第一层控制结构。随着程序层次的增加，PAD图逐渐向右衍生，每增加一个层次，图形向右扩展一条竖线。程序从最左边上端的节点开始按自上而下、从左到右的顺序执行。



图4.1.4 PAD图

4.2 数值计算

人们对计算机的最初应用大多是数值计算，主要借助计算机运算速度快、精确度高的特点来解决各种数学问题，如函数的计算、方程的求解、数列求和等都属于数值计算。



学习目标

- ★ 感受数据的图形化表示。
- ★ 设计解析式或迭代方程，进行数值计算，解决问题。
- ★ 了解数值类算法在实际问题解决时的应用及常用方法。

数学是计算机科学的理论基础之一。在实际问题的求解过程中，有许多数学分支，如解析几何、离散数学等，都是计算机科学所依赖的重要理论工具。借助计算机程序，可以解决很多数学方法所描绘的数值计算问题。本节我们将围绕“与数学公式面对面”项目探讨在中学数学领域中常见的数学公式与程序设计的有趣结合。

本项目主要包含“绘制数学函数曲线”和“求解斐波那契数列”两个任务。



任务一 绘制数学函数曲线

在数学课上经常需要手工绘制函数图像，今天我们借助计算机来绘制函数图像。

※ 活动1 用WPS表格绘制正弦曲线

利用平时使用的电子表格软件就能绘制函数图像。方法如下：若以 30° 为间隔，绘制 $0\sim 360^\circ$ 之间的正弦函数图像，则首先需要在WPS中完成下列表格数据的计算。请填写表4.2.1。



用描点法画函数图像一般分三步：首先建立平面直角坐标系，其次根据解析式计算出若干点的坐标并画在坐标系内，最后将这些点依次连接起来。

表4.2.1 函数计算

	A	B	C	D
1	x	sin(x)	sin(-x)	sin(2x)/2
2	0	=sin(pi()*A2/180)		
3	30			
4	60			
⋮	⋮			
14	360			

填表之后，选择相应的数据，建立折线图图表，设置x轴数据系列格式，可绘制出如图4.2.1所示的函数图像。

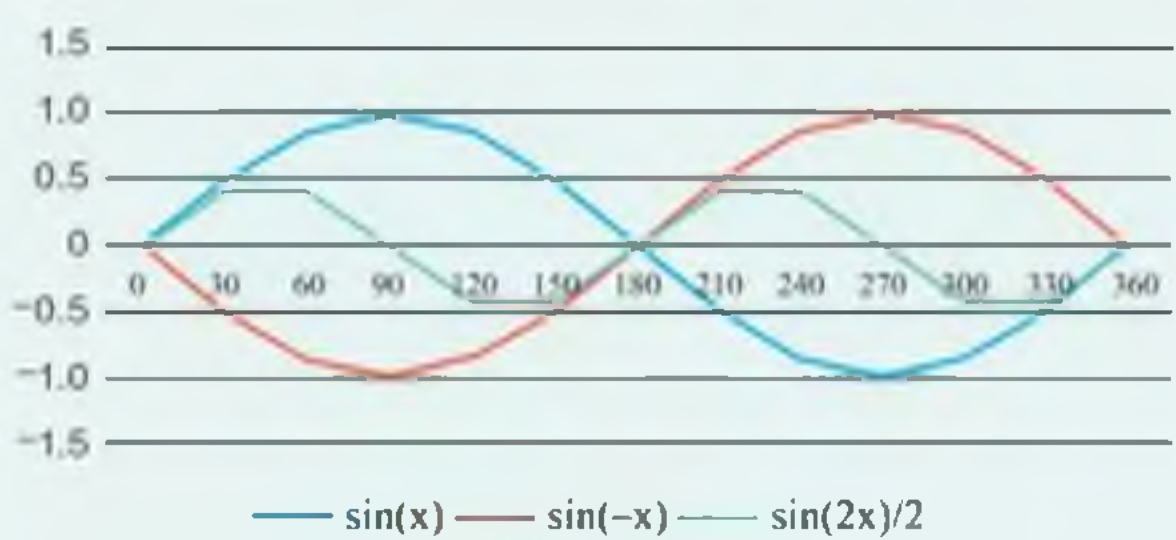


图4.2.1 利用WPS绘制的函数图像

仔细观察图像，会发现图像的关键点太少，精度不够，图像不光滑。要想提高图像的光滑程度，就要减小角度间隔，但间隔增加，工作量也会随之增加：每隔1°画一个点，数据表上就会增加300多行新数据；如果以0.1°为间隔，将有3000多行数据。

※ 活动2 利用Python绘制正弦曲线

借助计算机程序描点，可以达到速度快且精确度高的效果。下面我们尝试利用Python编写程序绘制正弦曲线。

在Python中，绘制函数图像一般要用到numpy和matplotlib两个模块，这两个模块需要另外安装。

• numpy模块简介

numpy是一个科学计算包，其中包含很多数学函数，如三角函数、矩阵计算方法等。通过该模块中的arange函数可以创建一个等差数列。如在 $0 \sim 2\pi$ 之间每隔0.01取个值，则可以用arange(0,2*numpy.pi,0.01)来表示，其中numpy.pi表示 π 。下列代码可以产生sin(x)的若干个关键点。

```
#加载numpy模块并取一个简洁的别名为np，便于后续引用
import numpy as np
#x在0到2π之间，每隔0.01取一个点
x=np.arange(0,2*np.pi,0.01)
y=np.sin(x)                #通过解析式计算列表x对应的列表y的值
```



numpy还支持处理大型矩阵、矢量运算、线性代数等功能。



模块安装方法：
(1) 把这些文件直接复制到你的Python路径下的Lib/site-packages文件夹中；
(2) 运行python setup.py install；
(3) 利用安装工具，如pip、easy_install、distribute等。

• matplotlib模块简介

matplotlib模块是Python中最出色的绘图库，功能很完善。调用matplotlib.pyplot时，坐标系可以根据数值范围自动生成。

matplotlib的绘图原理很简单，利用plot画线函数就可以在直角平面内轻松地将(x, y)坐标点对连接成平滑曲线。例如：在上述代码的适当位置增加下列语句，就可以将刚才生成的关键点连接起来。

```
#加载matplotlib.pyplot并取别名为plt
import matplotlib.pyplot as plt
plt.plot(x,y)                #将点对连线
plt.show()                  #将绘制的图像窗口显示出来
```

参考上述代码，让我们一起来完善以下Python程序，尝试绘出“sin(x)” “sin(-x)” 和 “sin(2x)/2” 的图像。

```
_____ #加载numpy模块并取别名为np
#加载matplotlib.pyplot并取别名为plt
import matplotlib.pyplot as plt
_____ #列表x在0到2π之间，每隔0.01取一个点
_____ #求sin(x)对应的列表y1的值
_____ #求sin(-x)对应的列表y2的值
y2=np.sin(-x)                #求sin(2x)/2对应的列表y3的值
_____ #绘制sin(x)的图像
plt.plot(x,y1)
```

```
plt.title('sin(x)')
plt.xlabel('X')
plt.ylabel('Y')
plt.show()
```

#绘制sin(-x)的图像
#绘制sin(2x)/2的图像
#设置图像标题
#设置X轴标题
#设置Y轴标题
#将绘制的函数图像窗口显示出来

运行上述代码，对比在活动1中生成的图像，程序绘制的函数图像果然平滑了很多，如图4.2.2所示。

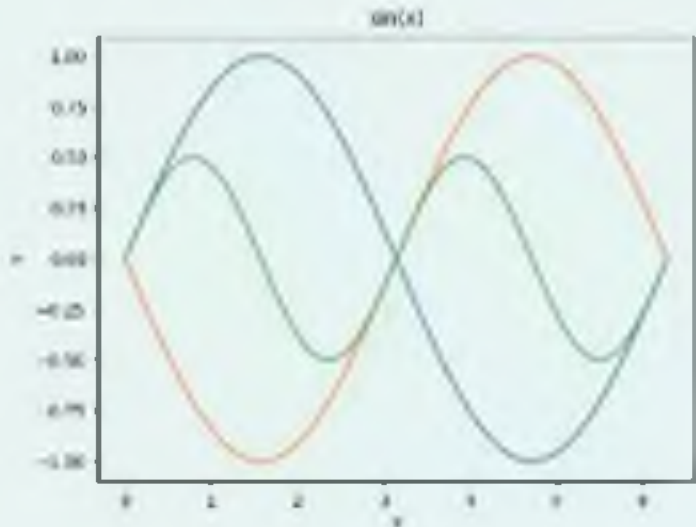


图4.2.2 利用Python程序绘制的函数图像

任务二 求解斐波那契数列

※ 活动1 用WPS求解数列

斐波那契在《计算之书》中提出了一个有趣的兔子问题：假设一对兔子每个月可以生一对小兔子，一对兔子出生后第2个月就开始生小兔子。则一对兔子一年内能繁殖成多少对？10年呢？

根据描述，兔子的对数如图4.2.3所示。



图4.2.3 兔子数量

从第3个月起，每个月大兔子的对数等于上个月大兔子与小兔子的对数之和（即上个月兔子总对数），每个月小兔子的对数等于上个月大兔子的对数（即上上个月兔子总对数）。

使用电子表格可以很方便地求解，如图4.2.4所示。

	A	B
1	月份	兔子对数
2	1月	1
3	2月	1
4	3月	2
5	4月	3
6	5月	=E5+E4

图4.2.4 求解数列的公式设置

我们发现，当计算到第74个月的时候，由于数据范围及表示精度的问题，导致结果出错，如图4.2.5所示。

72	71月	308061521170129
73	72月	498454011879264
74	73月	806515533049393
75	74月	1304969544928660
76	75月	2111485077978050

图4.2.5 数列求解过程

※ 活动2 用Python求解数列

第1个月和第2个月的兔子对数之和为第3个月的兔子对数，第2个月和第3个月的兔子对数之和为第4个月的兔子对数……，每个月的兔子对数是前两个月的兔子对数之和，又同时作为下一个月兔子对数的加数。这种重复反馈的过程称为迭代。

迭代法也称辗转法，是用计算机解决问题的一种基本方法。迭代通常是为了接近并到达所需的目标或结果。每一次对过程的重复被称为一次“迭代”，而每一次迭代得到的结果会被用来作为下一次迭代的初始值。

由于在迭代系列中的每个月份兔子对数只跟前两个月有关，因此在编写程序时，只需两个变量*i1*和*i2*分别记录上上月和上月的数据。

迭代计算的示意图如图4.2.6所示。



图4.2.6 计算的迭代示意图

请根据示意图完善下列程序代码。

```
def fib(n):
    #利用迭代求斐波那契数列的第n个数
    f2=f1=1                                #第1个月、第2个月初值设定
    for i in range(3,n+1):                #_____
        f1,f2=f2,f1+f2
    return f2
n=int(input('输入需要计算的月份数: '))
print('兔子总对数为:',fib(n))           #输出最终解
```

程序运行结果如下：
 输入需要计算的月份数：74
 兔子总对数为：1304969544928657

利用迭代算法解决问题，有三个关键步骤：（1）确定迭代变量，如活动2中的f1、f2；（2）建立迭代关系式；（3）对迭代过程进行控制。这是编写迭代程序必须考虑的问题，不能让迭代过程无休止地重复执行下去。

现代自然科学和工程电子技术的研究过程中，都离不开大规模的数学计算问题。例如：数学类课程中的线性方程求解、微分方程求解、概率统计等；实用性和实验性技术应用中的模拟核试验、油田开发、飞机设计等。



拓展练习

1. 尝试用Python绘制 $y = x^2 - 2x + 1$ 的图像。
2. 尝试用辗转相除法求解两个正整数的最大公约数。



拓展知识

计算 π 的值

圆周率 π 在许多数学分支领域都有非常重要的作用。每年的3月14日为圆周率日。

我国最早记载圆周率的古书是《周髀算经》，里面有“径一周三”的记载。古代数学家刘徽利用割圆术求得圆周率的近似值，后来祖冲之通过计算得到了更精确的圆周率，精确到小数点后7位，这个值被用了千年之久。

电子计算机的出现使 π 值计算有了突飞猛进的发展。目前，利用计算机可以求得更精确的 π 的近似值，精确位数超过了 10^{12} 。

4.3 非数值计算

在数值计算中，我们更多考虑的是“数”，但计算应该是一个更广泛的领域。计算的对象可以是自然界和人类社会的一切事物。更确切地说，计算的对象可以是某些信息，如数据、文字、语言、图形、知识、事物的运动过程及思维过程。如果说数值计算主要探讨数学问题的话，那么非数值计算更多探讨“算法”问题。

数据是普遍存在的，甚至可以说对象即数据；对数据的分析、处理都属于计算的范畴。选择一个合适的算法，设计出平实、易读、易懂的程序，正确、高效地解决实际需求，是计算的本质。



学习目标

- ★ 运用合适的算法形成解决问题的方案。
- ★ 了解算法设计中的分治思想，并运用二分查找解决实际问题。
- ★ 体验递归算法，并结合具体问题开展编程实践。

许多程序设计问题的解决，要依靠标准算法和现成的模型，更需要编程者开阔思路，提出一些新颖、巧妙的算法，或者设计出一些独特的数据结构来支撑和实现算法。在解决非数值类计算问题时，一些基础的思维方式可以借鉴，如分治、递归、解析等。本节我们将围绕“生活中的算法”项目，尝试用“算法的眼睛”看待生活，用“算法的思维”去解决实际问题。

本项目主要包含“巧翻字典”和“玩转‘汉诺塔’游戏”两个任务。



任务一 巧翻字典

※ 活动 统计查字典次数

查汉字、查单词、查成语等查字典的活动，早已成为我们学习生

活的一部分。假设一本字典大约1000页，目标信息在第328页，请在表4.3.1中记录你的翻页过程，和同学们比一比，看谁翻的次数最少。

表4.3.1 翻字典过程

次数	翻至页码	下一步决策
第1次		
第2次		
第3次		
第4次		
.....		

有的同学翻得特别快，他们用了什么方法呢？原来看似普通的翻字典，不仅是一门技术，更是一种能力，是算法思想的体现。

● 分治策略

分治的设计思想，是将一个难以直接解决的大问题，分割成一些较小的同类问题，各个击破，最终达到解决问题的目的，二分查找实际上就是分治策略的一种典型运用。

凡治众如治寡，分数是也。（摘自《孙子兵法》）

● 二分查找

二分查找又叫折半查找，该方法主要将数列有序排列，采用跳跃式的方式查找数据。以递增数列为例，先以中点位置的元素作为比较对象，如果要找的元素值小于该中点元素，则将待查序列缩小为左半部分，否则为右半部分。每一次比较后都可以将查找区间缩小一半。

查找的基本算法有：顺序查找、二分查找、分块查找、哈希查找等。

二分查找是一种高效的查找方法。它可以明显减少比较次数，提高查找效率。在一个有n个元素的有序序列中，利用二分查找大约需要log₂n次。但是，二分法查找的前提条件是被查找的数据必须是有序的。

在翻页过程中借助两个书签，划定目标所属范围，然后翻到两个书签的中间位置。每次目标区域都更新为原来的“二分之一”，当数据范围缩小到只有1个数的时候肯定能得到问题的解。1000以内的页码，最多翻10次肯定能找到解。

有了翻字典的实际操作经验，我们来尝试完善下面的二分查找程序。
x=int(input("请输入要查找的数据:"))



Python 中的 `sort()` 可以用于数据排序。例如，以下语句：
`x=[4,6,2,1,5,9]`
`x.sort()`
可以将列表 `x` 按从小到大的顺序排列。



当 `while` 循环中执行 `break` 语句时，循环会马上终止。

```
step=0
flag1=1
flag2=1000
while( ):
    mid=
    if mid>x:
        #右边界前移
    elif mid<x:
        #左边界后移
    else:
        break
        #恰好找到目标数据，退出循环
print("查找次数为:",step)
#输出次数
```

如果输入的数据不在范围内，会出现什么结果呢？程序还需要在哪些地方进行完善？大家一起来试试吧。

算法研究涉及计算机科学和数学中令人着迷的话题。面对海量数据的处理，算法选择得当，解决问题时便游刃有余；算法选择不当，则可能出现程序运行错误、运行时间长或占用空间大等问题。所有这些都促使我们深入学习，感悟化繁为简的编程魅力。

任务二 玩转“汉诺塔”游戏

※ 活动 剖析问题，设计游戏策略

“汉诺塔”游戏源于一个古老的印度传说。如图4.3.1所示，木板上有A、B、C三根杆，A杆上有若干木盘，规定每次移动一个木盘，且小的木盘只能叠在大的木盘上面。请设计算法，用尽可能少的次数把所有木盘从A杆全部移到C杆上。

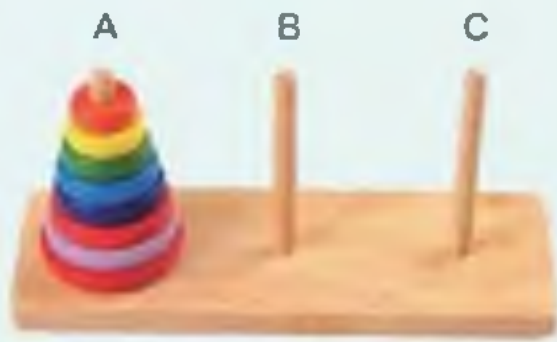


图4.3.1 汉诺塔

要使移动次数尽可能少，必须排除无效移动。现在有8个木盘，不妨先以3个木盘为例，观察一下移动的过程。请在图4.3.2中记录木盘移动的过程。

A B C	A B C 第1步: A→C	第2步:	

图4.3.2 汉诺塔的移动过程

● 递归

递归是计算科学领域中一种重要的计算思维模式。它既是一种抽象表达的手段，也是一种问题求解的重要方法。

直接或间接地调用自身的方法称为递归。可以将递归简单类比为具有自相似性重复的事物。图4.3.3所示就是递归的一种形象表示。



图4.3.3 递归图像

在数学与计算机领域中，递归函数是指用函数自身来定义该函数的方法。如著名的斐波那契数列“1, 1, 2, 3, 5, 8, 13, …”，可以递归定义为

$$F(n) = \begin{cases} 1 & (n=1 \text{ 或 } n=2) \\ F(n-1) + F(n-2) & (n>2) \end{cases}$$

递推关系是递归的重要组成，而边界条件是递归的另一要素，它保证递归能在有限次的计算后得出结果，而不会产生无限循环的情况。

面对一个大规模复杂问题的求解，递归的基本思想是把规模较大的问题层层转化为规模较小的同类问题求解。对递归而言，递推与回归，二者缺一不可。结合分治策略，递归也可用“分”“治”“合”三个字概括。

(1) 分：将原问题分解成k个子问题。

(2) 治：对这k个子问题分别求解。如果子问题的规模仍然不够小，则将其再分解为k个子问题，如此进行下去，直到问题足够小时，就很容易求出子问题的解。

(3) 合：将求出的小规模问题的解合并为一个更大规模问题的解，自下而上逐步求出原问题的解。

移动3个木盘的方法是：根据木盘叠放规则，要使A杆上最大的木盘（记为x）移动到C杆上（子问题1，如图4.3.2中的第4步），必须先把x上方的所有木盘移动到B杆上（子问题2，如图4.3.2中的前3步），然后再将B杆上所有的木盘移动到C杆上（子问题3，如图4.3.2中的后3步）。

3个木盘的移动问题成功解决了，就可以解决更多木盘的移动问题了。

将n个木盘从A杆移动到C杆，需要借助中间的B杆。只要超过一

 递归调用过程中，系统自动记录和传递参数的信息。

个木盘，在移动过程中，总会存在起始杆、过渡杆及目标杆的问题。因此，定义函数时，用到了4个参数： $\text{hanoi}(n, s, m, t)$ ， n 表示需要移动的盘子数量， s 表示盘子的起始杆， m 表示中间过渡杆， t 表示目标杆，如图4.3.4所示。

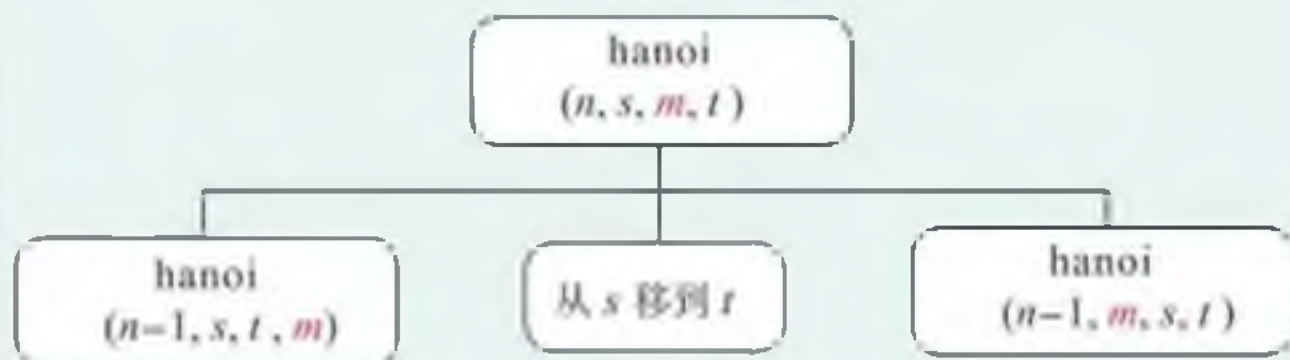


图4.3.4 汉诺塔递归过程图示

让我们根据图示一起完善程序吧。

```

def hanoi(n, s, m, t):
    # 定义一个函数，将n个木盘从s借助m移到t
    if n==1:          # 当只有一个木盘时，直接从起始杆移动到目标杆
        print(_____)
    else:
        # 将前n-1个木盘借助t从s移到m
        hanoi(_____, s, _____, _____)
        print(_____)          # 将最下面的木盘从s移到t
        # 将m上的n-1个木盘借助s移到t
        hanoi(_____, _____, _____, t)

# 主程序
n=int(input('请输入木盘的个数: '))
# 调用函数，将n个木盘从A借助B移动到C
hanoi(n, 'A', 'B', 'C')
  
```

运行程序，输入3个木盘，记录程序运行结果。

移动过程记录

将一个难以直接解决的大问题，分割成一些规模较小的同类问题，以便各个击破，分而治之，此为分治。分治与递归就像一对孪生兄弟，经常同时应用在算法设计中，并由此产生了许多高效的算法。



拓展练习

1. 结合4.2的知识，计算“汉诺塔”游戏移动的次数。
2. 尝试用二分法求解 $x^3 - x^2 + x - 1 = 0$ 。

操作提示：

令 $f(x) = x^3 - x^2 + x - 1$ ，针对有解的单调区间 (a, b) ，取 $x_0 = (a + b) / 2$ ：

若 $f(a) * f(x_0) < 0$ ，则 $f(x)$ 在 (a, x_0) 内有解；

若 $f(x_0) * f(b) < 0$ ，则 $f(x)$ 在 (x_0, b) 内有解；

若 $|f(x_0)| < 10^{-6}$ ，则 x_0 为方程的解。



拓展知识

迭代与递归的关系

迭代算法与递归算法都需要重复执行某些代码，两者既有区别又有着密切的联系。

迭代是重复反馈过程的活动，其目的通常是逼近所需目标或结果。递归是重复调用函数自身。递归中，遇到满足终止条件的情况时逐层返回。迭代则通常使用计数器结束循环。

迭代程序可以转换成等价的递归程序。以上一节中计算斐波那契数列第 n 项的值为例，程序间的转换如表4.3.2所示。

表4.3.2 递归与迭代

递归	迭代
<pre>def fib1(n): #递归求Fibonacci数列第n个数 if n==1 or n==2: return 1 else: return fib1(n-1)+fib1(n-2)</pre>	<pre>def fib2(n): #迭代求Fibonacci数列第n个数 f1=f2=1 for i in range(3,n+1): f1,f2=f2,f1+f2 return f2</pre>

4.4 综合问题的解决

在解决一个综合问题时，我们通常先考虑总体，后考虑细节：先面向整体，再细化局部。面对软件开发这类综合问题时，需要立足对象间的相互联系，强调便捷的人机交互模式，尽量向需求靠拢。



学习目标

- ★ 学会分析任务，描述需求分析，知道需要解决的关键问题。
- ★ 体验软件开发的基本流程。

春节是我国一个古老的节日，也是全年最重要的节日。“迎福”“纳福”是迎新春必不可少的传统项目。有道是：“大福小福全家福，有福享福处处福。知福来福有祝福，清福洪福添幸福。接福纳福年年福，守福祈福岁岁福！”

本节我们将围绕“‘接福’游戏”项目，创设游戏情境，实现通过单击鼠标，去接屏幕上不断出现的福字，并根据成功接到福字的次数记录分值。

本项目主要包含“基于需求，分解任务”和“组装程序，测试运行”两个任务。



任务一 基于需求，分解任务

“接福”游戏界面如图4.4.1所示。



图4.4.1 “接福”游戏界面

※ 活动1 了解需求，分析问题

打开教科书配套资源中的“接福.py”程序并运行，单击屏幕上不断出现的福字，观察屏幕变化。在下框中写出这款游戏的功能。



需求分析是软件工程中的一个关键过程。需求分析阶段的任务是确定软件系统的功能。通俗地说，就是确定需要计算机“做什么”，想达到什么样的效果。只有在确定了这些需求后，才能够分析和寻求系统的解决方法。

“接福”游戏描述

综合考虑游戏功能，需要导入的模块有pygame、random、sys。其中，pygame模块提供了图像、声音等函数；random模块用于产生随机数；sys模块包含了跟Python环境相关的函数。

● pygame模块

pygame是一个专门用来开发游戏的模块，可以包含图像、声音等。本项目中涉及的pygame相关内容如表4.4.1所示。

表4.4.1 pygame相关内容

模块	功能
pygame.display	访问显示设备
pygame.event	管理事件
pygame.font	使用字体
pygame.image	加载和存储图片

※ 活动2 分解任务，设计流程

本项目中设计的重点和难点是人机交互，即如何实时获得鼠标的状态并对鼠标单击做出正确的响应。也就是不断地进行监听、更新游戏状态、重绘屏幕三个步骤的操作。

通常情况下，游戏的基本结构如图4.4.2所示。其中的事件主要指鼠标事件、键盘事件等。



图4.4.2 游戏循环

游戏状态可以理解为程序中的变量值。随着鼠标单击，游戏中的分数需要更新，还有诸如游戏中人物的生命值、怪物数量等，这些状态经常随着鼠标单击或键盘输入等事件的发生而改变。

随着游戏状态更新或游戏场景变化，屏幕上显示的内容也会相应地发生改变。本游戏中福字在窗口随机出现，实际上就是显示不同位置的福字。要达到这个效果，需要完成两个步骤：（1）在新的位置上画出图像；（2）把原来的图像擦掉。计算机图像可不是铅笔画，用橡皮擦就能擦掉。要“擦掉”某个图像，可以选择“覆盖”它，通常采用重绘背景或者用其他对象来覆盖的方式。在该游戏中，我们通过重绘背景来覆盖前一个福字。

最后一个问题是，如何实现这种监听？

游戏状态通常随事件的发生而改变。例如鼠标单击、键盘输入或者时间的流逝。游戏循环不停地检查是否有新事件产生，并且会根据事件来更新游戏状态，这通常叫作事件处理。

pygame是专为游戏设计的，自带了一个监听类循环，不断检查用户的操作，比如用户按键、移动鼠标或者关闭窗口等。这个循环会在程序运行期间持续工作。其代码框架如下。

在Python中，True(真)、False(假)的第一个字符要大写。

```
while True:
    #监听所有事件，获取事件列表
    for event in pygame.event.get():
        if event.type==pygame.QUIT:      #Quit事件
            sys.exit()                  #退出程序
        #鼠标单击事件
        if event.type==pygame.MOUSEBUTTONDOWN:
            .....
```

看来要获取按键的情况，只要在监听事件中增加相应的判断语句就可以了。

以下是本游戏中要用到的程序功能块，包括游戏环境设置、绘制图像、设置初始值等。请将主程序的各功能语句放到合适的位置，完善图4.4.3所示的程序流程图。

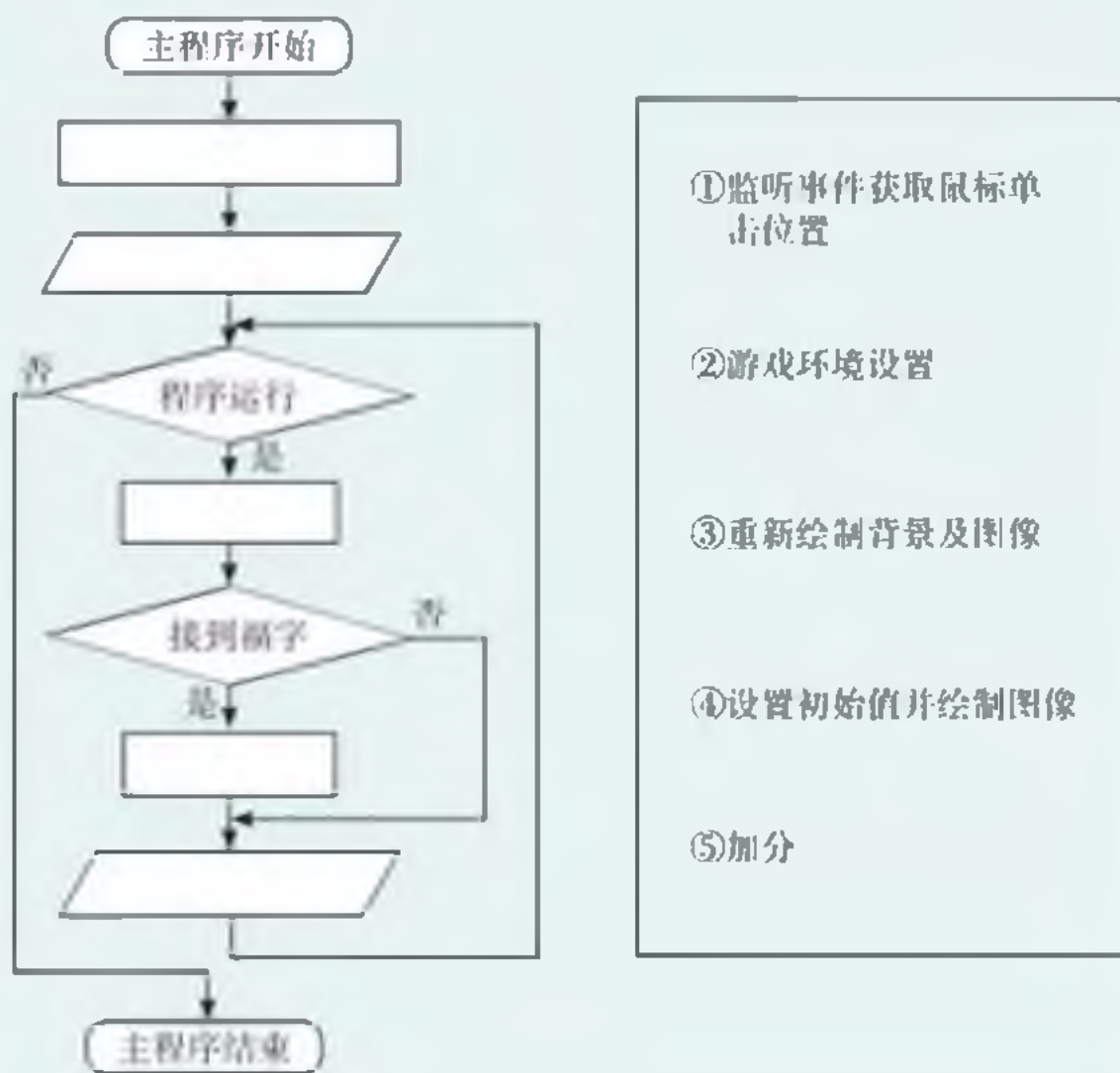


图4.4.3 流程组装

需求分析之后、程序设计之前，需要对系统进行总体设计和详细设计。总体设计就是在需求分析的基础上对模型细化、分解任务、明确程序由哪些模块组成。概括地说，就是系统应该如何实现。详细设计主要指界面设计、过程设计等。



任务二 组装程序，测试运行

※ 活动1 界面设计

1. 创建程序窗口，设置背景

调用pygame.display.set_mode()函数，将返回一个窗口。如果想在窗口上绘制或是放置对象，需要指定这个对象在窗口的具体位置，即坐标。pygame中的坐标系原点在左上角，水平方向为x轴，垂直方向

为y轴。

在pygame中，使用图像的最简单方法是调用image函数。通过pygame.image.load(filename)来载入图片，filename表示图像文件的路径和名称。图像文件必须和程序在同一文件夹下才可以使用相对路径的引用方式。

以下是一个样例程序。

```
import pygame                                #导入模块
pygame.init()                                #pygame模块初始化
#创建一个窗口screen，大小为800*600
screen=pygame.display.set_mode([800,600])
pygame.display.set_caption('实例')          #设置屏幕标题
screen.fill((144,238,144))                  #窗口填充淡绿色
img=pygame.image.load('ball.png')           #载入ball.png图像到内存
#将图像绘制在screen中，图像的左上角在[200,200]
screen.blit(img,[200,200])
pygame.display.update()                      #窗口刷新
```

运行结果如图4.4.4所示。



图4.4.4 创建实例窗口

你学会创建窗口了吗？仿照上面的代码尝试创建“接福”程序窗口，并设置图片作为背景。请注意图片文件路径的描述，思考背景图片应该在什么位置显示，有了背景图片窗口是否还要填充颜色等。画完以后一定记得利用update更新一下，否则画面会一片漆黑。

```
#创建一个窗口screen，大小为800*600
screen=_____
_____                                #设置窗口标题为“接福”
back=_____('bj.jpg')                #加载背景图
screen.blit(_____,[0,0])               #将背景图画在窗口[0,0]位置
_____                                #刷新画面
```



Python中使用RGB系统，即（红、绿、蓝），由一个包含3个0~255之间的整数列表来给出颜色。（255,0,0）就是红色。

测试代码，可得到一个带有背景图的窗口。

2. 绘制福字。

在屏幕上出现福字同样可以使用以上在屏幕上绘制图像的方法。为了便于游戏循环中反复调用，可以定义成一个函数showfu，在绘制福字时，需要设置图像显示的位置，因此showfu中有两个参数，分别表示图像位置的横坐标和纵坐标。

```
def showfu(x,y):
    #定义函数，在坐标(x,y)上显示图像
    gift=pygame.image.load('fu.png')    #加载福字图像
    screen.blit(gift,[x,y])              #显示福字
```

在实际使用中，只要提供一对坐标值，就可以调用 showfu 了。

3. 显示得分。

显示游戏得分的方法为：先定义显示字体，然后将score显示出来。为了增加代码的复用性，以便游戏循环中状态更新时反复调用，可以定义成一个函数showscore。

```
def showscore(score):
    #定义函数，显示分数
    textfont=pygame.font.SysFont('Arial',30) #字体为Arial，大小30
    #生成平滑的红色字符串
    t=textfont.render('score:'+str(score),True,(255,0,0))
    screen.blit(t,[50,50])                    #在窗口显示
```

仔细阅读代码并理解各语句的含义。

在pygame的游戏循环中，我们通过调用以下语句来完成游戏状态的更新和屏幕的重绘，请在横线上注明各语句的功能。

```
screen.blit(back,[0,0])          #_____
showscore(score)                  #_____
x=random.randint(50,700)         #_____
y=random.randint(50,500)         #_____
showfu(x,y)                      #_____
pygame.display.update()          #_____
pygame.time.delay(800)           #设置两次显示福字之间的延时
```

※ 活动2 人机交互及程序排装

在本项目中，人机交互主要体现在鼠标单击上。需要获取按键的情况，就要在监听事件中增加相应的判断语句。

打开教科书配套资源中的“接福.py”，找到相应的代码，仔细阅读并理解各语句的含义，在下面横线上用自然语言描述其对应的功能。

```
if event.type==pygame.MOUSEBUTTONDOWN:      # _____
    #获取鼠标按下的坐标并记录在mousex,mousey中
    mousex,mousey=pygame.mouse.get_pos()
    #判断鼠标是否击中福字，本例中福字宽为60，高为70
    if mousex in range(x,x+60) and mousey in range(y,y+70):
        score=score+5                        # _____
```

最后，我们可以参照流程图组装各程序块并调试运行，也可以在Python中打开教科书配套资源中的“接福.py”并运行。

虽然游戏项目已开发完成，但这款游戏还有不尽如人意的地方，如游戏难度无法调整等，请在下面的框中填写你对游戏的改进意见。

“接福”游戏改进意见



拓展练习

尝试给“接福”游戏设计一个难度策略或是给鼠标换一个手形图，以实现抓取的视觉效果，增加游戏的趣味性。难度策略可以考虑使用什么规则增减速度。尝试在你的游戏中实现自己的想法。



拓展知识

pygame键盘事件

利用键盘实现人机交互在游戏中很常见。如图4.4.5所示，这个实例是通过键盘的左、右方向键来控制小船的左右移动。

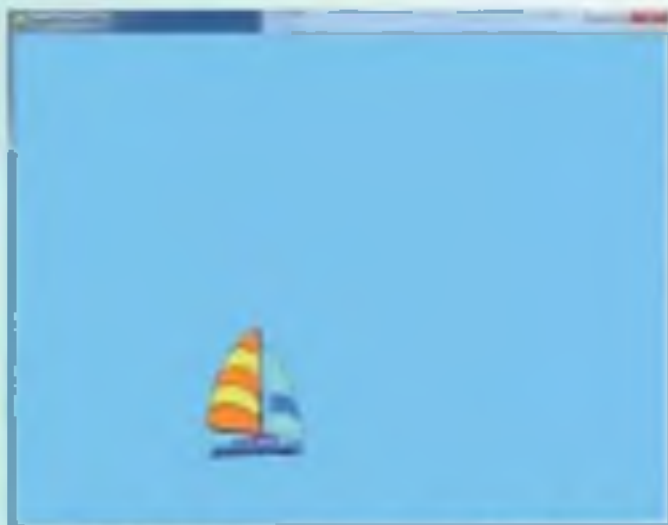


图4.4.5 利用键盘控制帆船左、右移动

```
import pygame, sys                                #导入模块
pygame.init()                                     #初始化
screen=pygame.display.set_mode([800,600])         #设置屏幕尺寸
screen.fill((135,206,250))                        #填充蓝色背景
x=100                                              #设置小船在x轴方向的初始位置
boat=pygame.image.load('boat.png')               #加载帆船图像
screen.blit(boat,[x,300])                         #在指定位置显示帆船图像
pygame.display.update()                          #更新窗口
while True:
    for event in pygame.event.get():              #监听所有事件，获取事件列表
        if event.type==pygame.QUIT:
            sys.exit()
        if event.type==pygame.KEYDOWN:           #键盘是否按下
            if event.key==pygame.K_LEFT:         #按下左箭头键
                x=x-20
            elif event.key==pygame.K_RIGHT:      #按下右箭头键
                x=x+20
        screen.fill((135,206,250))              #填充蓝色背景
        screen.blit(boat,[x,300])               #在新位置绘制帆船
        pygame.display.update()                 #更新窗口
```

单元学习评价

- (单选) 以下关于算法中输入、输出的描述正确的是()。
 - 算法可以没有输入,表示该算法不涉及任何数据信息
 - 算法可以没有输出,表示该算法运行结果为“无解”
 - 算法必须要有输入,否则算法无法进行
 - 算法至少要有有一个输出
- (单选) 采用盲目搜索的方法,在搜索的过程中,对所得的结果逐一筛选,排除不符合要求的结果,保留那些符合要求的结果,这种方法叫作()。
 - 解析法
 - 递推法
 - 枚举法
 - 选择法
- (单选) 在软件的生命周期中,明确软件系统具备哪些功能的阶段是()。
 - 可行性分析
 - 需求分析
 - 概要设计
 - 详细设计
- 有 N 根小棒(N 是偶数),每根小棒长1厘米,要将这些小棒拼成矩形,怎样拼才能使其面积最大?请编程实现,输出最大面积。

例如: $N=10$,有以下两种拼法,面积分别是4平方厘米和6平方厘米,按第2种拼法拼得的矩形面积最大。



评价要点:

- 能否针对实际问题,描述出解决问题的思路;
- 能否选择恰当的算法解决问题;
- 能否用程序设计语言实现简单的算法;
- 能否自主调试程序中出现的错误。



单元学习总结

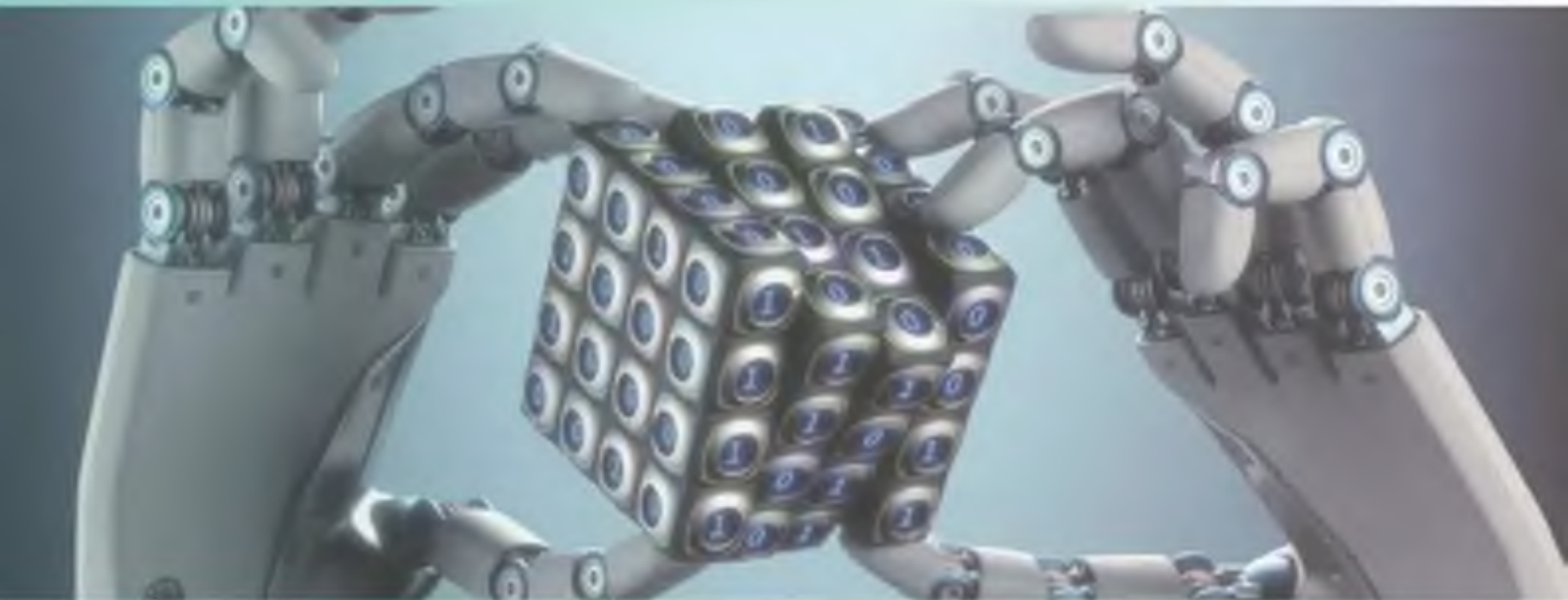


第 5 单元 数据分析与人工智能

让机器具有智能，是人类千百年来梦想。计算机技术的飞速发展，为人类的这一梦想插上了翅膀。特别是在当前的信息时代，遍布生活各个角落的海量数据积累，使人工智能技术及其应用得到了迅速的发展，并广泛地渗透进我们的生活。

每一天，你都置身其中。当你打开手机或电脑，新闻客户端推送的是高度个性化的新闻，地图软件提供的是更便捷的行车路线，天气软件给出了友好的生活指数建议……

计算机技术为什么能够实现“智能”？大数据在其中又发挥了怎样的作用？在本单元中，我们将以若干典型应用为例，了解数据分析的常用方法，知道人工智能的概念及应用领域，体验数据分析和人工智能，感受大数据和人工智能给人类生活带来的深刻变化。



人工智能的迅速发展将深刻改变人类社会生活、改变世界。

——《国务院关于印发《新一代人工智能发展规划》的通知》（国发〔2017〕35号）

5.1 走近数据分析

信息时代的日常生活中，碎片化的数据不断产生，它们往往杂乱而无序，仿佛并没有太大的价值。数据分析就是对数据进行计算、处理，从而得出一些有意义的结论的技术。了解数据分析并具备一定的数据分析能力，是大数据时代对我们的基本要求。

在本节中，我们将结合实例，通过对比分析和平均分析的实际操作，形成对数据分析的认识，并在此基础上，初步接触大数据的概念、特点和工作机制，感受大数据应用的魅力。



学习目标

- ★ 了解数据分析的几种常用方法。
- ★ 体验对比分析和平均分析的一般分析过程。
- ★ 了解大数据的含义，认识大数据分析在信息社会的重要作用。

随着我国经济的快速发展和城市化进程的加快，城市人口和机动车辆日益增多，城市交通拥堵现象日益严重。科学地发展和管理公共交通是解决问题的有效途径之一。例如，如何合理安排公交车辆的发车频率，方便乘车人选择交通线路等，都可以通过数据分析来实现。本节我们将围绕“感受智能公交”项目展开学习。同学们将通过该项目的活动，了解数据分析的几种常用方法，感受数据分析的一般过程，体验大数据应用给生活带来的便利。

本项目主要包含“体验公交出行”和“体验实时公交”两个任务。



任务一 体验公交出行

※ 活动1 分析公交高峰期

某同学每天7:00乘坐27路车去学校，17:00左右放学回家。他觉

得每天往返途中，公交车上并不拥挤。而妈妈每天8：00乘坐49路车出发，18：00下班。她总是抱怨乘车者太多。请尝试进行数据分析。

● 数据分析的方法

数据分析主要用于现状分析、原因分析和预测分析。进行数据分析时，首先要根据分析的目标提出假设，然后选择恰当的分析方法进行分析，验证假设是否正确，继而得出相应的结论。数据分析的方法有很多，如对比分析和平均分析。

● 对比分析

对比分析是指将两个或两个以上的数据进行比较，分析它们的差异，从而揭示这些数据所代表的事物的发展变化情况和规律。对比有横向对比和纵向对比两种情况。横向对比是在类似的或同类的事物之间进行比较，而纵向对比是将相同事物的不同时期进行比较。例如，我们可以横向比较27路和49路公交线路的客流量，以分析哪一条线路更加繁忙；也可以纵向比较同一条公交线路一天内不同时间段的客流量，分析高峰期出现的时间。

● 平均分析

平均分析是指运用计算平均值的方法反映总体在一定时间、地点条件下某一数量特征的一般水平。平均分析和对比分析常结合使用，例如比较不同线路的平均客流量。

步骤一、依据活动1要求，提出假设：_____

步骤二、27路和49路公交线路的月平均分时段客流量如表5.1.1所示，表中“6”表示时段6：00—7：00，其他以此类推。请对比分析表中数据。

表5.1.1 27路和49路公交线路客流量表

时间段	27路	49路	时间段	27路	49路
6	45	61	15	58	98
7	120	170	16	70	102
8	160	211	17	120	125

续表

时间段	27路	49路	时间段	27路	49路
9	143	149	18	121	123
10	53	122	19	65	57
11	72	99	20	50	30
12	59	65	21	30	25
13	58	100	22	16	13
14	56	96			

步骤三，你的结论是：_____

● 数据可视化表达

以图形、图像和动画等方式更加直观生动地呈现数据及数据分析结果，揭示数据之间的关系、趋势和规律等的表达方式称为数据可视化表达。数据可视化的应用范围广泛，如图 5.1.1 所示的第五届世界互联网大会关键词云和图 5.1.2 所示的手机记录的健身数据。

词云通过对网络文本中出现频率较高的“关键词”予以视觉上的突出，可使浏览者快速了解核心内容。



图5.1.1 第五届世界互联网大会关键词云 图5.1.2 手机记录的健身数据

图表是最常用的数据可视化表达方式之一。基本图表类型，如柱状图、饼图和折线图等，利用一般的表格加工软件即可绘制。如需创建表现形式更为丰富或者具有互动功能的图表，则必须借助专业性工具。

绘制如图 5.1.3 所示的折线图，进一步验证假设，并分析两条公交线路的高峰期出现时间和特点。

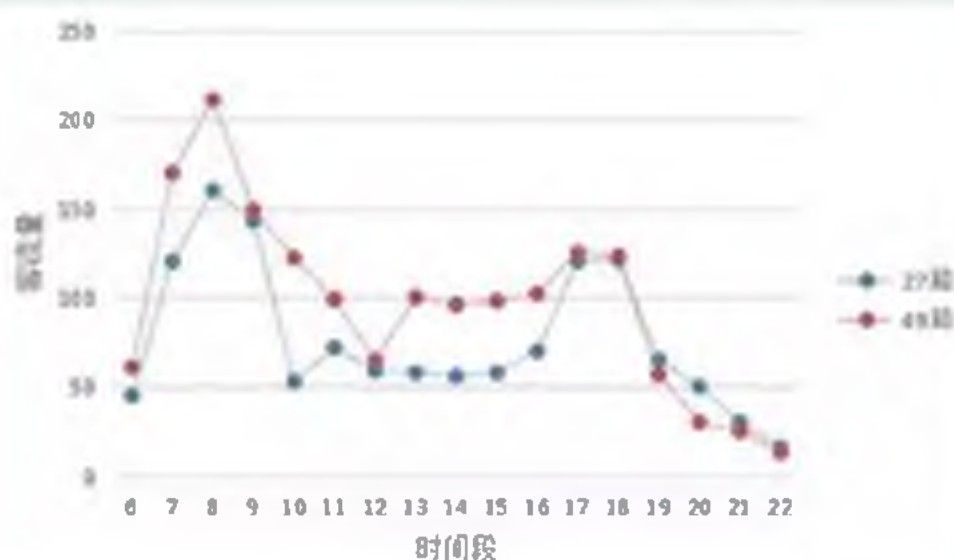


图5.1.3 27路和49路公交车线路客流图

※ 活动2 完成分析报告

数据分析报告是对整个数据分析过程的一个总结与呈现。请通过数据分析报告，将活动1中所做数据分析的起因、过程、结果及建议完整地呈现出来。

● 数据分析报告

数据分析报告是项目研究结果的展示，也是数据分析结论的有效承载形式。通过报告不仅把数据分析的起因、过程、结果及建议完整地展现出来，还可以为决策者提供科学、严谨的决策依据。例如，中国互联网络信息中心每半年发布一次的《中国互联网络发展状况统计报告》即是一份典型的反映我国互联网发展的数据分析报告。

在数据分析报告中，首先要明确数据分析的目的和背景，阐述目前存在的问题及通过分析希望解决的问题；其次需要描述数据来源和数据分析的思路、方法和模型；最后要重点呈现数据分析的过程、结论和建议。

信息时代的社会生活中，每天都在产生大量的数据，这些数据也在改变着我们的生活。让我们一起来感受数据给出行带来的变化。



任务二 体验实时公交

※ 活动1 我在哪里

假如来到一个陌生的地方，需要知道自己所处的具体位置，手机里的地图软件可以很方便地帮助你定位、计算并推荐去目的地的路线。

打开地图软件，屏幕上的蓝色标记就是使用者所处的位置，如图5.1.4所示。

很多地图软件还提供了“全景”模式。选择“查看全景”可以展示实地拍摄的“全景”照片，如图5.1.5所示。在照片上滑动手指，可以多角度查看场景。点击移动方向箭头，场景则根据位置改变。



图5.1.4 地图软件界面



图5.1.5 “全景”模式

让我们估算一下全景模式下照片数据的“体量”。根据全景照片的拍摄规则，一个拍摄点需要多角度拍摄数张照片进行拼接。如果一个拍摄点需要拍摄8张照片，每张照片的大小约为4MB，1个拍摄点的照片数据量是多少？考虑到地图的连续性，如果相邻拍摄点间隔为16米，那么一条长1千米的街道需要多少张照片？数据量是多少？如果是一个城市呢？

● 大数据

大数据是以容量大、类型多、存取速度快、应用价值高为主要特征的数据集合，它正快速发展为对数量巨大、来源分散、格式多样的数据进行采集、存储和关联分析，从中发现新知识、创造新价值、提升新能力的新一代信息技术和服务业态。

大数据的“大”是一个相对概念，没有具体标准。如果一定要定一个标准，那么10TB~100TB通常称为大数据的门槛。实际应用中，很多企业级用户把多个数据集放在一起，已经形成了PB级的数据量。而且随着数据来源的多样化，数据的类型也更加复杂，如网络日志、音频、视频、图片和地理位置信息等。大数据的意义在于，我们有可能从如此庞杂的数据中挖掘出有价值的信息，并运用于管理、农业、金融、医疗和教育等各个社会领域，为社会发展服务。



1TB=1024GB

1PB=1024TB

※ 活动2 公交车距离我多远

在地图软件中输入起点和终点，点击搜索按钮就可以获得所需的公交方案。但是在公交站台等车，久等不来总令人焦急。实时公交信息查询软件则可以实时反馈公交车辆的运行信息，提升乘客的出行效率，如图5.1.6所示。



图5.1.6 实时公交信息查询软件

● 大数据分析的应用

面对海量的数据，为了搜索、处理、分析、归纳和总结其深层次的规律，大数据分析应运而生。大数据分析是指对规模巨大的数据进行分析。它的应用日益丰富，例如实时公交系统根据用户当前位置，迅速定位最近的公交站点，查找系统中公交车辆的位置数据，选出经过该站点所有线路车辆的即将到站信息，使用户可以直观地在手机上查看车辆实时位置。用户和车辆的位置信息通常通过卫星定位系统、Wi-Fi热点和基站的位置确定。

下载一款实时公交信息查询软件，使用后填写表5.1.2，感受大数据分析在支持实时公交方面的应用。

表5.1.2 实时公交信息查询软件使用体验

序号	体验并思考的内容	结果
1	查看该软件为用户提供了哪些实用功能。	
2	如果有条件，乘公交车出行时使用该软件，并回答以下问题：①它所提供的公交车运行信息是否及时、准确？②它是否可以提供多种出行方案？	
3	定位公交车位置信息需要哪些技术？它们各有什么优缺点？	

大数据分析带来价值的同时，也在隐私保护、安全问题等方面带来危机。因此，合理、健康地使用大数据，遵循“数据道德”是我们的必备品质。



拓展练习

1. 调查分析：根据数据分析提出合理化建议。

某市第一期建设了46个公共自行车站点，主要布局在居民生活工作区、公交车站附近等地。1240个锁车柱已经全部建成，1000辆自行车也已经配备到位。在使用一段时间之后，相关管理部门对公共自行车使用数据进行采集（数据表见教科书配套资源中的“某市公共自行车一月份运营数据.xls”），同时对使用者进行访谈。受访者均表达了公共自行车方便了自己的出行，但也提出了很多问题，如自行车损坏严重、借不到车，以及还车桩满等问题。

请阅读整理相关文字和表格数据资料；对资料进行分析，发现该市公共自行车使用中存在的问题，为二期工程建设提出合理化建议，完成《某市公共自行车使用情况调查分析报告》。

2. 案例分析：根据大数据发现人们的出行规律。

在美国旧金山举行的KDD2016学术会议上，罗格斯大学的熊辉等学者向大家报告了他们的研究论文。论文指出，北京在2014年的前9个月就有350名扒手在地铁交通系统上被抓，490名扒手在公交车上被抓。因此，他们认为可以通过对北京智能公交一卡通数据进行分析研究，根据异常交通记录分析来抓小偷。如要实现这个功能，要调用哪些数据呢？同学们也可以上网搜索相关资料，了解大数据专家是如何分析的。

拓展知识

大数据分析的流程

因为大数据的量大到不能再使用常规的方法进行存储和处理，所以大数据分析与普通的数据分析也有所区别。大数据分析的一般流程如图 5.1.7 所示。

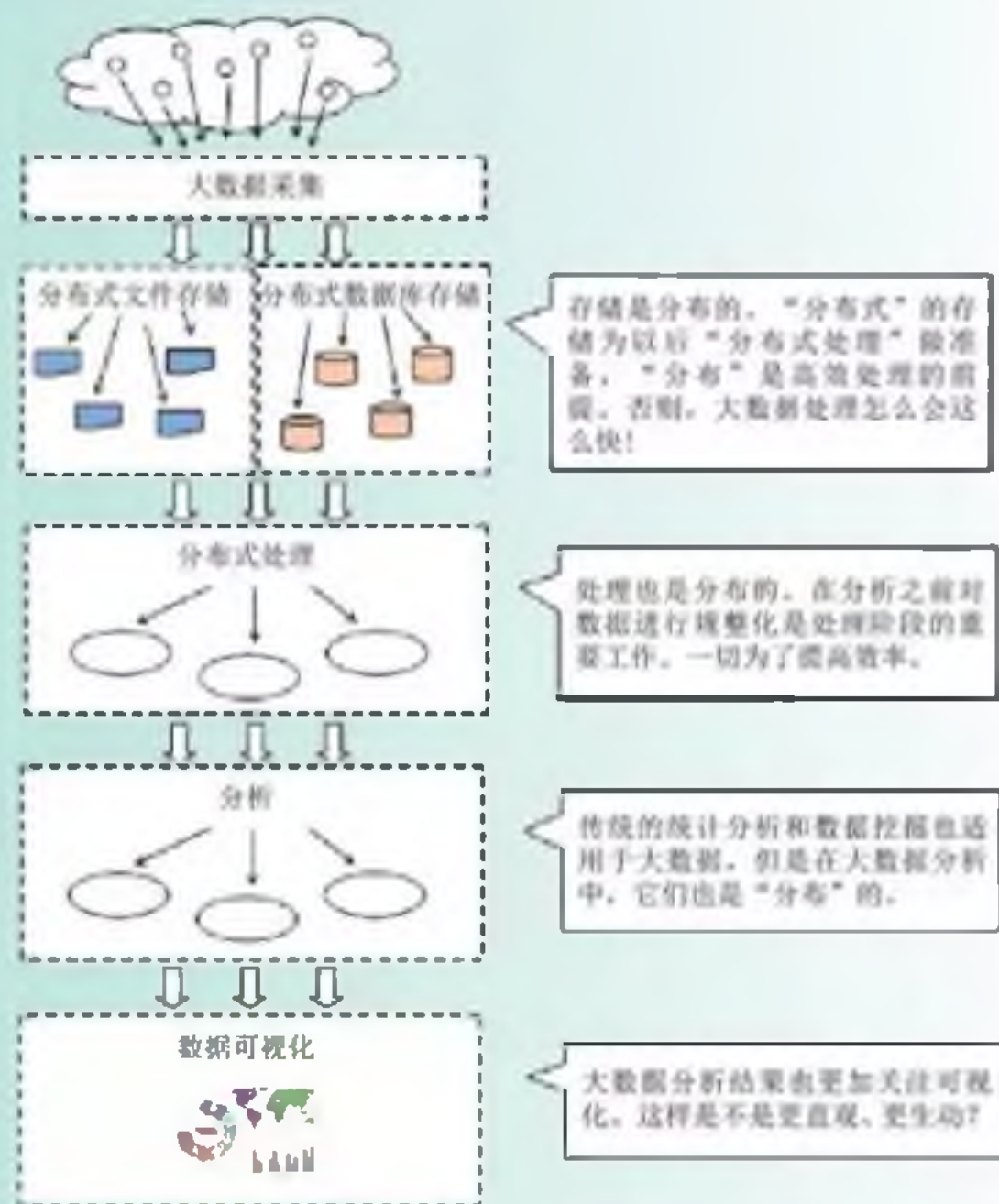


图 5.1.7 大数据分析的一般流程

5.2 探秘人工智能

人工智能作为国际竞争的新焦点，是引领未来的战略性技术，世界主要发达国家把发展人工智能作为提升国家竞争力、维护国家安全的重大战略。谁能引领人工智能，谁就掌控了人类的未来！在本节中，我们将通过对人工智能典型案例的剖析，了解智能信息处理的巨大进步和应用潜力，认识人工智能在信息社会中的重要作用。



学习目标

- ★ 了解人工智能技术的相关概念与应用领域。
- ★ 了解人工智能技术发展的新趋势。
- ★ 认识人工智能在信息社会中的重要作用。

新闻业正在智能化，医学正在智能化，甚至法律也正在智能化。在科技进步的竞赛中，我们听到了人工智能飞奔的声音。它可以诊断癌症，创造艺术，在很短的时间里从最复杂的游戏“学会”并挑战人类三千年传承的战略。本节我们将围绕“人工智能科普营”项目揭开人工智能的“隐身衣”，了解其核心技术及价值。

本项目主要包含“认识人工智能”和“揭秘智能算法”两个任务。



任务一 认识人工智能

“阿尔法围棋”的横空出世激发了全社会研究的热情。恰逢学校“人工智能科普营”活动，请你作为志愿者参与其中，协助搜集信息，并为活动出谋划策。

※ 活动1 探问最强大脑

回顾人类智能与人工智能的几次对战，可以感受到人工智能技术

的飞速发展。请搜集相关信息，并完善表5.2.1。

表5.2.1 人机对战历史

年份	机器	对战实录
1996	深蓝（Deep Blue）	战胜国际象棋世界冠军加里·卡斯帕罗夫（Garry Kasparov）
2011	沃森（IBM Watson）	在综艺节目《危险边缘》中战胜了最高奖金得主和连胜纪录保持者
2016	阿尔法围棋（AlphaGo）	战胜世界围棋冠军李世石
2017	准星数学高考机器人（AI-MATHS）	
2017	Aidam智能教育机器人	
2018		

在“深蓝”系列计算机中，存放了包括卡斯帕罗夫的棋谱在内的近百年棋谱记录，它的“智能”主要体现在对海量的实战棋谱的启发式搜索上。

● 启发式搜索

根据问题的实际，不断寻找可利用的知识，构造一条推理路线解决问题，这个过程就是搜索。按预定的控制策略进行搜索，而不考虑问题本身特性的搜索，称为盲目搜索，这种搜索一般适用于求解比较简单的问题。另一种搜索方式称为启发式搜索，它在搜索过程中加入估价函数等启发信息，不断自动调整搜索方向，加速求解进程。

以八数码问题为例，寻找从初始局面到目标局面移动方案的过程就是一次搜索，控制策略不同，搜索到的解决方案移动的步数也各不相同。某八数码问题的算法平台如图5.2.1所示。



八数码问题

在一个 3*3 的方棋盘上放上 1、2、3、4、5、6、7、8 八个数码，每个数码占一格，且有一个空格。这些数码可以在棋盘上由初始局面向目标局面移动，其移动规则是：与空格相邻的数码方格可以移入空格。



图5.2.1 八数码问题

路径规划也是启发式搜索的一种典型应用，如卫星定位、无人驾驶汽车等。请设计一份迷宫图，并表述自己的启发式搜索方案，看看谁的方案既快又准。

迷宫图：

我的启发式搜索方案：

※ 活动2 体验人工智能

有些同学提出了这样的问题：“计算机真的能像人类一样思考吗？何谓人工智能？”作为前沿科技，人工智能的内涵在不停地变化。

● 人工智能

1950年的“图灵测试”揭开了人工智能的序幕。一般而言，人工智能是利用数字计算机或者数字计算机控制的机器模拟、延伸和扩展人的智能，感知环境，获取知识并使用知识获得最佳结果的理论、方法、技术及应用系统。

有人把人工智能分成弱人工智能和强人工智能。弱人工智能一般指实现特定功能的专用智能设备，不能真正实现推理和解决问题。强人工智能是指真正能思考、有知觉、有自我意识的人类级别的智能机器。

自动驾驶、车牌识别、刷脸支付以及深入人心的智能机器人，这些生活中常见的技术都属于人工智能的不同应用。人工智能已快速地介入我们的衣食住行，刷新了我们的生活方式，它就在这里，唾手可得。

人工智能到底有哪些应用呢？请选择2~3个领域了解人工智能的应用现状，填写表5.2.2，并与同学交流讨论。

表5.2.2 人工智能的应用

功能	人工智能的体现形式
儿童及病人陪伴	健康伴侣机器人、聊天机器人、家政机器人、智能家居
搬家公司重体力劳动	搬家机器人、
环境测试	
虚拟现实、增强现实、混合现实	
自动驾驶	
工业制作	
无人机救援	

“人工智能科普营”举办了很多场专家讲座，学校希望将专家讲座整理成文字稿。人工智能可以帮助我们解决这个问题吗？图5.2.2所示是百度人工智能开放平台提供的部分人工智能应用服务，找找看，哪个应用可以解决这个问题？

语音技术	图像识别	视频内容分析	文本图像
语音识别	图像技术	视频封面识别	机器验证
语音识别	文字识别	视频比对检索	机器验证
快速语音识别	图像识别	视频内容审核	大数据处理
连续语音识别	压缩图像	AR与VR	大数据分析
语音合成	人脸与人体识别	增强现实	统计与推荐
语音唤醒	人脸识别	虚拟现实	大数据风控
智能呼叫中心	人体分析	自然语言处理	知识图谱
实时语音识别	视频技术	语言处理基础技术	

图5.2.2 百度人工智能开放平台的部分应用

语音识别可以轻松地将语音音频转成文字。录制一段1分钟以内的语音，下载并使用语音识别APP，将这段音频转成文字，并对其识别的准确性进行评价。



任务二 揭秘智能算法

※ 活动1 初探关键算法

回顾任务一所说的人机对战，那些具有“智慧”的机器能战胜人类，是因为它们的“脑”中存储了大量的优秀“算法”。闪亮登场的“阿尔法围棋”蕴藏了哪些算法呢？新华网提供的资料如图5.2.3所示。

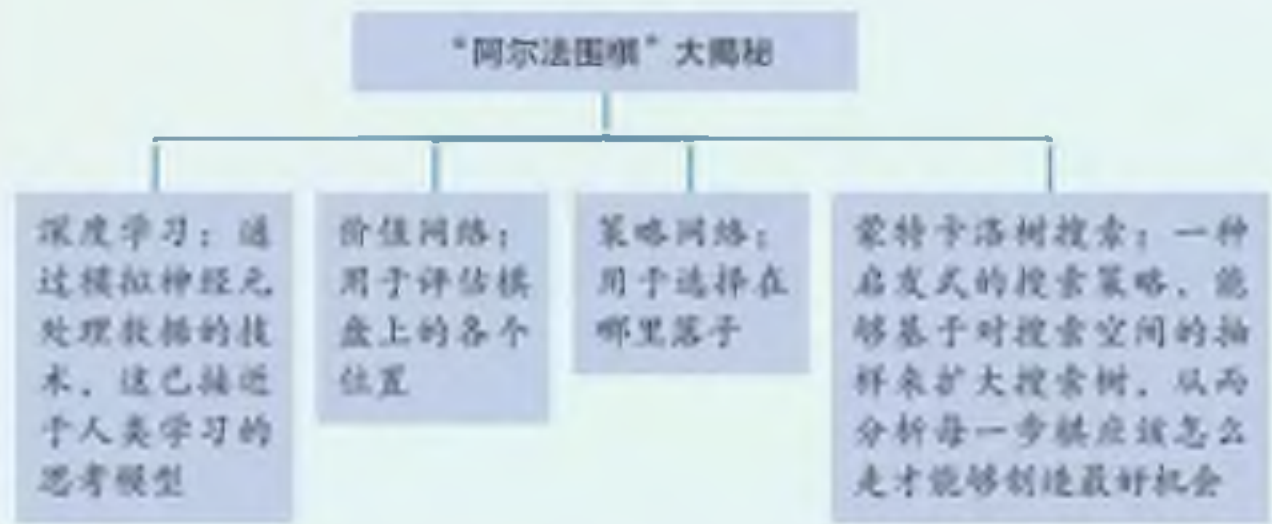


图5.2.3 “阿尔法围棋”的关键技术

“阿尔法围棋”的核心系统属于基于神经网络的深度学习，即模拟人脑神经网络，通过大量数据分析学习了三千万盘职业棋手的棋谱，再通过增强学习的方法自我博弈，寻找比基础棋谱更多的打点来击败对手。

● 机器学习

机器学习是当前人工智能的核心技术之一，目的是使计算机能模拟或实现人类的学习行为，获取新的知识或技能，重新组织已有的知识结构，从而不断完善自身的性能。

打开教科书配套资源中的“plane.exe”，体验智能飞机躲避导弹的过程，近距离地观察机器学习。程序开始时，首先繁殖一些懵懂的个体，如图5.2.4所示。正如仿达尔文进化论，它们在自我繁衍的过程中优胜劣汰，经过几百代之后，飞机就可以轻松地躲避导弹进攻，生存时间明显改观，如图5.2.5所示。

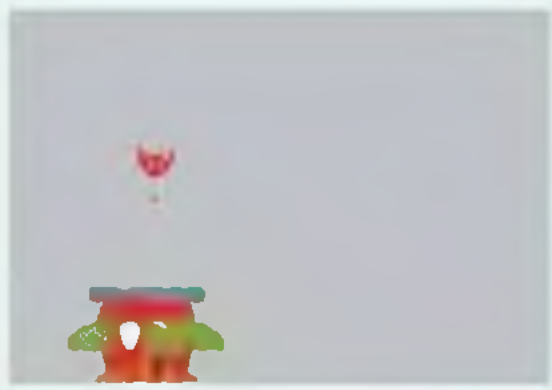


图5.2.4 飞机被导弹击中

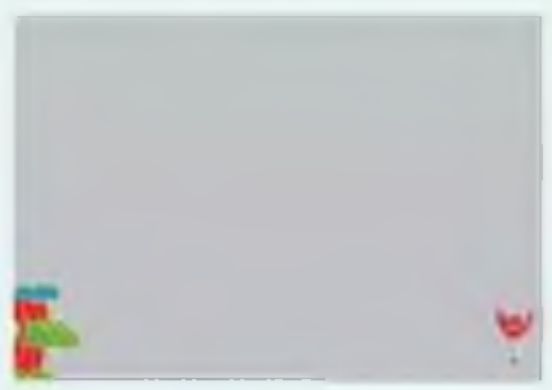


图5.2.5 飞机可以躲避导弹

人工智能、机器学习、深度学习之间的关系如图5.2.6所示。

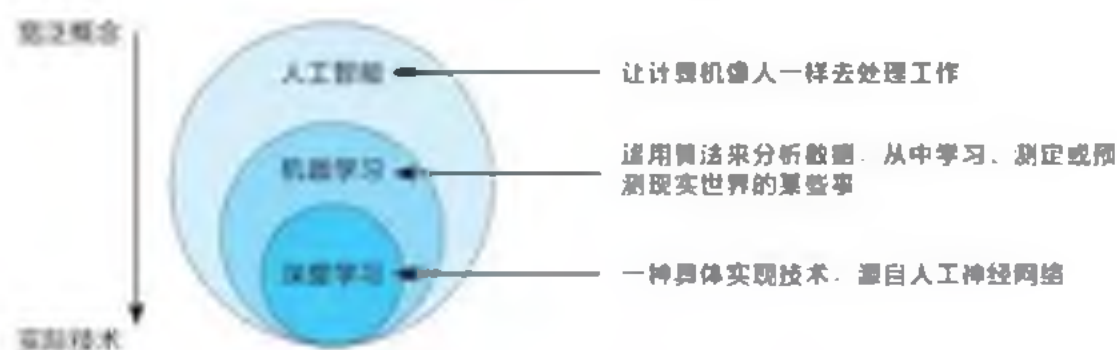


图5.2.6 人工智能、机器学习、深度学习的关系

● 神经网络

神经网络是目前人工智能领域的研究热点之一，是一种模仿动物神经网络行为特征，进行并行信息处理的算法模型。人工神经网络无须事先确定反映输入、输出之间映射关系的数学方程，它通过自身的训练，学习某种规则，最终在给定输入时得到最接近期望输出的结果。BP（Back Propagation）神经网络是目前应用最广泛的算法，它是一种按照误差逆向传播算法训练的多层前馈神经网络，其结构大致如图5.2.7所示。



图5.2.7 神经网络结构

算法由信息的正向传递与误差的反向传播两部分组成，它们由可修正的权值互连。在正向传播过程中，输入信息从输入层经隐藏层逐层计算传向输出层，当前层神经元的状态只影响下一层神经元的状态。如果输出层没有得到期望的输出，则计算输出层的误差变化值，然后转向反向传播，通过网络将误差信号沿原来的连接通路反传回来，修改各层神经元的权值直至达到期望目标。

让我们搜索“好玩的神经网络”，通过简单实例进一步了解神经网络吧。

结合信息感知、信息表示与形成、智能推理、智能决策、智能执行与信息输出的一般过程，人工智能领域的关键技术目前主要包括机器学习、知识图谱、自然语言处理、计算机视觉、人机交互、生物特征识别、虚拟现实与增强现实等。

※ 活动2 探寻强大后盾

在研究各类人工智能软件时，我们经常抱怨“电脑太慢了！”“手机内存太小了”，我们总会感觉机器性能跟不上软件需求。智能

算法对硬件的依赖到底有多大？

人工智能的核心是算法，基础是数据，本质是计算。在一辆无人驾驶汽车中，一般大约有12TB的数据，这些无人驾驶汽车背后的支撑就是快速的计算能力。要实现弱人工智能向强人工智能的跨越，必须提高计算机处理速度，使计算机变得更智能。

● 神经元芯片

目前通用型处理器的存储和运算处理分离，因此需要大量读写运行操作的深度神经网络，在此过程中不可避免地会受到传输数据带宽的制约，从而效率低下。2010年，训练一个识别猫脸的深度学习神经网络使用1.6万个处理器运行了7天，而“阿尔法围棋”则使用了更多的处理器。未来人工智能若想实现像人脑一样的千亿个神经元网络，则需要海量机器来完成。

2016年中国科学院计算技术研究所发布了寒武纪神经元芯片，用硬件直接“绘制”大脑的结构。它采用专门的硬件神经元，并将几百万个神经元连接在一起，使每个神经元都能通过位数众多的路径向其他的神经元发送信息。实验表明，该芯片的速度是普通中央处理器（CPU）的1000倍。

请同学们上网搜索目前研发成功的神经元芯片，填写表5.2.3。

表5.2.3 神经元芯片

公司名称	神经元芯片的名称	特点
英特尔	Loihi	

● 智能传感器

智能传感器是具有信息处理功能的传感器。智能传感器带有微处理机，具备采集、处理、交换信息等功能，是传感器集成化与微处理机相结合的产物。智能传感器属于人工智能的神经末梢，用于全面感知外界环境。各类传感器的大规模部署和应用为实现人工智能创造了不可或缺的条件，如智能隐形眼镜、葡萄糖手表等。未来，高敏度、高精度、高可靠性、微型化、集成化将成为智能传感器发展的重要趋势。

了解了这么多人工智能的知识，让我们设计一期“人工智能科普营”的展板吧！

展板内容规划



拓展练习

人工智能已经从一个科幻小说式的探索领域，发展成一个蓬勃发展的科技领域，同时也带来了新一轮的恐慌：人类是否会被人工智能取代？

“人工智能科普营”正在举行一场辩论赛：人类以后会被人工智能所取代吗？根据你的理解和掌握的资料给出你的观点。

资料1：电气和电子工程师协会（IEEE）最新宣布的三项人工智能伦理标准

资料2：社会和公众必须遵循一些人工智能共识原则

资料3：个人信息的合理使用

资料4：个人隐私的保护

.....

你的观点：



拓展知识

人工智能基础理论体系

2017年7月20日,《国务院关于印发〈新一代人工智能发展规划〉的通知》(国发〔2017〕35号)中明确了新一代人工智能的基础理论体系。

1. 大数据智能理论。研究数据驱动与知识引导相结合的人工智能新方法、以自然语言理解和图像图形为核心的认知计算理论和方法、综合深度推理与创意人工智能理论与方法、非完全信息下智能决策基础理论与框架、数据驱动的通用人工智能数学模型与理论等。

2. 跨媒体感知计算理论。研究超越人类视觉能力的感知获取、面向真实世界的主动视觉感知及计算、自然声学场景的听知觉感知及计算、自然交互环境的言语感知及计算、面向异步序列的类人感知及计算、面向媒体智能感知的自主学习、城市全维度智能感知推理引擎。

3. 混合增强智能理论。研究“人在回路”的混合增强智能、人机智能共生的行为增强与脑机协同、机器直觉推理与因果模型、联想记忆模型与知识演化方法、复杂数据和任务的混合增强智能学习方法、云机器人协同计算方法、真实世界环境下的情境理解及人机群组协同。

4. 群体智能理论。研究群体智能结构理论与组织方法、群体智能激励机制与涌现机理、群体智能学习理论与方法、群体智能通用计算范式与模型。

5. 自主协同控制与优化决策理论。研究面向自主无人系统的协同感知与交互、面向自主无人系统的协同控制与优化决策、知识驱动的人、机、物三元协同与互操作等理论。

6. 高级机器学习理论。研究统计学习基础理论、不确定性推理与决策、分布式学习与交互、隐私保护学习、小样本学习、深度强化学习、无监督学习、半监督学习、主动学习等学习理论和高效模型。

7. 类脑智能计算理论。研究类脑感知、类脑学习、类脑记忆机制与计算融合、类脑复杂系统、类脑控制等理论与方法。

8. 量子智能计算理论。探索脑认知的量子模式与内在机制,研究高效的量子智能模型和算法、高性能高比特的量子人工智能处理器、可与外界环境交互信息的实时量子人工智能系统等。



单元学习评价

当你在网上查阅一个数学定理、浏览一则新闻、订购一本图书时，你的年龄、爱好、IP地址等信息都将被记录下来，在你下一次使用网络时，它会预测你的需求和偏好，生活推荐、学习推荐、新闻推荐、商务推荐接踵而来。“个性化推荐系统”是互联网和电子商务发展的产物，它是建立在海量数据挖掘基础上的一种信息过滤系统，提供决策支持和信息服务。请寻找具有自动推荐系统的网站，近距离感受它的“千人千面”，分析该系统的工作流程和核心算法，讨论个性化推荐服务的优缺点，填写以下内容。

个性化推荐方案评析

网站名称：_____

网站地址：_____

个性化推荐的流程一般分三步，请用自己的语言描述：



① _____

② _____

③ _____

网站体验优点分析：

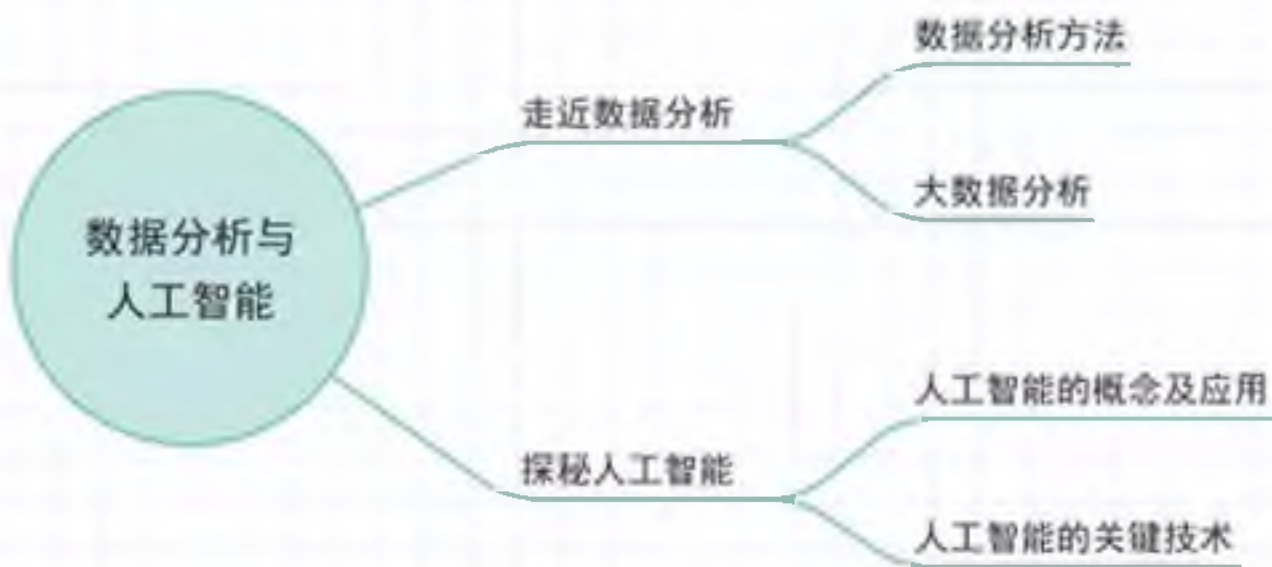
网站个性化推荐修改建议：

（请描述目前的个性化推荐有哪些不尽人意的地方，提出改进设想）

评价要点：

1. 能否了解人工智能的研究方向；
2. 能否认识大数据对智能算法的支撑；
3. 能否正确应对人工智能给予学习和生活的巨大影响。

单元学习总结



后 记

为全面落实立德树人根本任务，着力发展学生的核心素养，根据《普通高中课程方案（2017年版）》的精神，我们按照《普通高中信息技术课程标准（2017年版）》的要求对高中信息技术教科书进行了修订。

本书的修订由李艺、丁婧、白晓东、张钰、王静、荆晓虹、陈雅蓉、潘安娜、伍先军等直接参与，董玉琦、解月光、张剑平、李冬梅、张义兵等在整体设计的过程中给予指导，其间，窦万峰教授、李千日教授、王超老师审阅了本书修订稿并提出了宝贵意见。在此，我们对所有关心、支持本书编写与修订的专家、学者和老师们的表示衷心的感谢。

本书选用了一些图片和文字资料，对相关作者和出版社，我们一并表示诚挚的谢意。

编者

2019年6月