# **AP计算机科学原理：课程标准**

* **持续理解：1.1** - 创造性的开发可以是创建计算工件的重要过程。
* **学习目标：1.1.1** - 在创建计算工件时应用一个创造性的开发流程。[P2]

♦ 基本知识：1.1.1B **-** 创建计算工件通常采用迭代法和探索性的过程将想法转化为有形的形式。

♦ 基本知识：1.1.1A **-** 在开发计算机工件中可以包括创造性的过程，但不局限于采用非传统、非规范的技术；使用新奇的工件、工具和技术组合以及出于个人好奇心的探索。

* **持续理解:1.2** - 计算机应用可以使人们能够运用创造性的开发过程来创建计算工件，从而具有创造性的运算式或解决问题的方法。
* **学习目标：1.2.5** - 分析计算工件的正确性，可用性，功能性和适用性。[P4]

♦ 基本知识：1.2.5D - 计算工件的适用性（或适当性）可能与其使用或感知方式有关。

♦ 基本知识：1.2.5C - 计算工件的功能可能与其使用或感知方式有关。

♦ 基本知识：1.2.5B - 根据工件的类型，计算工件可能存在弱点、错误或误差。

♦ 基本知识： 1.2.5A - 工件中的配置指令决定了工件的正确性、可用性、功能性和适用性。

* **学习目标：1.2.4** - 在计算工件的创建中进行合作。[P6]

♦ 基本知识：1.2.4F - 合作创建的计算工件可以反映个人运算式的想法。

♦ 基本知识：1.2.4E - 在开发计算工件中合作促进了多种观点（包括社会文化观点）以及多样化人才和技能的应用。

♦ 基本知识：1.2.4D - 有效的合作策略可提高性能。

♦ 基本知识：1.2.4C - 有效的合作团队可以锻炼人际沟通,建立共识，解决冲突和谈判的能力。

♦ 基本知识：1.2.4B - 有效的合作团队会考虑使用在线协作工具。

♦ 基本知识：1.2.4A - 合作创建的计算工件可以反映出是不止一个人的努力。

* **学习目标：1.2.3** - 通过组合或修改现有的工件来创建新的计算工件。

♦ 基本知识：1.2.3C - 组合或修改现有的工件可以显示个人运算式的想法。

♦ 基本知识：1.2.3B - 计算有助于以更高的细节和精度创建和修改计算工件。

♦ 基本知识：1.2.3A - 创建计算工件可以通过组合和修改现有的工件或创建新工件来完成。

* **学习目标：1.2.2** - 使用计算工具和技术创建计算工件来解决问题。

♦ 基本知识：1.2.2B - 当传统或规定的计算技术无效时，一个创造性的计算工件开发过程可以用来解决问题。

♦ 基本知识：1.2.2A - 计算工具和技术可以加快找到解决问题方法的过程。

* **学习目标：1.2.1** - 为创造性的运算式创建计算工件。[P2]

♦ 基本知识：1.2.1E - 在计算工件中创造性的运算式可以反映个人的想法或兴趣。

♦ 基本知识：1.2.1D - 可以通过使用非传统和非规定的计算技术创建一个创造性开发的计算工件。

♦ 基本知识：1.2.1C - 用于创建计算工件的计算工具和技术，可以包括但不仅限于编程电子集成驱动器，电子表格，3D打印机或文本编辑器。

♦ 基本知识：1.2.1B - 创建计算工件需要理解和使用软件工具和服务。

♦ 基本知识：1.2.1A - 一个计算工件可以是由人类使用计算机创建的任何东西，但不仅限于程式，图像，音频，视频，演示文稿或网页文件。

* **持续理解：1.3** - 计算机应用可以扩展人类传统的运算式和体验。
* **学习目标：1.3.1** - 创造性的运算式要使用计算工具和技术。[P2]

♦ 基本知识：1.3.1E - 计算能够对真实和虚拟现象进行创造性探索。

♦ 基本知识：1.3.1D - 可以通过使用现有的软件或修改的软件来创建数字特效和动画，包括实现特效和动画的功能。

♦ 基本知识：1.3.1C - 可以通过生成像素图案，操作现有的数字图像或组合图像来创建数字图像。

♦ 基本知识：1.3.1B-可以通过合成声音，采集现有的音频和音乐，以及记录和操作声音（包括分层和循环）来创建数字音频和音乐。

♦ 基本知识：1.3.1A - 创建数字特效，图像，音频，视频和动画已经改变了行业。

* **持续理解：2.1** - 基于二进制序列构建的各种抽象数据可用于表示所有数字数据。
* **学习目标：2.1.2** - 解释如何使用二进制序列表示数字数据。[P5]

♦ 基本知识：2.1.2F - 位序列可能表示不同配置指令中的不同类型的数据。

♦ 基本知识：2.1.2E - 位序列可能表述指令或数据。

♦ 基本知识：2.1.2D - 二进制序列的译码取决于它如何使用。

♦ 基本知识：2.1.2C - 在许多编程语言中，用于表示真实数据（作为浮点数）的固定位数限制了浮点值和数字运算的范围，此限制可能导致舍入。

♦ 基本知识：2.1.2B - 在许多编程语言中，用于表示字符或整数的固定位数限制了整数值和数学运算的范围，此限制可能导致溢位或其他错误。

♦ 基本知识：2.1.2A - 使用有限表示法模拟数字的无限数学概念。

* **学习目标：2.1.1** - 描述用于表示数据的各种抽象符号。[P3]

♦ 基本知识：2.1.1G - 数字可以从任何基数转换为任何其他基数。

♦ 基本知识：2.1.1F - 十六进制（基数16）用于表示数字数据，因为十六进制比二进制数使用的数字少。

♦ 基本知识：2.1.1E - 在最低级别的抽象符号中，二进制（基数2）数据仅使用数字零和一的组合。

♦ 基本知识：2.1.1D - 数位基数用于表示和研究数字数据，包括二进制，十进制和十六进制。

♦ 基本知识：2.1.1C - 在更高级别中，位被分组以表示抽象符号，包括但不仅限于数字，字符和颜色。

♦ 基本知识：2.1.1B - 在最低级别中所有的数字数据都以位表示。

♦ 基本知识：2.1.1A - 数字数据由不同层次的抽象符号表示。

* **持续理解：2.2** - 多层次的抽象符号用于编写程式或创建其他计算工件。
* **学习目标：2.2.3** - 在编写程式时识别所使用的多层次抽象符号。[P3]

♦ 基本知识：2.2.3K - 低级别的抽象符号可以结合起来转变成更高级别的抽象符号，如短信服务（SMS）或电子邮件消息，图像，音频文件和视频。

♦ 基本知识：2.2.3J - 使用硬件，软件和概念层次抽象设计，开发和分析应用程序和系统。

♦ 基本知识：2.2.3I - 硬件是采用多层次的抽象技术构建的，如晶体管，逻辑门，芯片，内存，主板，特殊用途卡和存储设备。

♦ 基本知识：2.2.3H - 硬件组件可以像晶体管一样低电平或像视频卡那样高电平。

♦ 基本知识：2.2.3G - 芯片是由低级组件和电路排版后的具有特殊功能的抽象产品，。

♦ 基本知识：2.2.3F - 逻辑门是以布尔函数为模型的抽象硬件。

♦ 基本知识：2.2.3E - 二进制数据是由计算硬件的物理层，包括控制极，芯片和组件处理的。

♦ 基本知识：2.2.3D - 在抽象层中，更高层次的抽象（最通用的概念）将被放置在顶端，更低层次的抽象（更具体的概念）将被放置在底端。

♦ 基本知识：2.2.3C - 编程语言中的代码通常被转换成另一种（低级）语言的代码，以便在计算机上执行。

♦ 基本知识：2.2.3B - 高级编程语言可以为程序员提供更多抽象符号，并使人们更容易读写程式。

♦ 基本知识：2.2.3A - 不同的编程语言提供不同级别的抽象符号。

* **学习目标：2.2.2** - 使用多个层次的抽象符号来编写程式。[P3]

♦ 基本知识：2.2.2B - 在开发程式时意识到并使用多个抽象层次符号有助于更有效地应用可用资源和工具来解决问题。

♦ 基本知识：2.2.2A - 软件的开发采用多层抽象化，如常数，运算式，语句，函数和程式库。

* **学习目标：2.2.1** - 在编写程式或创建其他计算工件时开发抽象符号。[P2]

♦ 基本知识：2.2.1C - 抽象符号通过输入参数功能来允许软件复用。

♦ 基本知识：2.2.1B - 抽象符号从特定示例中提取共同特征，以概括概念。

♦ 基本知识：2.2.1A - 开发抽象符号的过程中涉及删除细节和对功能进行通用化。

* **持续理解：2.3** - 模型和模拟使用抽象符号来生成新的可读懂指令和知识。
* **学习目标：2.3.2** - 使用模型和模拟来制定，细化和测试假说。[P3]

♦ 基本知识：2.3.2H - 快速和大量的测试允许更改模型以精确地反映正在建立模型的对象或现象。

♦ 基本知识：2.3.2G - 模拟所需的时间受模型的细节和质量影响，也受用于模拟的软件和硬件影响。

♦ 基本知识：2.3.2F - 模拟可以促进模型的广泛和快速测试。

♦ 基本知识：2.3.2E - 模拟允许假说在没有现实世界的约束下进行测试。

♦ 基本知识：2.3.2D - 模拟结果可能会产生新的知识和正在建立模型的新假说。

♦ 基本知识：2.3.2C - 假说通过检查模型和模拟提供给对象或现象的反馈而得到改善。

♦ 基本知识：2.3.2B - 假说的制定是为了解释正在建立模型的对象或现象。

♦ 基本知识：2.3.2A - 模型和模拟可以促进在考虑之下的假说对象和现象形式化和精化。

* **学习目标：2.3.1** - 使用模型和模拟来表示现象。[P3]

♦ 基本知识：2.3.1D - 在现实世界中没有成本和危险地建造和测试现象，模拟真实世界事件。

♦ 基本知识：2.3.1C - 模型通常省略对象或现象的不必要特征。

♦ 基本知识：2.3.1B - 模型可能使用不同的抽象符号或抽象层次，具体取决于所呈现的对象或现象。

♦ 基本知识：2.3.1A - 模型和模拟是更复杂的对象或现象的简化表示。

* **持续理解：3.1** - 人们使用计算机程式来处理信息以获得洞察力和知识。
* **学习目标：3.1.3** - 通过适当的可视化，记数法和精确语言获得数字数据，以解释洞察力和知识。[P5]

♦ 基本知识：3.1.3E - 数据的交互性是交流的一个方面。

♦ 基本知识：3.1.3D - 在交流知识中获得的数据可以有效地转换信息。

♦ 基本知识：3.1.3C- 在交流洞察力和知识中获得的数字信息可以有效地计算分析汇总的数据。

♦ 基本知识：3.1.3B - 在交流洞察力和知识中获得的数据可用于表格，图表和文本显示。

♦ 基本知识：3.1.3A - 可视化工具和软件可以传达有关数据的信息。

* **学习目标：3.1.2** - 协作处理信息以获得洞察力和知识。[P6]

♦ 基本知识：3.1.2F - 当单独工作时协作研究大数据集不能获得洞察力和知识。

♦ 基本知识：3.1.2E - 面对面和使用在线工具协作可以促进处理信息以获得洞察力和知识。

♦ 基本知识：3.1.2D - 在开发假说和问题，测试假说和回答问题中协作，获得的数据有助于参与者获得洞察力和知识。

♦ 基本知识：3.1.2C - 参与者之间就数据驱动问题开展交流可以增强洞察力和知识。

♦ 基本知识：3.1.2B - 协作促进人通过应用多样的观点，经验和技术技能来解决计算问题。

♦ 基本知识：3.1.2A - 协作是解决数据驱动问题的一个重要部分。

* **学习目标：3.1.1** - 查找型样并测试关于数字处理信息的假说以获得洞察力和知识。[P4]

♦ 基本知识：3.1.1E - 当使用计算工具转换数据时，型样就会呈现出来。

♦ 基本知识：3.1.1D - 洞察力和知识可以从转换数字信息中获得。

♦ 基本知识：3.1.1C - 组合数据源，聚类数据和数据分类是使用计算机处理信息的部分过程。

♦ 基本知识：3.1.1B - 可以通过计算机处理信息来过滤和清理数字信息。

♦ 基本知识：3.1.1A - 当处理数字信息以获得洞察力和知识时，计算机通常以迭代式和交互式运行。

* **持续理解：3.2** - 计算有助于探索和发现信息中的关联。
* **学习目标：3.2.2** - 确定大数据集是如何影响计算进程的使用以发现信息和知识。[P3]

♦ 基本知识：3.2.2H - 存储，管理，传输和处理数据集的分析技术随着数据集规模的扩大而变化。

♦ 基本知识：3.2.2G - 大数据集的有效使用需要计算对策。

♦ 基本知识：3.2.2F - 存储数据的系统的容量或规模会影响数据集的使用方式。

♦ 基本知识：3.2.2E - 当数据集很大时，需要考虑系统的可扩展性是很重要的一部分。

♦ 基本知识：3.2.2D - 维护包含个人信息的大型数据集的隐私可能具有挑战性。

♦ 基本知识：3.2.2C - 为分析构建大型数据集可能具有挑战性。

♦ 基本知识：3.2.2B - 存储，处理和管理大型数据集具有挑战性。

♦ 基本知识：3.2.2A - 大型数据集包括事务处理，测量，文本，声音，图像和视频等数据。

* **学习目标：3.2.1** - 从数据中抽取信息可以发现和解释数据中的关联或趋势。[P1]

♦ 基本知识：3.2.1I - 元数据可以通过提供有关数据各方面的附加信息来增加数据或数据集的有效使用。

♦ 基本知识：3.2.1H - 元数据可以是关于图像，网页或其他复杂对象的描述性数据。

♦ 基本知识：3.2.1G - 元数据是关于数据的数据。

♦ 基本知识：3.2.1F - 软件工具包括电子表格和数据库，有助于有效组织和发现信息中的趋势。

♦ 基本知识：3.2.1E-信息过滤系统是查找信息和识别信息模式的重要工具。

♦ 基本知识：3.2.1D - 搜索工具对于有效地查找信息至关重要。

♦ 基本知识：3.2.1C - 计算工具有助于发现大型数据集内的信息连接。

♦ 基本知识：3.2.1B - 大数据集为识别趋势，建立数据连接和解决问题提供了机会。

♦ 基本知识：3.2.1A - 大数据集为提取信息和知识提供了机会和挑战。

* **持续理解：3.3** - 将信息表示为数字数据时有参数折衷。
* **学习目标：3.3.1** - 分析如何使数据表示，存储，安全性和传输涉及计算机操作信息。[P4]

♦ 基本知识：3.3.1I - 读取数据和更新数据具有不同的存储要求。

♦ 基本知识：3.3.1H - 存储媒介的选择会影响操作其中包含的数据的方法和成本。

♦ 基本知识：3.3.1G - 根据数据的性能（例如容量和预期用途），数据以多种格式存储。

♦ 基本知识：3.3.1F - 包含个人信息的数据会引发安全和隐私问题。

♦ 基本知识：3.3.1E-有损数据压缩可以减少所存储或传输的数据量，其代价是有损压缩不能恢复成原始数据，却可以和原数据非常近似。

♦ 基本知识：3.3.1D - 无损数据压缩不但可以减少所存储或传输的数据量，无损压缩后的数据也可以复原成原始数据。

♦ 基本知识：3.3.1C - 使用有损和无损压缩技术存储和传输数据的过程中存在参数折衷。

♦ 基本知识：3.3.1B - 安全性问题在存储和传输信息时会产生参数折衷。

♦ 基本知识：3.3.1A - 数字数据涉及的参数折衷与存储，安全性和隐私问题有关。

* **持续理解：4.1** - 算法是由计算机执行并使用编程语言精确指令。
* **学习目标：4.1.2** - 用语言表达算法。[P5]

♦ 基本知识：4.1.2I-用语言表达算法时，清晰性和可读性是重要的考虑因素。

♦ 基本知识：4.1.2H - 几乎所有的程序语言在表达任何算法时都是等效的。

♦ 基本知识：4.1.2G - 每个算法都可以仅使用排序，选择和迭代来构建。

♦ 基本知识：4.1.2F - 表达算法的语言可能会影响清晰度或可读性，但不影响算法解决方案是否存在。

♦ 基本知识：4.1.2E - 一些程序语言是为特定领域设计的，并且在此领域中这些程序语言可以更好地表达算法。

♦ 基本知识：4.1.2D - 不同的语言更适合表达不同的算法。

♦ 基本知识：4.1.2C - 在程序语言中描述的算法可以在计算机上执行。

♦ 基本知识：4.1.2B - 自然语言和伪码描述算法，以便人们可以理解它们。

♦ 基本知识：4.1.2A - 算法的语言包括自然语言，伪代码以及可视化和文本程序语言。

* **学习目标：4.1.1** - 在程式中为编译系统开发算法。[P2]

♦ 基本知识：4.1.1I - 开发一种新算法来解决问题可以深入了解问题。

♦ 基本知识：4.1.1H - 可以开发不同的算法来解决相同的问题。

♦ 基本知识：4.1.1G - 基本演算法知识有助于构建新的算法。

♦ 基本知识：4.1.1F - 使用现有的正确算法作为构建模块来构建新算法有助于确保新算法的正确性。

♦ 基本知识：4.1.1E - 算法可以被组合起来生成新的算法。

♦ 基本知识：4.1.1D - 迭代是反复执行算法指令，直到满足条件或达到指定次数。

♦ 基本知识：4.1.1C - 判断利用布尔函数来确定算法使用哪个部分。

♦ 基本知识：4.1.1B - 排序是应用算法的每个步骤有序地排列所给出的计算机指令。

♦ 基本知识：4.1.1A - 排序，判定和迭代是构建算法的模块。

* **持续理解：4.2** - 算法可以解决许多计算问题，但不能解决所有计算问题。
* **学习目标：4.2.4** - 分析性和经验性地评估算法的效率，正确性和明确性。[P4]

♦ 基本知识：4.2.4H - 在任何列表中搜索项目时可以使用线性搜索;二进制搜索只能在列表排序时使用。

♦ 基本知识：4.2.4G - 效率包括执行时间和内存使用量。

♦ 基本知识：4.2.4F - 为问题找到一个有效的算法可以帮助解决更大的问题。

♦ 基本知识：4.2.4E - 有时更高效的算法更复杂。

♦ 基本知识：4.2.4D - 针对相同的问题，不同的正确算法具有不同的效率。

♦ 基本知识：4.2.4C - 算法的正确性是通过形式上或数学上推导来确定的，而不是通过测试算法编译系统来确定。

♦ 基本知识：4.2.4B - 通过不同的输入数据执行和运行算法可以完成算法的实验分析。

♦ 基本知识：4.2.4A - 通过形式上或数学上推导算法来确定算法的效率。

* **学习目标：4.2.3** - 解释计算机科学中存在的不可判定的问题。[P1]

♦ 基本知识：4.2.3C - 一个不可判定的问题不可以建立算法，因为它总是对输入数据回答“是”或没有答案。

♦ 基本知识：4.2.3B - 一个可判定的问题是可以建立一个算法对所有的输入数据回答“是”或“否”（例如，“甚至是数字”）。

♦ 基本知识：4.2.3A - 一个不可判定的问题可能有一个算法解决方案，但是没有一个算法解决方案可以解决所有问题。

* **学习目标：4.2.2** - 解释计算机科学中可解和不可解问题的区别。[P1]

♦ 基本知识：4.2.2D - 一些问题使用任何算法都无法解决。

♦ 基本知识：4.2.2C - 一些最佳化的问题，比如“找到最好的”或“找到最小的”不能在合理的时间内解决，但是可以有近似的选择方案。

♦ 基本知识：4.2.2B - 当精确方法太慢时，启发式方法可能会更快地找到近似解决方案。

♦ 基本知识：4.2.2A - 启发式技术可以让我们在典型方法无法找到确切解决方案时找到近似解决方案。

* **学习目标：4.2.1** - 解释在合理的时间内运行和不运行算法之间的差异。[P1]

♦ 基本知识：4.2.1D - 一些问题可以解决，但不能在合理的时间内解决。在这些情况下，启发式方法可能有助于在合理的时间内找到解决方案。

♦ 基本知识：4.2.1C - 即使对于小的输入范围，一些问题也无法在合理的时间内解决。

♦ 基本知识：4.2.1B - 合理时间意味着算法所采用的步数小于或等于多项式函数输入的大小（常数，线性，平方，立方体等）。

♦ 基本知识：4.2.1A - 许多问题可以在合理的时间内解决。

* **持续理解：5.1** - 可以开发程序用于创意运算式，满足个人好奇心，创造新知识或解决问题（帮助人们，组织或社会）。
* **学习目标：5.1.3** - 协作开发一个程式。[P6]

♦ 基本知识：5.1.3F - 在开发程式时成功的协作需要参与者之间的有效沟通。

♦ 基本知识：5.1.3E - 协作有助于独立开发程式组件。

♦ 基本知识：5.1.3D - 在开发程式时协作更容易找到并纠正错误。

♦ 基本知识：5.1.3C - 在程式的迭代开发中的协作比单独开发程式需要更多不同技能。

♦ 基本知识：5.1.3B - 协作有助于通过编程来开发解决问题的多种观点。

♦ 基本知识：5.1.3A - 协作可以减少程序员个人任务的大小和复杂性。

* **学习目标：5.1.2** - 开发正确程式来解决问题。[P2]

♦ 基本知识：5.1.2J - 程序员在解决问题时设计，执行，测试，调试和维护程式。

♦ 基本知识：5.1.2I - 程序员的知识和技能影响程式的开发方式和程序如何解决问题。

♦ 基本知识：5.1.2H - 与程式用户进行协商和交流是程式开发的重要方面，以便解决问题。

♦ 基本知识：5.1.2G - 程式开发包括识别程序员和用户影响问题解决方案的关注点。

♦ 基本知识：5.1.2F - 在单独或协作的编程环境中工作，文件编制有助于开发和维护程式

♦ 基本知识：5.1.2E - 关于程式组件的文件编制，例如节段和函数，有助于开发和维护程式。

♦ 基本知识：5.1.2D - 程式文件编制有助于程序员开发和维护正确的程序以有效地解决问题。

♦ 基本知识：5.1.2C - 逐步增加测试程式节段和应用程式有助于创建大的正确程式。

♦ 基本知识：5.1.2B - 开发正确的程式组件，然后将它们组合起来有助于创建正确的程式。

♦ 基本知识：5.1.2A - 程式开发的迭代过程有助于开发正确的程式来解决问题。

* **学习目标：5.1.1** - 为创意运算式开发的程式，可以满足个人好奇心或创造新知识。[P2]

♦ 基本知识：5.1.1F - 计算技术的进步在其他领域产生并提高了创造力。

♦ 基本知识：5.1.1E - 一个计算机程式或运行程式的结果可能会迅速与大量用户分享，并可能对个人，组织和社会产生广泛的影响。

♦ 基本知识：5.1.1D - 额外的预期结果可能会各自实现程式的原始目的。

♦ 基本知识：5.1.1C - 为了满足个人好奇心或创造新知识而开发的用于创意运算式的程式，比其他为广泛分布而开发的程式更需要不同的开发水平和方法。

♦ 基本知识：5.1.1B - 为了满足个人好奇心或创造新知识而开发的用于创意运算式的程式，可能会有视觉，听觉或触觉输入和输出。

♦ 基本知识：5.1.1A - 根据程序员的开发目标，人们以多种方式开发和使用程式。

* **持续理解：5.2** - 人们编写程式来执行算法。
* **学习目标：5.2.1** - 解释程式如何使算法生效。[P3]

♦ 基本知识：5.2.1K - 算法，硬件和软件的改进增加了编程可解决的问题的类型和大小。

♦ 基本知识：5.2.1J - 在自动化时，简单的算法可以解决大量问题。

♦ 基本知识：5.2.1I - 可执行程式增加了可解决问题的比例。

♦ 基本知识：5.2.1H - 一段程序进程可以在一个或多个中央处理器上执行。

♦ 基本知识：5.2.1G - 一段程序进程可以单独执行或与其他程序进程一起执行。

♦ 基本知识：5.2.1F - 程序进程使用内部存储器，中央处理器（CPU）以及输入和输出设备。

♦ 基本知识：5.2.1E - 程式执行使流程自动化。

♦ 基本知识：5.2.1D - 对指令处理和程式执行的理解有助于编程。

♦ 基本知识：5.2.1C - 程式指令可能涉及初始化，更新，读取和写入的变量

♦ 基本知识：5.2.1B - 程式指令按顺序执行。

♦ 基本知识：5.2.1A - 在程式执行期间使用被处理过的程式指令，从而使算法生效。

* **持续理解：5.3** - 适当的抽象化可促进编程。
* **学习目标：5.3.1** - 用抽象化来管理程式中的复杂性。[P3]

♦ 基本知识：5.3.1 - 应用程序编程接口（API）连接软件组件，允许它们进行访问。

♦ 基本知识：5.3.1N - 应用程序编程接口（API）/程式库中的文档是编程的一个重要方面。

♦ 基本知识：5.3.1M - 应用程序编程接口（API）和程式库简化了复杂的编程任务。

♦ 基本知识：5.3.1L - 在编程中使用列表和函数作为抽象化可以更容易开发和维护程式。

♦ 基本知识：5.3.1K - 列表和列表操作（如添加，删除和搜索）在许多程式中很常见。

♦ 基本知识：5.3.1J - 不需要了解用于程序中的整数和浮点数是如何执行的。

♦ 基本知识：5.3.1I - 字符串和字符串操作（包括串联和某种形式的子串）在许多程式中很常见。

♦ 基本知识：5.3.1H - 数据抽象提供了一种将运转与模拟法分离的方法。

♦ 基本知识：5.3.1G - 当参数在程式中被调用时，参数给函数提供了不同的值。

♦ 基本知识：5.3.1F - 参数通过允许使用函数而不是重复代码来概括解决方案。

♦ 基本知识：5.3.1E - 参数化可以概括一个具体的解决方案。

♦ 基本知识：5.3.1D - 函数有名称并可能有参数和返回值。

♦ 基本知识：5.3.1C - 函数减少了编写和维护程式的复杂性。

♦ 基本知识：5.3.1B - 一个函数是一群已命名的编程指令。

♦ 基本知识：5.3.1A - 函数是可重复使用的编程抽象。

* **持续理解：5.4** - 程序是由人们为不同的目的而开发，维护和使用的。
* **学习目标：5.4.1** - 评估一个程序的正确性。[P4]

♦ 基本知识：5.4.1N - 程序功能的描述，是在高层次下程序运转什么的描述，而不是在低层次下程序语句如何完成它的工作。

♦ 基本知识：5.4.1M - 程序的功能通常由用户如何与它交互来描述。

♦ 基本知识：5.4.1L - 对程序的解释有助于人们理解其功能和目的。

♦ 基本知识：5.4.1K - 程序的正确性取决于程序组件的正确性，包括代码段和函数。

♦ 基本知识：5.4.1J - 调整可以包括关于程序如何符合其规格的书面说明。

♦ 基本知识：5.4.1I-程序员证明并解释程序的正确性。

♦ 基本知识：5.4.1H - 程序状态的可视化显示（或不同形式）可以帮助查找错误。

♦ 基本知识：5.4.1G - 预期输入具体数据的行为有助于人们理解程序应该做什么。

♦ 基本知识：5.4.1F - 了解程序应该做什么以便找到大多数程序错误。

♦ 基本知识：5.4.1E - 查找和纠正程序中的错误称为调试程序。

♦ 基本知识：5.4.1D - 在程序中较长的代码段比较短的代码段更难推理。

♦ 基本知识：5.4.1C - 变量和函数的有意义的名称有助于人们更好地理解程序。

♦ 基本知识：5.4.1B - 重复的代码可能会对推理程序造成困难。

♦ 基本知识：5.4.1A - 程序样式会影响程序正确性的测定。

* **持续理解：5.5** - 编程运用了数学和逻辑概念。
* **学习目标：5.5.1** - 在编程中运用适当的数学和逻辑概念。[P1]

♦ 基本知识：5.5.1J - 集合的基本操作包括添加元素，删除元素，迭代所有元素，以及确定元素是否在集合中。

♦ 基本知识：5.5.1I - 在开发程序时列表和其他集合可被视为抽象数据类型（ADT）。

♦ 基本知识：5.5.1H - 计算方法可以使用列表和集合来解决问题。

♦ 基本知识：5.5.1G - 使用布尔概念对程序组件进行直观和形式化的推理有助于开发正确的程序。

♦ 基本知识：5.5.1F - 使用“且”，“或”，“否”的复合运算式是大多数编程语言的一部分。

♦ 基本知识：5.5.1E - 逻辑概念和布尔代数是编程的基础。

♦ 基本知识：5.5.1D - 使用算术运算符的数学运算式是大多数编程语言的一部分。

♦ 基本知识：5.5.1C - 通过浮点来近似地表示实数，但浮点不一定具有无限的精度。

♦ 基本知识：5.5.1B - 由于存储限制，在程序中整数可能会以最大值或最小值呈现。

♦ 基本知识：5.5.1A - 数字和数值概念是编程的基础。

* **持续理解：6.1** - 互联网是自治系统的网络。
* **学习目标：6.1.1** - 解释互联网中的抽象化以及互联网如何运作。[P3]

♦ 基本知识：6.1.1I - 诸如超文本传输协议（HTTP），互联网协议（IP）和简单邮件传输协议（SMTP）等规范都是由互联网工程任务组（IETF）开发和监督的。

♦ 基本知识：6.1.1H - 可以使用IP地址的设备数量增长迅速，一个新的协议（IPv6）已经建立来处理更多设备的路由选择。

♦ 基本知识：6.1.1G - 域名系统（DNS）将域名转换为IP地址。

♦ 基本知识：6.1.1F - 互联网建立在不断发展的标准之上，包括地址和名称标准。

♦ 基本知识：6.1.1E - 通过分配互联网协议（IP）地址可以将新设备连接到互联网。

♦ 基本知识：6.1.1D - 互联网及互联网上的系统促进协作。

♦ 基本知识：6.1.1C - 设备和网络组成的互联网使用地址和协议进行连接和通信。

♦ 基本知识：6.1.1B - 端到端架构有助于连接互联网上的新设备和网络。

♦ 基本知识：6.1.1A - 互联网连接世界各地的设备和网络。

* **持续理解：6.2** - 互联网的特征影响建立在它上面的系统。
* **学习目标：6.2.2** - 解释互联网的特征如何影响建立在它上面的系统。[P4]

♦ 基本知识：6.2.2K - 系统的执行时间是传输和接收请求之间的时间。

♦ 基本知识：6.2.2J - 系统的带宽是传输速率的度量——数据量（以位为单位）可以在固定时间内发送。

♦ 基本知识：6.2.2I - 系统的大小和速度影响其使用。

♦ 基本知识：6.2.2H - 在Web浏览器和服务器之间共享信息和通信的标准包括超文本传输协议（HTTP)和安全套接字层/传输层安全性（SSL / TLS）。

♦ 基本知识：6.2.2G - 数据包和路由标准包括传输控制协议/因特网协议（TCP / IP）。

♦ 基本知识：6.2.2F-互联网是一种分组交换系统，通过分组交换系统将数据分解成称为数据包的比特块，其中包含正在传输的数据和用于路由数据的控制信息。

♦ 基本知识：6.2.2E - 开放的标准促进互联网的发展。

♦ 基本知识：6.2.2D - 接口和协议使互联网广泛被使用。

♦ 基本知识：6.2.2C - 域名服务器（DNS）中的层次结构有助于系统扩展。

♦ 基本知识：6.2.2B--互联网上两点之间的路由冗余（例如不止一种路由数据）增加了互联网的可靠性，并有助于互联网扩展到更多的设备和更多用户。

♦ 基本知识：6.2.2A - 层次和冗余有助于系统规模。

* **学习目标：6.2.1** - 解释互联网的特征及其构建的系统。[P5]

♦ 基本知识：6.2.1D - 互联网上的路由具有容错性和冗余性。

♦ 基本知识：6.2.1C - IP地址是分层的。

♦ 基本知识：6.2.1B - 域名语法是分层的

♦ 基本知识：6.2.1A - 互联网及互联网上的系统是分层和冗余的。

* **持续理解：6.3** - 网络安全是互联网和互联网系统的一个重要关注点。
* **学习目标：6.3.1** - 识别现存的网络安全问题和潜在问题选项，以用互联网及其系统解决这些问题。[P1]

♦ 基本知识：6.3.1M - 认证中心（CA）颁发数字证书，用于验证在安全通信中使用的加密密钥的所有权，并以此建立信任模型。

♦ 基本知识：6.3.1L - 不对称的公钥加密是一种广泛使用的加密方法，因为使用它可以加强安全性。

♦ 基本知识：6.3.1K - 对称加密是一种涉及密钥加密和解密的加密方法。

♦ 基本知识：6.3.1J - 开放标准有助于确保加密是安全的。

♦ 基本知识：6.3.1I - 密码学有数学基础。

♦ 基本知识：6.3.1H - 密码学对许多网络安全模型至关重要。

♦ 基本知识：6.3.1G - 防病毒软件和防火墙可以帮助防止未经授权访问私人数据。

♦ 基本知识：6.3.1F - 网络钓鱼，病毒和其他攻击有人员和软件组件。

♦ 基本知识：6.3.1E - 分散式阻断服务（DDoS）通过用来自多个系统的请求淹没目标，从而攻击和破解目标。

♦ 基本知识：6.3.1D - 网络战争和网络犯罪具有广泛且潜在的破坏性影响。

♦ 基本知识：6.3.1C - 实现网络安全包括软件，硬件和人员组成部分。

♦ 基本知识：6.3.1B - 域名系统（DNS）的设计并非完全安全。

♦ 基本知识：6.3.1A - 互联网的信任模式需要权衡。

* **持续理解：7.1** - 计算增强了沟通，互动和认知。
* **学习目标：7.1.2** - 解释人们如何参与解决问题的过程。[P4]

♦ 基本知识：7.1.2G - 从台式电脑到随时联机的笔记本电脑的扩散，正导致产生新的应用。

♦ 基本知识：7.1.2F - 众包为合作提供了新模式，例如通过资金将人员与工作和企业联系起来。

♦ 基本知识：7.1.2E - 一些在线服务利用许多人的贡献来造福个人和社会。

♦ 基本知识：7.1.2D - 数字化协作增强人的能力。

♦ 基本知识：7.1.2C - 人工智能利用众多人类的贡献来解决与数字数据和Web相关的问题。

♦ 基本知识：7.1.2B-在科学研究中，使用家用计算机，且利用量表和“公众科学”来解决科学问题影响着科学。

♦ 基本知识：7.1.2A - 分布式解决方案必须扩展以解决一些问题。

* **学习目标：7.1.1** - 解释计算创新如何影响沟通，互动和认知。[P4]

♦ 基本知识：7.1.1-因特网和万维网在许多领域对生产力产生了积极和消极的影响。

♦ 基本知识：7.1.1N - 因特网和万维网已经改变了许多领域，包括电子商务，医疗保健，信息，娱乐和在线学习。

♦ 基本知识：7.1.1M - 因特网和万维网增强了沟通和协作的方法和机会。

♦ 基本知识：7.1.1L - 计算有助于许多辅助技术提高人类能力。

♦ 基本知识：7.1.1K - 智能电网，智能建筑和智能交通正在改变和促进人类的能力。

♦ 基本知识：7.1.1J - 传感器网络有助于制约物理和环境系统。

♦ 基本知识：7.1.1I-全球定位系统（GPS）及相关技术已改变了人类旅行，导航和查找定位信息的方式。

♦ 基本知识：7.1.1H - 社交媒体，如博客和推特，加强了传播。

♦ 基本知识：7.1.1G - 搜索趋势是预报器。

♦ 基本知识：7.1.1F - 公共数据提供了广泛的使用权，并为识别出的问题提供解决方案。

♦ 基本知识：7.1.1E - 广泛获取信息便于查明问题，制定解决方案和传播结果。

♦ 基本知识：7.1.1D - 云计算促进沟通和协作的新方式。

♦ 基本知识：7.1.1C - 社交媒体不断发展并助长了新的沟通方式。

♦ 基本知识：7.1.1B - 视频会议和视频聊天促进了沟通和协作的新方式。

♦ 基本知识：7.1.1A - 电子邮件，短信服务（SMS）和网络聊天系统助长了沟通和协作的新方式。

* **持续理解：7.2** - 在几乎每个领域，计算机都能实现创新。
* **学习目标：7.2.1** - 解释计算在其他领域如何影响创新。[P1]

♦ 基本知识：7.2.1G - 作为一种赋能技术的计算进步已经在其他领域产生并增加了创造力。

♦ 基本知识：7.2.1F - 摩尔定律鼓励使用计算机的行业根据计算能力的预期增长有效地规划未来的研究和开发。

♦ 基本知识：7.2.1E-开放和精心策划的科学数据库使科学研究人员受益。

♦ 基本知识：7.2.1D - 开放的使用权和知识共享可以广泛地获取数字信息

♦ 基本知识：7.2.1C - 计算通过提供信息使用权和共享信息来实现创新。

♦ 基本知识：7.2.1B-科学的计算促进了科学和商业的创新。

♦ 基本知识：7.2.1A - 机器学习和数据挖掘已经实现了医学，商业和科学领域的创新。

* **持续理解：7.3** - 计算对人类和社会具有既有益也有害的全球影响。
* **学习目标：7.3.1** - 分析计算的有益和有害影响。[P4]

♦ 基本知识：7.3.1Q-开源软件和免费软件对广泛访问程序，程式库和代码具有实际，商业和道德影响。

♦ 基本知识：7.3.1P-数位千年版权法（DMCA）在广泛获取受版权保护的数字资料方面是益处也是挑战。

♦ 基本知识：7.3.1-通过结合现有内容来创建数字音频，视频和文本内容受到版权问题的影响。

♦ 基本知识：7.3.1N - 广泛获取数字信息会引发有关知识产权的问题。

♦ 基本知识：7.3.1M - 目标广告用于帮助个体，但它可能会在个人和集合层中被滥用。

♦ 基本知识：7.3.1L - 如果隐私和其他保护措施被忽略，商业和政府管理信息可能会被利用。

♦ 基本知识：7.3.1K - 人们可以在线即时访问大量信息，访问这些信息可以收集个人和集体数据。

♦ 基本知识：7.3.1J-技术能够收集，使用和利用关于个人，团体和机构的信息。

♦ 基本知识：7.3.1I - 匿名在线互动可以使用在线匿名软件和代理服务器。

♦ 基本知识：7.3.1H - 信息聚合，如地理定位，信息记录程序和浏览历史记录会引发隐私和安全问题。

♦ 基本知识：7.3.1G-隐私和安全问题出现在计算系统和工件的开发和使用之中。

♦ 基本知识：7.3.1F - 开源软件以及软件和内容的许可会引发法律和道德方面的关注。

♦ 基本知识：7.3.1E-商业和政府对数字信息的审查会引起法律和道德的关注。

♦ 基本知识：7.3.1D - 通过验证访问或匿名访问数字信息都会引起法律和道德的关注。

♦ 基本知识：7.3.1C - 访问对等网络数字内容会引发法律和道德的关注。

♦ 基本知识：7.3.1B - 商业性地获取下载或流式传输音乐和电影会引发法律和道德的关注。

♦ 基本知识：7.3.1A - 通过计算实现的创新会增加法律和道德方面的关注。

* **持续理解：7.4** - 计算创新与通过其设计和使用的经济，社会和文化环境相互影响。
* **学习目标：7.4.1** - 解释计算与现实世界环境之间的关系，包括经济，社会和文化环境。[P1]

♦ 基本知识：7.4.1E - 商业和政府支持网络和基础设施。

♦ 基本知识：7.4.1D - 集体和个人受到“数位划分”的影响——基于社会经济或地理特征，不同地访问计算机和互联网。

♦ 基本知识：7.4.1C-计算资源的全球分布引发了公平，准入和功耗问题。

♦ 基本知识：7.4.1B - 移动，无线和网络计算对全球创新产生影响。

♦ 基本知识：7.4.1A-社交媒体和在线访问的创新和影响在不同国家和不同社会经济群体中有所不同。

* **持续理解：7.5** - 调查过程得到有效组织和资源选择的帮助。适当的技术和工具有助于获取信息，并有助于评估信息来源的可信度。
* **学习目标：7.5.2** - 在线评估和打印资源的适当性和可信度。[P5]

♦ 基本知识：7.5.2B - 信息在支持适当声明或调查目的时被认为是相关的。

♦ 基本知识：7.5.2A - 确定一个资源的可信度需要考虑和评估作者，出版商，网站所有者和赞助商的声誉及证书。

* **学习目标：7.5.1** - 使用有效策略访问，管理和标志信息。[P1]

♦ 基本知识：7.5.1C - 抄袭是一种严重的犯罪行为，即当一个人将别人的想法或言辞说成他自己的。通过准确地确认来源可以避免抄袭。

♦ 基本知识：7.5.1B-高级搜索工具，布尔逻辑和关键词可以根据各种因素（例如，同行评审状态，出版类型）来改进搜索焦点和限制搜索结果。

* ♦ 基本知识：7.5.1A - 在线数据库，程序馆目录，内部辅助和一些主要资源。