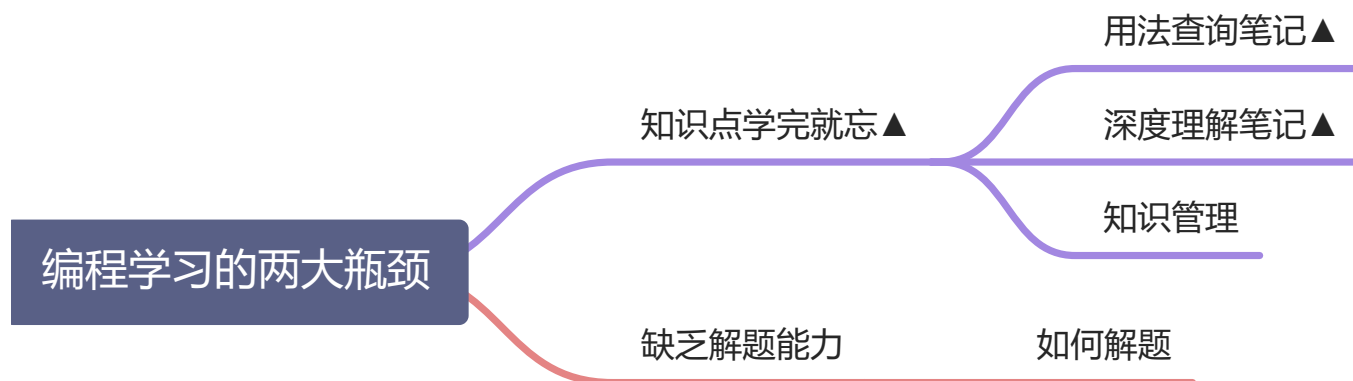


第8课 如何解决问题

一、课程结构导图



注：▲为重点知识点。

二、知识点讲解

2.1 知识点学完就忘(学习瓶颈一)

2.1.1 用法查询笔记

概念： 记录知识点的基础用法，能供你快速查阅，加深对知识的印象。

用法： 注释说明代码含义，运行结果。

示例：

```
1 知识：列表
2 _____
3 s = [1,2,3,4,5]
4
5 print(s) # 打印列表
6 # >>> [1,2,3,4,5]
7
8 print(len(s)) # 打印列表长度
9 # >>> 4
10
11 print(s[2]) # 打印列表中第3个元素
12 # >>> 3
13
14 print(s[-1]) # 打印列表中第倒数第1个元素
15 # >>> 5
```

补充点：

1. 课程中涉及到的知识点都可以用用法查询笔记来记录，对知识点进行归纳和整理。
2. 在记录的过程中，要记什么内容取决于你对具体知识点的熟悉度。对知识点越熟悉，记录的内容越多。

2.1.2 深度理解笔记

概念：【深度理解笔记】重在“理解”，当我们对知识点有印象的时，可以通过深度理解笔记来总结、理解知识的使用场景。

用法：描述说明知识点的使用场景（什么时候用，怎么用）。

示例：

```
1 理解循环
2 _____
3 1、什么时候用到循环？
4 当出现重复性或规律性任务时，可以用循环来解决重复性劳动
5
```

- 6 2、什么时候用到for循环，什么时候用到while循环？
- 7 当循环次数知道时，优先用for循环，不知道循环循环时用while循环
- 8
- 9 3、for循环的常见用法
- 10 遍历range()函数
- 11 遍历列表、字典
- 12 遍历字符串

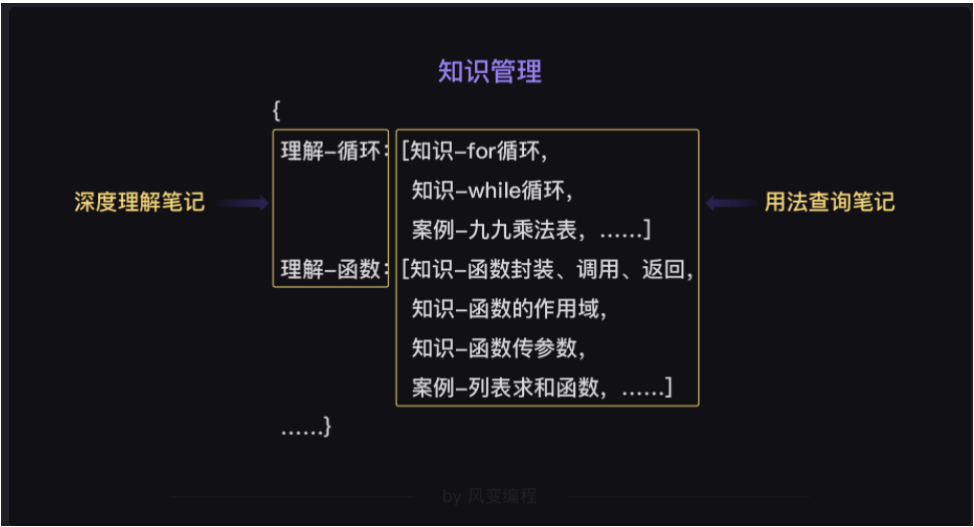
补充点：【用法查询笔记】解决的是“知识点是什么”的问题，那么【深度理解笔记】更侧重解决“什么时候要用”以及“怎么用这个知识点”的问题。

2.1.3 知识管理

概念：在解决一个编程问题的时候，【深度理解笔记】可以轻易想到需要用到哪块知识，【用法查询笔记】里寻找相关的知识细节和具体案例，形成解题的思路。因此我们可以搭建出一个知识框架。

用法：【深度理解笔记】作为键，【用法查询笔记】作为键值。

示例：



补充点：知识管理其实是用法查询笔记和深度理解笔记的结合运用，主要围绕着三点，知识点是什么，什么场景下用，怎么用。前者是用法查询笔记的作用，后两者是深度理解笔记的作用。

2.2 缺乏解题能力（学习瓶颈二）

2.2.1 如何解题

概念：当我们遇到一个问题时，需要一个解题技巧，来帮助我们提供解题思路。



示例：

```
1 # 打印出九九乘法表
2 for i in range(1,10):
3     for j in range(1,i+1):
4         print( '%d X %d = %d' % (j,i,i*j),end = ' ' )
5     print(' ')
```

代码解析：

1. 该代码是想实现九乘九乘法表，采用的是循环嵌套模式，内层循环for i in range(1,i+1)，功能是打印出每行1*n（n为1-9）的结果，而外层循环for j in range(1,10)，功能是重复9次内层循环，打印出9行1*n的结果。
2. end是用来控制换行行数和结尾字符，是print函数中的一个参数。例如print('hello',end='!')，print('world')，打印结果为'Hello!world'。

- **所需知识点：**

1. for循环

- 2. end参数
- 3. 去格式化字符 (%)

• 思考切入点:

- 1. 发现一个规律：每一行的等式里，第一位数会递增，第二位数则会保持不变，且在第几行就会有多少个等式。
- 2. 每一行其实都是一个小循环，那我们就可以以此为切入点，先把每一行的小循环写出来，再进一步寻找规律。

• 代码执行流程:

i的取值	j的取值	打印结果
1	1	1*1
2	1, 2	1*2, 2*2
3	1, 2, 3	1*3, 2*3, 3*3
4	1, 2, 3, 4	1*4, 2*4, 3*4, 4*4
5	1, 2, 3, 4, 5	1*5, 2*5, 3*5, 4*5, 5*5
.....
9	1, 2, 3, 4, 5, 6, 7, 8, 9	1*9, 2*9, 3*9, 4*9 9*9