

作业 1

下列题目中，第 1、4、7 题和附加题需要提交代码，其余内容为纸面作业，请用 word 或 pdf 的格式进行提交。

1. 写一段 python 程序，计算 $1+2+3+\dots+1,000,000$ ，输出计算所需时间。（10 分）
2. 将下列函数按增长速度进行排序，并指出增长速度相同的函数：（15 分）

$N, N^{\frac{1}{2}}, N^{1.5}, N^2, N \log N, N \log(\log N), N(\log N)^2, N \log(N^2), 2^N, 2^{N/2}, 37, N^2 \log N, N^3$

3. 对下列 6 段程序：（18 分）

- a) 分析其运行时间，用大 O 符号表示
- b) 在 Python 中运行这些程序，给出 N 取不同的几个值时的运行时间
- c) 将你的时间复杂度分析与实际运行时间相比较，它们是否是一致的？

(1) <pre>sum = 0 for i in range(N): sum += 1</pre>	(2) <pre>sum = 0 for i in range(N): for j in range(N): sum += 1</pre>
(3) <pre>sum = 0 for i in range(N): for j in range(N * N): sum += 1</pre>	(4) <pre>sum = 0 for i in range(N): for j in range(i): sum += 1</pre>
(5) <pre>sum = 0 for i in range(N): for j in range(i * i): for k in range(j): sum += 1</pre>	(6) <pre>sum = 0 for i in range(N): for j in range(i * i): if j % i == 0: for k in range(j): sum += 1</pre>

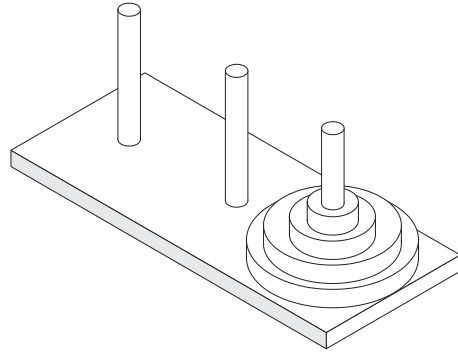
4. 汉诺塔问题：

一个古老的印度神庙里有三根柱，其中一个自上而下放置了由小到大的 64 个金盘。僧侣们依照以下规则把 64 个金盘移动到另一个柱子上：

- 一次只能移动一个金盘；
- 金盘只能在柱子上存放；
- 小盘必须始终放置在大盘上方；

传说中，64 个金盘移动完世界毁灭……

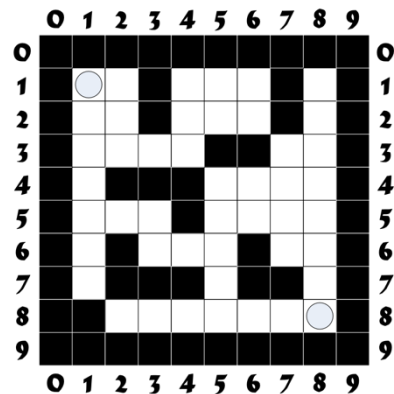
请设计并用 Python 实现算法，接收用户输入的金盘数目 N ，输出将金盘从柱子 0 移动到柱子 2 的全过程。输出格式为：move(a , x , y)，其中 a 为盘子编号 ($1 \sim N$)， x 和 y 为柱子编号 ($0 \sim 2$)。 (15 分)



5. 如果 3 个元素进栈顺序为 X、Y、Z，试写出所有可能的出栈顺序（可能后来者进栈时，先来者已经出栈）。 (12 分)

6. 迷宫问题：

考虑一个长为 M 宽为 N 的迷宫。其入口在 $(1, 1)$ 处，出口在 (M, N) 处。例如，下图中 $M=N=8$ 。请设计一个算法，找到一条从入口到出口的路（也可能没有这样的路），并给出其运行的时间复杂度。 (15 分)



7. 八皇后问题：

请设计并用 Python 实现一个算法，找出所有在 8×8 的国际象棋棋盘上放置皇后的方式，使得这些皇后互相之间不能处于可互吃的状态。 (15 分)

（注：皇后是国际象棋所有棋子中威力最大的，可以吃掉与其同一行、列、或对角线上的棋子）

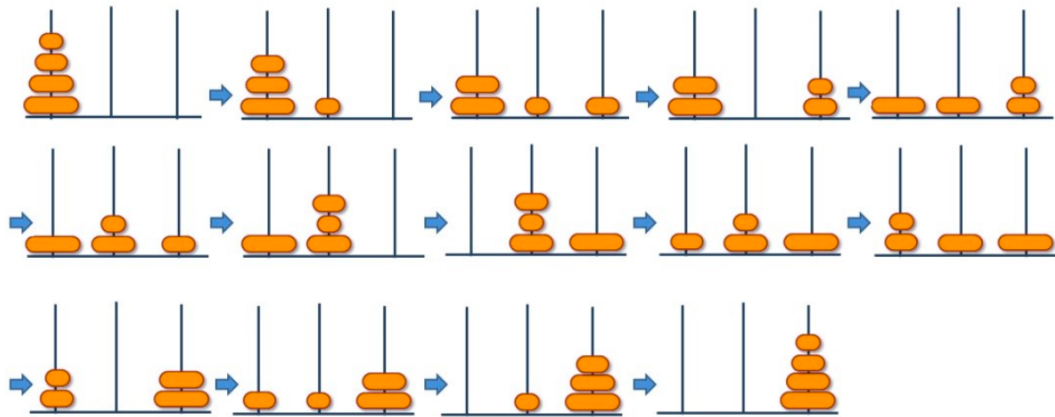
附加题：

懒和尚汉诺塔问题：

由于汉诺塔移动的次数实在太多，神庙的和尚们决定偷懒。为了虔诚地偷懒，他们用严格的语言描述了新的规则：

1. 有三根柱子，编号为 0、1、2。初始状态下，0 号柱子从底向上按照从大到小的次序依次放置了 n 个金盘。

2. 每次移动将一根柱子最顶部的一个金盘移动到另一根柱子的最顶部。在整个操作过程中，每根柱子上的最底部的金盘必须是最大的，而其他金盘的顺序可以和大小无关。
3. 希望通过移动达到的目标状态是，2 号柱子从底向上按照从大到小的次序依次放置这 n 个金盘。



下图的移动在新的规则下是合法的，总体的移动次数也从原规则下的 15 次减少到了 13 次。

编写 Python 程序，用尽量少的移动次数解决懒和尚汉诺塔问题。

请仔细阅读以下提交说明（以后同）：

1. 每道题应该在独立的 py 文件中编写，注意缩进规范、命名直观、注释充分、有必要的 print 命令来输出最后的结果，并且在输出的文字中有清晰的提示来说明输出的东西是什么；
2. 请将每个 py 文件不要打包，写清题号 “week2_1.py” “week2_2.py” 依次类推，逐个上传到 elearning 的作业区。