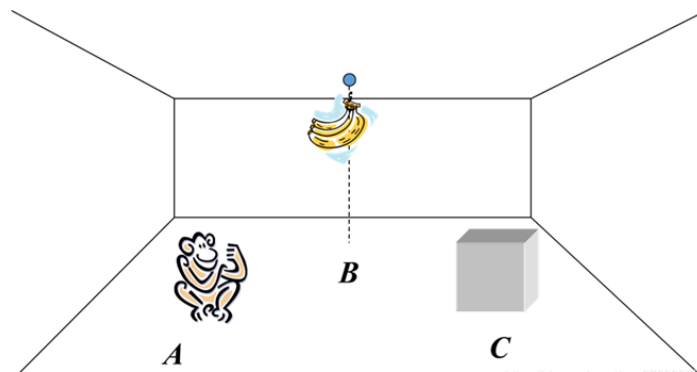


实验一 知识表示方法

一. 猴子摘香蕉问题

1. 问题描述

房内有一个猴子，一个箱子，天花板上挂了一串香蕉，其位置如图 1 所示，猴子为了拿到香蕉，它必须把箱子搬到香蕉下面，然后再爬到箱子上。请定义必要的谓词，列出问题的初始化状态（即下图所示状态），目标状态（猴子拿到了香蕉，站在箱子上，箱子位于位置 b）。



2. 样例

输入样例

分别输入三个整数，代表猴子，香蕉，盒子的位置。

```
0 1 2
```

输出样例

输出所需步骤。

```
step: 1 Monkey goes to 2 from 0
step: 2 Monkey catches the box from 2 to 1
step: 3 Monkey climbs the box.
step: 4 Monkey gets the banana!
```

3. 实验分析

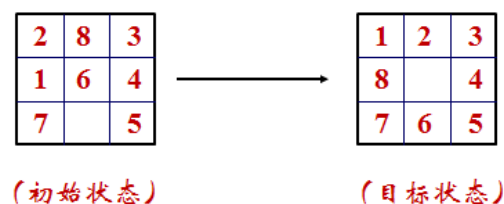
该问题较为简单，编写好四个函数即可实现。

```
1. Monkey_go_box(monkey, box)
2. Monkey_move_box(box, banana)
3. Monkey_on_box()
4. Monkey_get_banana()
```

二. 八数码问题

1. 问题描述

3×3 九宫棋盘，放置数码为 1 -8 的 8 个棋牌，剩下一个空格，只能通过棋牌向空格的移动来改变棋盘的布局。我们希望根据给定初始布局（即初始状态）和目标布局（即目标状态），如何移动棋牌才能从初始布局到达目标布局，找到合法的走步序列？



我们为了降低难度，对问题做了一个简化：输入一组初始状态的顺序（空格用小写字母 x 进行代替），输出为目标状态为 12345678x 需要的步骤数。

2. 样例

输入样例

输入一组初始状态的顺序（空格用小写字母 x 进行代替）

```
2 3 4 1 5 x 7 6 8
```

输出样例

输出为目标状态为 12345678x 所需要步骤数。

```
19
```

3. 问题分析

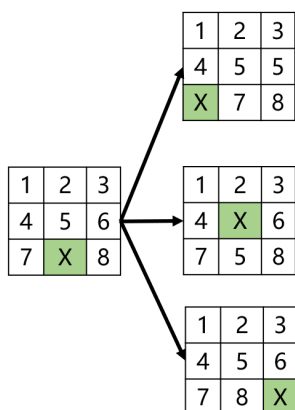
八数码问题其实本质上是一个搜索问题，我们要先对该问题做一个简单的分析。

1	2	3
4	5	6
7	X	8

我们将该状态记作：
1 2 3 4 5 6 7 x 8

1	2	3
4	5	6
7	8	x

目标状态：
1 2 3 4 5 6 7 8 x



- 可以和 X 交换的是 7, 5, 8 三个数。
- 直接存储二维表太复杂了, 这里用一个一维表来记录。
- X 在一维表的下标为 $K = 7$
- X 在二维表的下标可以记作 $A = K / 3, B = K \% 3, (A,B) = (2,1)$
- 随后利用 for 循环进行遍历其可以到达的结点, 这里用两个数组

$\begin{cases} dx = [0,1,0,-1] \\ dy = [1,0,-1,0] \end{cases}$ 来表示 X 周围的点 $(A + dx[i], B + dy[i])$ 。

由于该问题为一个搜索问题, 这里采用 BFS 来进行解决:

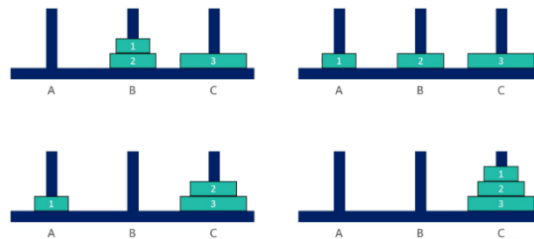
1. 初始化状态队列 Q 和标记数组 st;
2. 遍历队列 Q:
 - a) 取出状态队列的队头元素, 并拿出 x 所处的一维数组下标 K;
 - b) 如果已经到达目标状态, 则直接返回输出步骤数;
 - c) 将 K 借助上面的分析转换为二维坐标(A, B)
 - d) For (借助 dx, dy 遍历 X 的邻居):
 - i. If (满足条件, 可以交换):
 1. 交换位置
 2. If (该状态从未出现过):
 - a) 压入队列, 标记, 记录步骤数;
 3. 交换回来位置用于回溯
 - e) Return -1

三. 梵塔问题

1. 问题描述

相传在古印度圣庙中, 有一种被称为汉诺塔(Hanoi)的游戏。该游戏是在一块铜板装置上, 有三根杆(编号 A、B、C), 在 A 杆自下而上、由大到小按顺序放置

多个金盘(如下图所示)。游戏的目标：把 A 杆上的金盘全部移到 C 杆上，并仍保持原有顺序叠好。操作规则：每次只能移动一个盘子，并且在移动过程中三根杆上都始终保持大盘在下，小盘在上，操作过程中盘子可以置于 A、B、C 任一杆上？



2. 样例

输入样例

输入一个整数 n ，代表一共 n 层汉诺塔。

3

输出样例

输出步骤，每一步代表将一个杆上最高层的金盘放置到目标杆上。

```
A > C
A > B
C > B
A > C
B > A
B > C
A > C
```

3. 问题分析

对于该问题，这里给出递归算法求解。

对于递归算法可以这样进行设想：

第一步：先将 A 上 (1 ~ n-1) 号盘子放到 B 上；

第二步：再将 A 上 (n) 号盘子放在 C 上；

第三步：最后将 B 上的 (1 ~ n-1) 号盘子放到 C 上。

如果按照这样的设想，如何完成这第一步：将 A 上 (1 ~ n-1) 号盘子放到 B 上？可以这样进行设想：

第一步：先将 A 上 (1 ~ n-2) 号盘子放到 C 上；

第二步：再将 A 上 (n-1) 号盘子放在 B 上；

第三步：最后将 C 上的 (1 ~ n-2) 号盘子放到 B 上。

这样设想之后会发现，其实这两步中间有许多共同之处。

最后再考虑一下边界问题：如果只有一个盘子的时候，直接放置就可以。

伪代码如下：

Function: Move (将 1 ~ n 号盘子从杆 A 借助杆 B 放到杆 C 上)：

1. If ($n=1$):
 - i. 直接将杆 A 顶层的盘子放在 C 上；
2. Else:
 - i. Move (将 1 ~ n-1 号盘子从杆 A 借助杆 C 放到杆 B 上)
 - ii. 将杆 A 的 n 号盘子放到杆 C 上；
 - iii. Move (将 1 ~ n-1 号盘子从杆 B 借助杆 A 放到杆 C 上)

四. 实验提交要求

将实验 1-3 问题分析，实验运行情况按照实验报告样例编写并且保存到以“学号+姓名.doc/docx”形式的文件中，例如“202222408001+张三.doc/docx”。实验报告样例如下：

封面（提交时删除）

《人工智能》实验报告

实验名称 知识表征方法（每次实验自行更改）

学 号 _____

姓 名 _____

日期

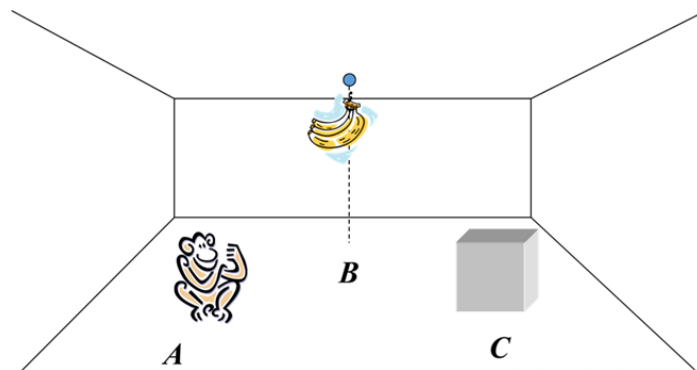
内页：（提交时删除）

实验一 知识表示方法

一、猴子摘香蕉问题

1.实验内容：

房内有一个猴子，一个箱子，天花板上挂了一串香蕉，其位置如图 1 所示，猴子为了拿到香蕉，它必须把箱子搬到香蕉下面，然后再爬到箱子上。请定义必要的谓词，列出问题的初始化状态（即下图所示状态），目标状态（猴子拿到了香蕉，站在箱子上，箱子位于位置 b）。



2.实验思路：

3.程序清单：需加适当注释

4.运行结果说明：

二、

...

文字用小 4 号或 4 号；程序和注释用 5 号，程序不能截图，需可再运行。