

# 人工智能作业一

姓名：黄亮铭

学号：2022155028

## 1、利用 python 实现堆栈或队列(二选一)，并尝试解决如下对应的问题。

- a. 四月一日快到了，Vayko 想了个愚人的好办法——送礼物。嘿嘿，不要想的太好，这礼物可没那么简单，Vayko 为了愚人，准备了一堆盒子，其中有一个盒子里面装了礼物。盒子里面可以再放零个或者多个盒子。假设放礼物的盒子里不再放其他盒子。用()表示一个盒子，B 表示礼物，Vayko 想让你帮她算出愚人指数，即最少需要拆多少个盒子才能拿到礼物。

### 输入格式

输入一行字符串代表 Vayko 的礼物透视图

### 输出格式

输出愚人指数，即最少需要拆多少个盒子才能拿到礼物。

### 输入输出样例

输入：(((O(B)O))O)

输出：4

**解题思路：**（1）首先创建 Stack 类，并编写 top、pop、push、size 等方法；

（2）然后将 Stack 类实例化；

（3）再读取输入并存储到列表 line 中；

（4）每次从 line 中读取一个字符，

I. 如果为左括号，则将其压入栈中；

II. 如果为右括号，则将栈顶元素出栈；

III. 如果为字符 B，则保存答案并退出，此时答案为留存于栈中的元素的个数。

（5）输出答案。

- b. 一个算术表达式中包括圆括号、方括号和花括号三种形式的括号，判别表达式中括号是否正确配对

括号匹配结果有四种：

- 0：左右括号匹配正确       $\{[(1+2)*3]-1\}$
- -1：左右括号配对次序不正确       $\{[(1+2)*3]-1\}$
- -2：右括号多于左括号       $(1+2)*3)-1\}$
- -3：左括号多于右括号       $\{[(1+2)*3-1]$

### 输入格式

输入一个算术表达式

### 输出格式

输出对应的符号匹配结果

### 输入输出样例

输入： $\{[(1+2)*3]-1\}$       输出：0

输入:  $\{[(1+2)*3]-1\}$       输出: -1

输入:  $(1+2)*3-1\}$       输出: -2

输入:  $\{[(1+2)*3-1]$       输出: -3

**解题思路:** (1) 首先创建 Queue 类, 并编写 push、pop\_front、pop\_back、size\_等方法;

(2) 然后将 Queue 类实例化;

(3) 再读取输入并存储到 line 中;

(4) 每次从 line 中读取一个字符,

I. 如果为左括号 (不论大中小), 将其入队;

II. 如果为数字或运算符, 直接跳过;

III. 如果为右括号, 首先看队尾元素是否与其匹配, 如匹配, 则继续; 否则存储相应结果, 程序跳到最后一步;

IV. 最后看队列中是否存在元素, 如存在, 则说明左括号多于右括号, 否则, 左右括号匹配, 然后存储相应结果;

(5) 输出答案。

## 2、利用 python 实现树或图的遍历 (二选一), 并指出该遍历方法的优缺点。

**解题思路:** (1) 创建 Tree 类, 并编写 add、dfs 等方法, 定义 h、to、val、idx、root 等成员变量, 其中前 3 个为列表, 后 2 个为 int 类型变量。

(2) h[i] 表示节点 i 所连接的儿子的数据所在的地址, nxt[i] 表示节点 to[i] 的兄弟节点的数据所在的地址, val[i] 表示节点 to[i] 与其父亲的边的权值, idx 表示当前树边的总数, root 表示树的根节点;

(3) add 方法可以向树中加入新边, dfs 方法遍历树;

(4) 首先从文件中读入树边等信息到 lines 中, 通过 add 方法构造树;

(5) 在 dfs 方法的不同位置输出节点可以实现不同顺序的遍历。

优点: (1) 时间复杂度是线性的;

(2) 容易在不同顺序的遍历中转换;

缺点: (1) 需要建立双向边, 空间复杂度较高。