形式语言与自动机第一次 Python 编程作业

# 第一题（5分）

假设 *L*1 是一个语言（即一组字符串构成的集合），*L*2 是另一个语言，定义两个语言的串接为 *L*1*L*2 = {*xy*∣*x*是*L*1中的一个字符串且*y*是*L*2中的一个字符串}。编写一个以两个语言 *L*1 与 *L*2 为输入参数的串接函数，完成 *L*1 与 *L*2 的串接，要求自行构造 *L*1 与 *L*2，使得每个

print

语言里面的字符串个数大于或等于 3，并且使用打印出函数返回的语言中的所有字符串。

# 第二题（5分）

语句打印出串接函数调用执行的结果，即

假设 *L* 是一个语言（即一组字符串构成的集合），定义 *L* **上的上界为** *T* **的星操作**：

*ST* (*L*) = {*w*1*w*2 ⋯ *wn*∣这里*n* <= *T* ，且每个*wi*是*L*中的一个字符串}

编写一个**以** *L* **与** *T*

**为参数的上界为** *T*

**的星操作函数**，要求自行构造 *L*，使得 *L* 里面的字符串

个数大于或等于 2，并且使用

print

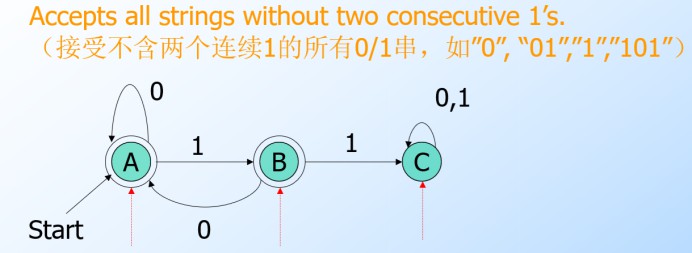
语句打印出**上界为** *T*

**的星操作**函数在 *T* = 3 时调用执行的

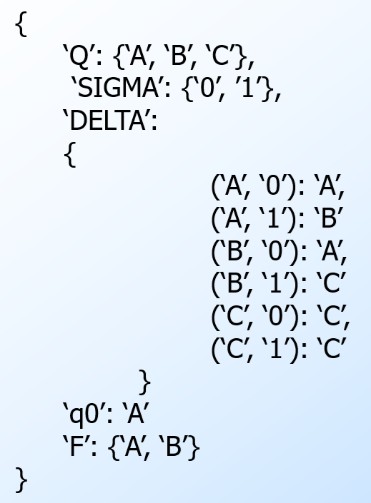
结果，即打印出函数返回的语言中的所有字符串。

# 第三题（10分）

考虑课件 **lec1-dfa** 上的 DFA:



参照以下描述上述 DFA 的 Python 代码：



编写一个以状态 *q* 与字符 *a* 为参数的转移函数 delta(*q*, *a*)，以及一个以状态 *q* 与字符串 *s* 为参数的扩展转移函数 Edelta(*q*, *s*)，要求 Edelta(*q*, *s*) 通过调用 delta(*q*, *a*) 而实现。自行构造一个被上述 DFA 接受的正例字符串 *Sp*，以及一个被上述 DFA 拒绝的负例字符串 *Sn*，*Sp* 与 *Sn*分别包含的字符个数要大于或等于 3。要求通过调用扩展转移函数 Edelta(*q*, *s*) 实现上述 DFA

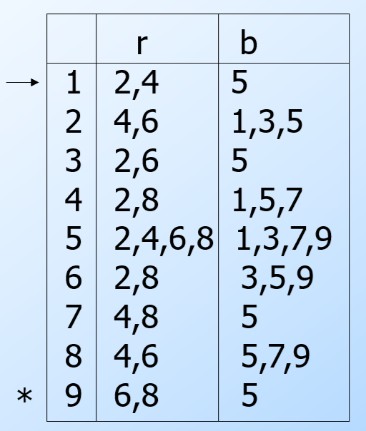
print

分别以 *Sp* 与 *Sn* 作为输入串的执行过程，使用

# 第四题（10分）

考虑课件 **lec2-nfa** 上以如下转移表定义的 NFA:

语句打印出上述 DFA 的每一次转移。



编写 Python 程序完成课件 **lec2-nfa** 上的 **subset construction 算法**，要求使用出上述 NFA 通过执行 **subset construction 算法**得出的 DFA 的转移表。

print

# 提交方式

语句打印

1. 11 月 6 日晚上 12 点之前以个人姓名为邮件标题，将 Python 代码以及自行构造的语言或用例等发送邮件提交到 [linkh8@mail2.sysu.edu.cn](mailto:linkh8@mail2.sysu.edu.cn) ；
2. 11 月 7 日课上每组提交双面打印版一份, 签署组长姓名与学号。