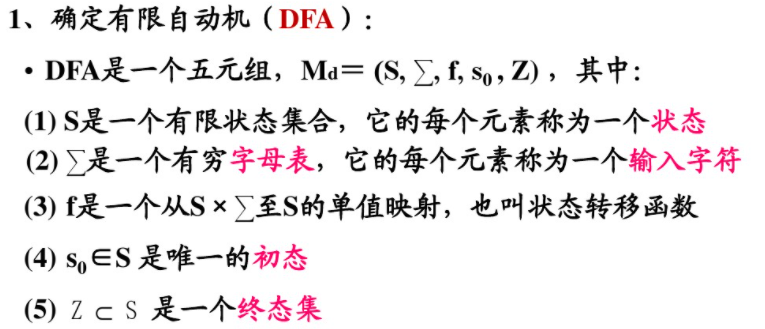
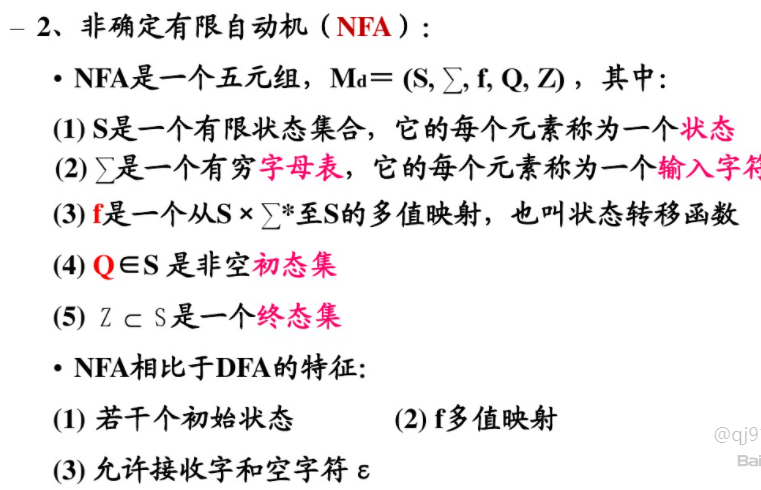
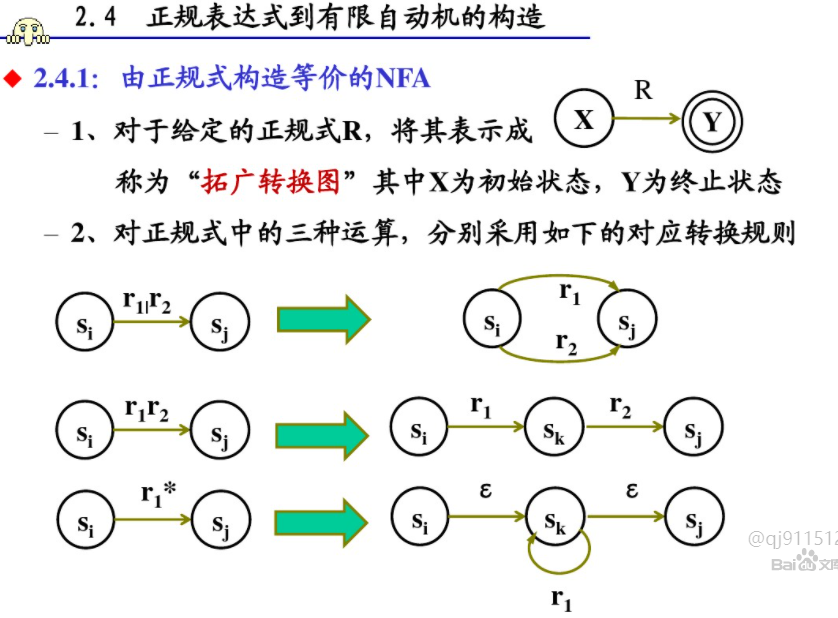
# 定义

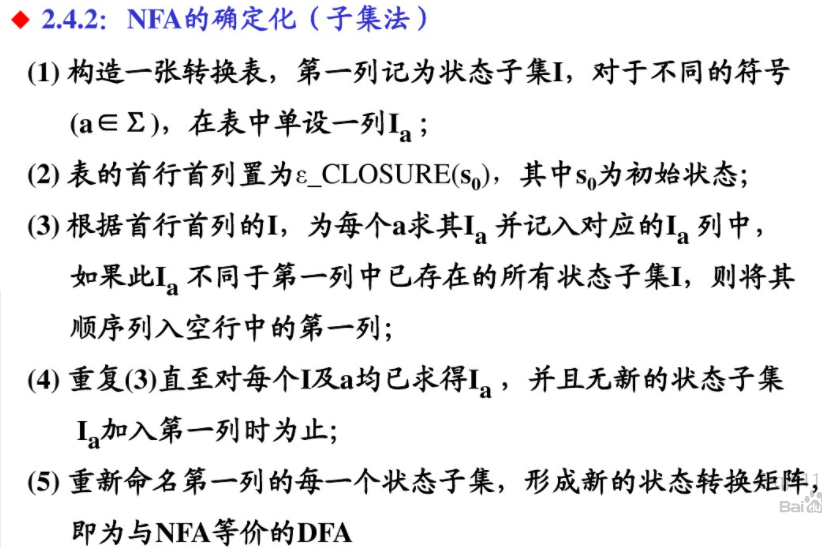




# CreateNFA



# Nfa2dfa



Subset Construction

初始Dstates，它仅仅含有状态(D的状态)ε-closure(s0)，并且状态未被标记，s0表示开始状态，注意，Dstates放的是D的状态  
while ( Dstates 有未标记的状态 T ) { // T是D中的一个状态，也是N中一个状态集  
    标记 T;  
    for ( input symbol a ){                  // 遍历所有的input symbol  
       U = ε-closure(move(T, a));        // move为NFA的move函数  
       if ( U 不在 Dstates 中 )  
          把U作为尚未标记的状态加入Dstates;  
       Dtran[T, a] = U  
    }  
}  
  
注意，状态s，ε-closure(s)一定包含s

把T的所有状态压入stack(栈);  
ε-closure(T)的初始值为 T 中的所有元素 ;  // 也就是一定包含他们本身  
while( 栈非空 ) {  
    弹出栈顶元素 t ;  
    for( 每个属于 move(t, ε) 的状态 u ){  
       if( u 不在 ε-closure(T) 中 ){  
          u 加入 ε-closure(T);  
          把 u 入栈;  
       }  
    }  
}

# Dfa化简

4.DFA最小化

## 4.1 最小状态DFA的含义

1.没有多余状态(死状态)

2. 没有两个状态是互相等价（不可区别）

## 4.2 两个状态s和t等价的条件

兼容性（一致性）条件——同是终态或同是非终态

传播性（蔓延性）条件——从s出发读入某个a和从t出发经过某个a并且经过某个b到达的状态等价。

————————————————

版权声明：本文为CSDN博主「xxc1605629895」的原创文章，遵循CC 4.0 BY-SA版权协议，转载请附上原文出处链接及本声明。

原文链接：<https://blog.csdn.net/johnf_nash/article/details/21988565>

