* 1. 下图给出了两台DFA M1和M2的状态图。 回答下述关于这两台机器的问题。

M1：

     a. 它们的起始状态是什么?

**起始状态是q1**

     b. 它们的接受状态集是什么?

**接收状态集是{q2}**

     c. 对输入aabb，它们经过的状态序列是什么?

**状态序列是q1 q2 q3 q2 q3**

     d. 它们接受字符串aabb吗?

**不接受**

     e.它们接受字符串ε吗?

**不接受**

图示

描述已自动生成

M2：

     a. 它们的起始状态是什么?

**起始状态是q1**

     b. 它们的接受状态集是什么?

**接收状态集{q1,q4}**

     c. 对输入aabb，它们经过的状态序列是什么?

**q1 q1 q1 q2 q4**

     d. 它们接受字符串aabb吗?

**接受**

     e.它们接受字符串ε吗?

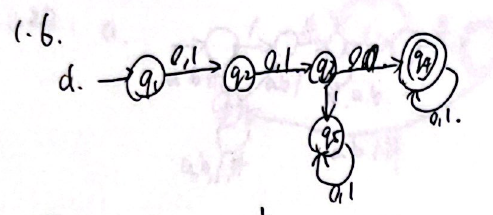
接受

图示

描述已自动生成

1.6 画出识别下述语言的DFA状态图。字母表为{0,1}

     d. { w | w的长度不小于3, 并且第3个符号为0};



1.7. 给出下述语言的NFA，并且符合规定的状态数。字母表为{0,1}

     e. 语言0\*1\*0\*0, 3个状态。

卡通人物

中度可信度描述已自动生成

1.16(b)  将如下图的非确定有限自动机转换成等价的确定有限自动机.

图示

描述已自动生成

图示, 工程绘图

描述已自动生成

1.21(a)  将如下图的有限自动机转换成等价的正则表达式.

图片包含 游戏机, 物体, 钟表, 画

描述已自动生成文本, 信件

描述已自动生成

1.22 在某些程序设计语言中, 注释出现在两个分隔符之间, 如/#和#/. 设C是所有有效注释串

  形成的语言. C中的成员必须以/#开始, #/结束, 并且在开始和结束之间没有#/. 为简便起见,

  所有注释都由符号a和b写成; 因此C的字母表S={a, b, /, #}.

  a. 给出识别C的DFA.

  b. 给出产生C的正则表达式.

图示

描述已自动生成

1.29 使用泵引理证明下述语言不是正则的。

     b. A = { www | w属于{a,b}\* }

文本, 信件

描述已自动生成