1. Write a regular expression for the language of strings over the alphabet {0,1} containing exactly

   two 0s.

   $1^*01^*01^*$

2. Describe in the English language what the following regular expression represents

   (a + b)*bab(a*b*)*

   (a+b)* ==> select either a or b 0 or more times

   bab ==> must appear once in this string, between the first and last part, which can both

   occur 0 or more times

   (a*b*)* ==> can choose a and b and amount of times from 0 or more, and this can repeat 0

   or more times

   In the end, it is any number of (a or b)s, followed by bab, followed by any number of (any

   number of as and any number of bs)

   Accepts strings that contain bab

3. Find a right-linear grammar for the machine below:
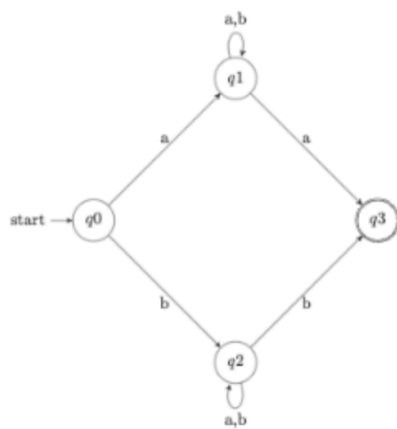

$q_0 \rightarrow aq_0 | bq_1$

$q_1 \rightarrow aq_2$
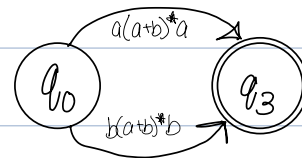
$q_2 \rightarrow bq_3$

$q_3 \rightarrow \lambda | aq_3$

4. Find a left-linear grammar for the same machine from problem #3.

$$q_3 \rightarrow q_3 a \mid q_2 b$$
$$q_2 \rightarrow q_1 a$$
$$q_1 \rightarrow q_0 b$$
$$q_0 \rightarrow q_0 a \mid \lambda$$

5. Using the state-bypass and elimination technique, convert the NFA below into a regular expression:

Path	Exp
$q_0 \rightarrow q_1 \rightarrow q_3$	$a(a+b)^*a$
$q_0 \rightarrow q_4 \rightarrow q_4$	$b(a+b)^*b$

$$\boxed{\left(a(a+b)^*a\right) + \left(b(a+b)^*b\right)}$$