MA 20500, Spring 2018 Exam 3 Name: **Solutions**

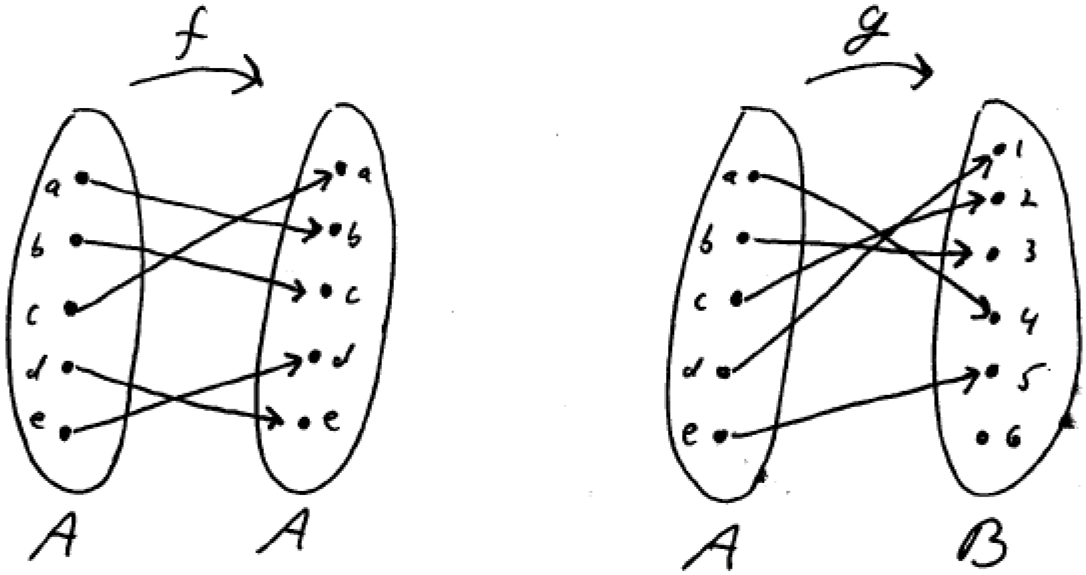
1. Let $A = \{a, b, c, d, e\}$ and $B = \{1, 2, 3, 4, 5, 6\}$.

   Let $f : A \to A$ be defined as $f(a) = b$, $f(b) = c$, $f(c) = a$, $f(d) = e$, $f(e) = d$.

   Let $g : A \to B$ be defined as $g(a) = 4$, $g(b) = 3$, $g(c) = 2$, $g(d) = 1$, $g(e) = 5$.

   (a) (8 points) Draw an arrow diagram for each of f and g.

Solution:



   (b) (4 points) Is f one-to-one or onto? (Be sure to answer both questions.)

Solution:

f is one-to-one.

f is onto.

   (c) (4 points) Is g one-to-one or onto? (Be sure to answer both questions.)

Solution:

g is one-to-one.

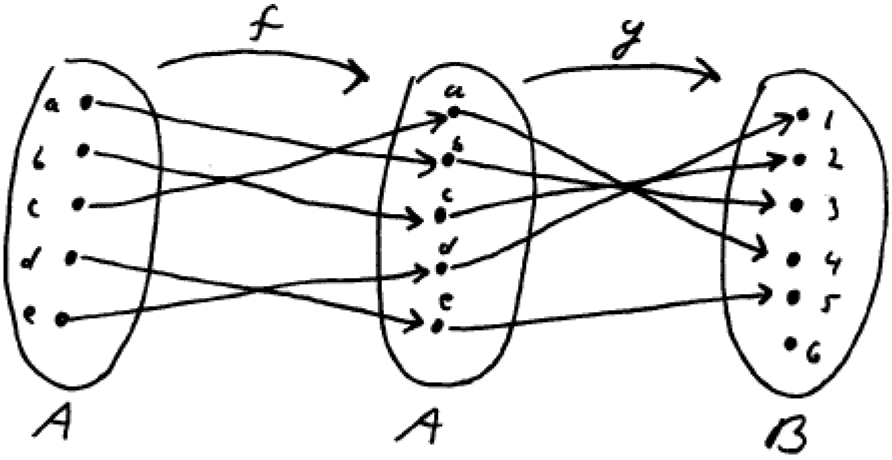
g is not onto.

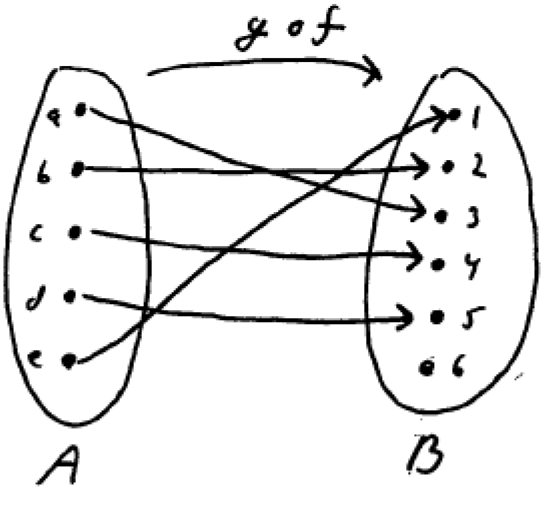
(d) (10 points) One of the functions $g \circ f$ and $f \circ g$ makes sense and the other one doesn't. Which one doesn't make sense? Explain why. Which one does make sense? Draw an arrow diagram for the one that does make sense. (Be sure to answer all the parts of this question.)

Solution:

The function $f \circ g$ does not make sense because the outputs of the function g are in the set B, but the inputs to the function f must come from the set A. So you cannot do the composition $f(g(x))$ (an output from g is not an input for f).

The function $g \circ f : A \to B$ does make sense. The outputs from f are in set A which is the set where the inputs to g must come from. Here is an arrow diagram of f followed by g, and below it is the arrow diagram for $g \circ f$.





2. (10 points) Find a regular expression for the language consisting of all strings of a's, b's and c's which have all three of the following properties,

- begin with either a or b,
- end with c,
- and contain the string "cab".

Solution:

```
(a|b)(a|b|c)*cab(a|b|c)*c
```

3. Let $L_1 = \{ab,\ ba\}$ and let $L_2 = \{c,\ cd\}$.

(a) (5 points) Find all the words in the language $L_3 = L_2 \cup L_1$.

Solution:

$$\{c,\ cd,\ ab,\ ba\}$$

(b) (5 points) Find all the words in the language $L_4 = L_2 \cdot L_1$.

Solution:

$$\{cab,\ cba,\ cdab,\ cdba\}$$

(c) (5 points) Find all the words in the language $L_5 = L_2 \cdot (L_1 \cup L_2)$.

Solution:

$$\{cc,\ ccd,\ cab,\ cba,\ cdc,\ cdcd,\ cdab,\ cdba\}$$

(d) (5 points) Write down a regular expression for the language $(L_1)*$.

Solution:

```
(ab|ba)*
```

4. Find all the words in the language defined by each of these regular expressions.

   (a) (5 points) $(0\,|\,1)(00\,|\,11)\,11$

   Solution:

   $\{00011,\ 01111,\ 10011,\ 11111\}$

   (b) (5 points) $(0\,|\,(01\,|\,10))\,111$

   Solution:

   $\{0111,\ 01111,\ 10111\}$

5. (8 points) Explain the difference between this regular expression $(a|b)*$ and this regular expression $(ab)*$ (and don't answer that one has the | symbol in it and the other doesn't; explain the difference in their meanings).

   Solution:

   The regular expression $(a|b)*$ asks you to repeatedly choose between a and b. It defines the language of all possible words made up of the letters a and b.

   The regular expression $(ab)*$ asks you to repeat the string ab zero or more times. It defines words that alternate the letter a with the letter b.

6. (10 points) When this regular expression,

```
\b\d[12](x([0-9]{2}))([a-z]*)\1\b
```

is applied to this string,

```
92x22xyzx22  93x22xyzx22  01x45dogx45  02x00x00   x12applex12
```

determine what substrings get matched by the regular expression and specify, for each matched substring, what the values of `\1` and `\2` and `\3` are.

Solution:

The first, third, and fourth words are matched.

When

```
92x22xyzx22
```

is matched

```
\1 is x22
\2 is 22
\3 is xyz
```

When

```
01x45dogx45
```

is matched

```
\1 is x45
\2 is 45
\3 is dog
```

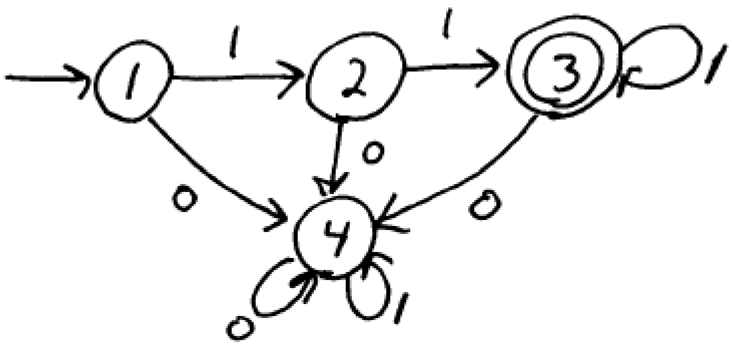
When

```
02x00x00
```

is matched

```
\1 is x00
\2 is 00
\3 is the empty string
```

7. (8 points) Write the next-state table for the finite-state automaton described by the following transition diagram.



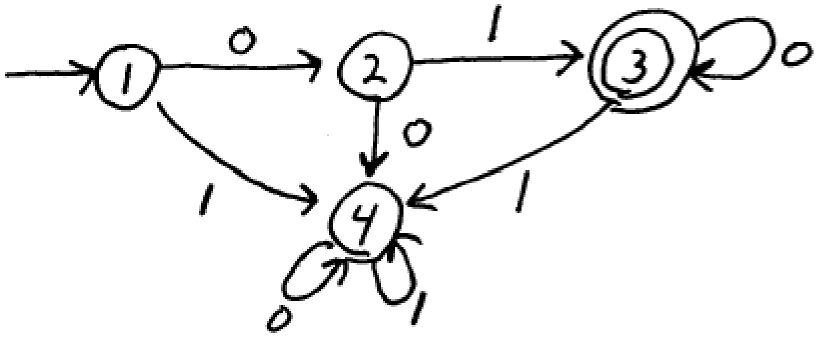
	0	1
1	4	2
2	4	3
3	4	3
4	4	4

8. (8 points) Draw a finite-state automaton that accepts the language defined by this regular expression.

$$0\,1\,0*$$

Hint: You need four states. Only one of them should be an accepting state. One of the four states should act like a "rejecting state," every rejected string will land on that state and stay there. Also, look at the previous problem.

Solution: Modify the finite-state machine from the previous problem.


1
2
3
4
0
1
0
0
1
1
0
1