

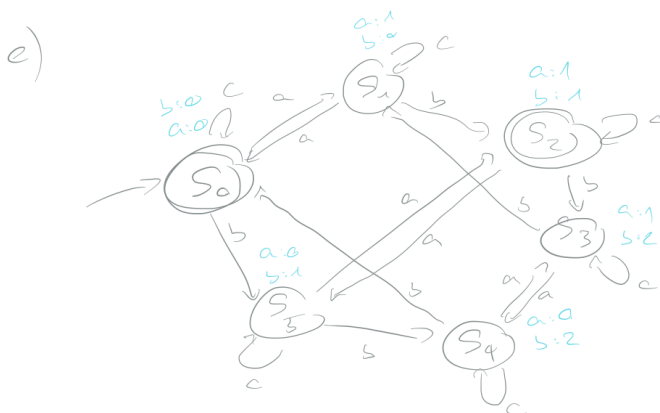
HW1

1.

b) Two 0s or two 1s at the beginning or three contiguous 1s or 0s later in the word followed by any more 0s or 1s.

c) The words accepted start with an unlimited amount of as and bs followed by two contiguous as. After that, two bs, each preceded by an unlimited amount of as. Then, any as and bs before two contiguous as. Finally, an unlimited amount of as and bs. Alternate explanation (with only a and b considered valid letters): Anything, two contiguous as, anything including at least two bs, two contiguous as, anything.

2.



4. c)

- $d : 0, 1, \dots, 9$
- $d' : 1, 2, \dots, 9$
- $\$((d'dd|d'd|d')(\cdot ddd)^*|0).dd$

5. E)

- Not possible using just regular expression (because it is not a regular language because it needs to keep track of the amount of brackets)
- Only possible if recursive symbols are allowed:
 - Bold brackets () signify the symbol
 - $op : \{+ | - | \times | \div\}$

$$L = id|(L op L)|L op L$$

7. (LAST RE)

7) a)

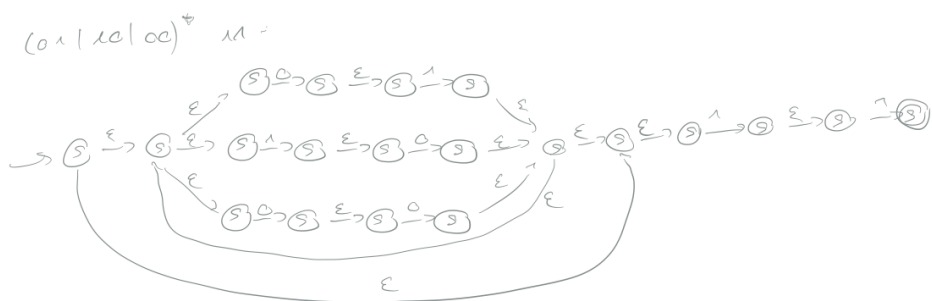
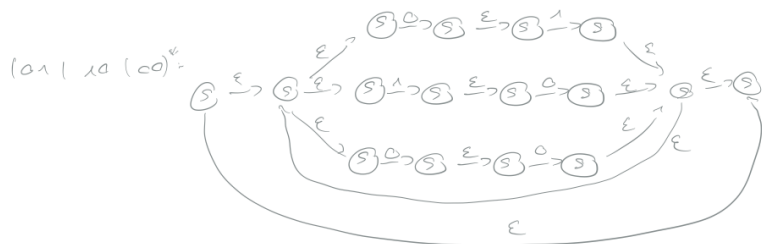
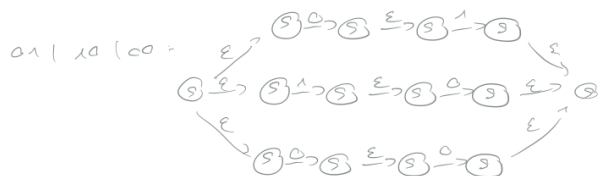
(states not numbered during construction)

01: $\epsilon \rightarrow (S) \xrightarrow{0} (S) \xrightarrow{\epsilon} (S) \xrightarrow{1} (S) \xrightarrow{\epsilon}$

10: $\epsilon \rightarrow (S) \xrightarrow{1} (S) \xrightarrow{\epsilon} (S) \xrightarrow{0} (S) \xrightarrow{\epsilon}$

00: $\epsilon \rightarrow (S) \xrightarrow{0} (S) \xrightarrow{\epsilon} (S) \xrightarrow{0} (S) \xrightarrow{\epsilon}$

11: $\epsilon \rightarrow (S) \xrightarrow{1} (S) \xrightarrow{\epsilon} (S) \xrightarrow{1} (S) \xrightarrow{\epsilon}$



removing unnecessary ϵ :



converting to DFA:

