(a)

$$\begin{aligned} S &= aSd \mid aBc \mid bCd \mid bDc \mid \epsilon \\ B &= aBc \mid bDc \mid \epsilon \\ C &= bCd \mid bDc \mid \epsilon \\ D &= bDc \mid \epsilon \end{aligned}$$

S is the starting non-terminal. The B is the case when there is already c after B (in this case, it could not have d otherwise it is not in the language). C is the case when there is already b before C (it could not have a otherwise it is not in the language). D is the case when there is already b before D and there is already c after D. (in this case it could not have d and b other it's not in the language). S is all the other cases, where it's possible for us to choose either a, b, c, or d. In each case, we have made sure when it's "unfold", the $i + j = k + l$ holds This is achieved by make sure when a or b added on left side, c or d always added the right side. ϵ also in either case won't violate the rule in language.

(b)

$$\begin{gathered} S = 000S2 \mid 001W2 \mid 011W2 \mid 111W2 \mid \epsilon \\ W = 111W2 \mid \epsilon \end{gathered}$$

S is the starting non-terminal. The W is the case when there is already 1 before it. It could not have 0 in it so it won't violate the role in the language. S is all other cases. In each case, we have made sure that when it's unfold, the $k = 3(i + j)$. It's achieved when we add a 2 on right, we add 3 `0` or `1`s or the left side. ϵ also in either case won't violate the rule in language.

(c)

$$\begin{aligned} S &= R_1 | R_2 | R_3 | R_4 | R_5 \\ A &= 0A \mid 1A \mid \epsilon \\ A' &= 0A \mid 1A \mid \#A \mid \epsilon \\ A_1 &= \epsilon \\ A_2 &= \# \\ A_3 &= \#A\# \\ A_4 &= \#A\#A\# \\ R_1 &= A_1 \mid 0R_10 \mid 1R_11 \\ R_2 &= A_2 \mid 0R_20 \mid 1R_21 \\ R_3 &= A_3 \mid 0R_30 \mid 1R_31 \mid R_2\#A \mid A\#R_2 \\ R_4 &= A_4 \mid 0R_40 \mid 1R_41 \mid R_3\#A \mid A\#R_3 \\ R_5 &= A'\#R\#A' \\ R &= A_3 \mid 0R0 \mid 1R1 \end{aligned}$$

S is the starting non-terminal. R_i handles the case when there's exactly i count of $\#$ within the string. R_5 handles the case when there's at least 5 count of $\#$ within the string.

(d)

$$\begin{gathered} S = 0 \mid 1 \mid 0A0 \mid 1A1 \mid 0A1 \mid 1S0 \\ A = 0A \mid 1A \mid \epsilon \end{gathered}$$

S is the starting non-terminal. The A is a case where any string or epsilon is accepted. The case 0, 1, $0A0$, $1A1$, $0A1$ is all the case directly violate the L', so we use A to indicates that in this case we could fill whatever we want. The case $1S0$, is when the string revealed so far conforms to the language L', so we still need put a S inside to make sure the resulted string violates L'.