Solutions to Homework #10

1. Directly use the construction given in the lectures (I only write down δ) :
$\delta(q_0, \Lambda, Z_0) = \{(q, SZ_0)\}$,
$\delta(q, *_1, *_1) = \{(q, \Lambda)\}$, $*_1 = a, b$.
$\delta(q, \Lambda, S) = \{(q, aaB), (q, bbA), (q, AB), (q, aabb)\}$ (This corresponds to rule $S \to aaB|bbA|AB|aabb$)
$\delta(q, \Lambda, A) = \{(q, BA), (q, aB), (q, a)\}$ (This corresponds to rule $A \to BA|aB|a$)
$\delta(q, \Lambda, B) = \{(q, AB), (q, bA), (q, b)\}$ (This corresponds to rule $B \to AB|bA|b$)
$\delta(q, \Lambda, Z_0) = \{(q_1, Z_0)\}$
$A = \{q_1\}$.

2. Directly use the construction given in the lectures :
$[q_0, *_2, *_3] \to *_1[q_0, *_1, *_4][*_4, *_2, *_3]$ for each $*_3, *_4 \in Q$, and $*_1 = 0, 1, *_2 = 0, 1$. (This corresponds to $\delta(q_0, *_1, *_2) = \{(q_0, *_1 *_2)\}$ with $*_1 = 0, 1, *_2 = 0, 1$)
$[q_0, 1, *_1] \to \Lambda[q_{even}, 1, *_1]|\Lambda[q_{odd}, 1, *_1]$ for each $*_1 \in Q$ (This corresponds to $\delta(q_0, \Lambda, 1) = \{(q_{even}, 1), (q_{odd}, 1)\}$)
$[q_0, 0, *_1] \to \Lambda[q_{even}, 0, *_1]|\Lambda[q_{odd}, 0, *_1]$ for each $*_1 \in Q$ (This corresponds to $\delta(q_0, \Lambda, 0) = \{(q_{even}, 0), (q_{odd}, 0)\}$)
$[q_{odd}, *_2, *_3] \to *_1[q_{even}, *_2, *_3]$ for each $*_3 \in Q$ and $*_1 = 0, 1, *_2 = 0, 1$ (This corresponds to $\delta(q_{odd}, *_1, *_2) = \{(q_{even}, *_2)\}$ with $*_1 = 0, 1, *_2 = 0, 1$)
$[q_{even}, 1, q_{even}] \to 1$ (This corresponds to $\delta(q_{even}, 1, 1) = \{(q_{even}, \Lambda)\}$)
$[q_{even}, 0, q_{even}] \to 0$ (This corresponds to $\delta(q_{even}, 0, 0) = \{(q_{even}, \Lambda)\}$)
$[q_{even}, Z_0, q_2] \to \Lambda$ (This corresponds to $\delta(q_{even}, \Lambda, Z_0) = \{(q_2, \Lambda)\}$)
$[q_0, Z_0, q_2] \to \Lambda$ (This corresponds to $\delta(q_0, \Lambda, Z_0) = \{(q_2, \Lambda)\}$)

3. Assume L_1 and L_2 are accepted by NFA M_1 and M_2. Now we describe a PDA M_3 that accepts L_3. M_3 works as follows on an input y. Using its finite control, M_3 simulates the NFA M_1 while reading y. During the simulation, M_3 pushes every input symbol it reads into its pushdown stack. At some moment, M_3 guesses it is the right time that M_3 should start to simulate M_2. Then M_3 checks that M_1 is in an accepting state (otherwise M_3 aborts). After this, M_3 starts to simulate M_2 using its finite control and compares every input symbol M_3 reads with the top of the stack while popping the stack. M_3 accepts the input if after reading the entire input word y, M_2 is an accepting state and the stack is empty (Z_0 is the top).