Problem 1

(a) Prove it.

(1) A → B given

(2) AC → BC by applying *Augmentation Rule IR2* to (1) with *C*

(3) C → D given

(4) BC → BD by applying *Augmentation Rule IR2* to (3) with *B*

(5) AC → BD by applying *Transitive IR3* to (2) and (4).

(b) Disprove it by showing a relational instance refuting the rule derivation

Create a table instance for *R(A,B,C,D)* (a four column table) in which *AB → C* and *C → D* hold but *A → D* does not.

A	B	C	D
1	2	X	5
1	2	X	5
1	3	Y	6
2	5	Y	6

Problem 2

Compute AB^+ = {*ABCDE*} from {*AB → C*, *CE → D*, *A → E*}.
Since *D* is in AB^+, *AB → D* holds.

Problem 3

Consider the relation schema *R* = {*A, B, C, D, E, F, G, H, I, J*} and the set of functional dependencies *F* = {*AB → C, A → DE*, *B → F*, *F → GH*, *D → IJ}* that hold true for *R*. Determine whether each of the decompositions given below has (1) the dependency preservation property, and (2) the lossless join property, with respect to *F*. Also determine which normal form each relation in the decomposition is in.

(a) *D1* = {*R1*, *R2*, *R3*} where *R1* = {*A, B, C, D, E*}, *R2* = {*B, F, G, H*}, and *R3* = {*D, I, J*},

(b) *D2* = {*R1*, *R2*, *R3*, *R4*, *R5*} where *R1* = {*A, B, C, D*}, *R2* = {*D, E*}, *R3* = {*B, F*}, *R4* = {*F, G, H*}, and *R5* = {*D, I, J*}.

3(a)

Dependency Preserving Test

$\mathbf{\Pi_{R1}(F)}$ = {AB -> C, A-> DE} $\Pi_{\mathbf{R2}}\mathbf{(F)}$ = {B -> F, F-> GH} $\mathbf{\Pi_{R3}(F)}$ = {D -> IJ}

Π_{R1}(F) U Π_{R2}(F) U Π_{R3}(F) = {AB -> C, A-> DE, B -> F, F-> GH, D -> IJ} = F

So, it is dependency preserving.

Lossless join

R1 ∩ R2 = {B} R1 – R2 = {A, C, D, E} R2 – R1 = {F, G, H}

B -> {F, G, H} ϵ F^+

R12 = R1 * R2 = {A, B, C, D, E, F, G, H}

R12 ∩ R3 = {D}   R12 – R3 = {A, B, C, E, F, G, H}   R3 – R12 = {I, J}

D -> {I, J} ϵ F^+

So, it is lossless join decomposition.

Normal Form of decomposed relations.
R1 = {A, B, C, D, E} => 1NF. Atomic; partial dependency test fails.

R2 = {B, F, G, H} => 2NF. Transitive dependency test fails.

R3 = {D, I, J} => BCNF. Check the BCNF definition.

3(b)
Dependency Preserving test
{AB -> C, A-> DE, B -> F, F-> GH, D -> IJ}
Π_{R1}(F) = {AB -> C, A->D}
Π_{R2}(F) = {}
Π_{R3}(F) = {B -> F}
Π_{R4}(F) = {F-> GH}
Π_{R5}(F) = {D -> IJ}
Π_{R1}(F) ∪ Π_{R2}(F) ∪ Π_{R3}(F) ∪ Π_{R4}(F) ∪ Π_{R5}(F) = {AB -> C, A-> D, B -> F, F-> GH, D -> IJ}
A -> E is missing, so NOT dependency preserving.

Lossless join test
Using Algorithm 16.3 from the textbook

	A	B	C	D	E	F	G	H	I	J
R1	b1,1	b1,2	b1,3	b1,4	b1,5	b1,6	b1,7	b1,8	b1,9	b1,10
R2	b2,1	b2,2	b2,3	b2,4	b2,5	b2,6	b2,7	b2,8	b2,9	b2,10
R3	b3,1	b3,2	b3,3	b3,4	b3,5	b3,6	b3,7	b3,8	b3,9	b3,10
R4	b4,1	b4,2	b4,3	b4,4	b4,5	b4,6	b4,7	b4,8	b4,9	b4,10
R5	b5,1	b5,2	b5,3	b5,4	b5,5	b5,6	b5,7	b5,8	b5,9	b5,10

	A	B	C	D	E	F	G	H	I	J
R1	a1	a2	a3	a4	b1,5	b1,6	b1,7	b1,8	b1,9	b1,10
R2	b2,1	b2,2	b2,3	a4	a5	b2,6	b2,7	b2,8	b2,9	b2,10
R3	b3,1	a2	b3,3	b3,4	b3,5	a6	b3,7	b3,8	b3,9	b3,10
R4	b4,1	b4,2	b4,3	b4,4	b4,5	a6	a7	a8	b4,9	b4,10
R5	b5,1	b5,2	b5,3	a4	b5,5	b5,6	b5,7	b5,8	a9	a10

	A	B	C	D	E	F	G	H	I	J
R1	a1	a2	a3	a4	b1,5	a6	a7	a8	a9	a10
R2	b2,1	b2,2	b2,3	a4	a5	b2,6	b2,7	b2,8	a9	a10
R3	b3,1	a2	b3,3	b3,4	b3,5	a6	a7	a8	b3,9	b3,10
R4	b4,1	b4,2	b4,3	b4,4	b4,5	a6	a7	a8	b4,9	b4,10

R5	b5,1	b5,2	b5,3	a4	b5,5	b5,6	b5,7	b5,8	a9	a10

	A	B	C	D	E	F	G	H	I	J
R1	a1	a2	a3	a4	b1,5	a6	a7	a8	a9	a10
R2	b2,1	b2,2	b2,3	a4	a5	b2,6	b2,7	b2,8	a9	a10
R3	b3,1	a2	b3,3	b3,4	b3,5	a6	a7	a8	b3,9	b3,10
R4	b4,1	b4,2	b4,3	b4,4	b4,5	a6	a7	a8	b4,9	b4,10
R5	b5,1	b5,2	b5,3	a4	b5,5	b5,6	b5,7	b5,8	a9	a10

This is the final step. No change in the matrix occurs anymore. No row has all a's so it's a lossy decomposition.

Alternate method,
R1 ∩ R2 = {D}   R1 – R2 = {A, B, C}   R2 – R1 = {E}
D -> ABC ϵ F^+ does not hold.

D -> E ϵ F^+ does not hold either.

It is also clear that any natural join with R2 will always generate superfluous rows because DE is key in R2 and no other relation has DE occurring together.

Normal Form of decomposed relations.
R1 = {A, B, C, D} => 1NF. Atomic attributes; partial dependency test fails.
R2 = {D, E} => BCNF. There is no dependency.
R3 = {B, F} => BCNF.
R4 = {F, G, H} => BCNF.
R5 = {D, I, J} => BCNF

Problem 4

(a) R_1 = {*D*, *F*} is in BCNF because given $D \rightarrow F$ (the only nontrivial FD in R_1), *D* is the key is of R_1**.**

R_2 = {*A*, *B*, *C*, *D*, *E*} is in 1NF at best because *E* is partially dependent on *BC,* which is a subset of the key *{A, B, C}* of R_2**.** You can validate *{A, B, C}* is the key of R_2 by computing $\{A, B, C\}^+$ given $\{ABC \rightarrow D, D \rightarrow E, BC \rightarrow E\}$. Notice that $\{A, B, C\}^+$ includes all the attributes of R_2.

(b) R_3 = {*D*, *E*} and R_4 = {*A*, *B*, *C*, *D*}
It is lossless because for the common attribute *D* between R_3 and R_4 , $D \rightarrow E$ is in $\{ABC \rightarrow D, D \rightarrow E, BC \rightarrow E\}^+$ where { *E* } = {*D*, *E*} – {*A*, *B*, *C*, *D*}.
R_3 is in BCNF because given $D \rightarrow E$ (the only nontrivial FD in R_3), *D* is the key of R_3**.**
R_4 is in BCNF because given $ABC \rightarrow D$ (the only nontrivial FD in R_4), *ABC* is the key of R_4**.**