<u>Question 1:</u>

Part a:

BH+: BHADGEFC, so BH is a superkey and BH → AD does not violate BCNF.

D+: DBHAGEFC, so D is a superkey and D → BH does not violate BCNF.

BCE+: BCEF, so BCE → F violates BCNF.

F+: FC, so F → C violates BCNF.

A+: AGEFC, so A → GEF violates BCNF.

Part b:

Decomposing R using BCE → F gives us R1 = BCEF and R2 = BCEADHG.

Projecting the FDs onto R1:

B	C	E	F	Closure	FDs
x				B+: B	None
	x			C+: C	None
		x		E+: E	None
			x	F+: FC	F → C; violates BCNF

We must decompose R1 further. Using F → C, we get R3 = CF and R4 = BEF.

Projecting the FDs onto R3:

C	F	Closure	FDs
x		C+: C	None
	x	F+: FC	F → C; F is a superkey

R3 satisfies BCNF.

Projecting the FDs onto R4:

B	E	F	Closure	FDs
x			B+: B	None
	x		E+: E	None
		x	F+: FC	None
x	x		BE+: BE	None
x		x	BF+: BFC	None
	x	x	EF+: EFC	None

R4 satisfies BCNF.

Projecting the FDs onto R2:

A	B	C	D	E	G	H	Closure	FDs
x							A+: AGEFC	A → GE; violates BCNF

We must decompose R2 further. Using A → GE, we get R5 = AGE and R6 = BCDH.

Projecting the FDs onto R5:

A	E	G	Closure	FDs
x			A+:AGEFC	A → GE; A is a superkey
	x		E+:E	None
		x	G+: G	None
We can ignore any superset of A				
	x	x	EG+: EG	None

R5 satisfies BCNF.

Projecting the FDs onto R6:

B	C	D	H	Closure	FDs
x				B+: B	None
	x			C+: C	None
		x		D+: DBHAGEFC	D → BCH; D is a superkey
			x	H+: H	None
We can ignore any superset of D					
x	x			BC+: BC	None
x			x	BH+: BHADGEFC	BH → CD; BH is a superkey
	x		x	CH+: CH	None
We can ignore any superset of BH					

R6 satisfies BCNF.

Therefore our final decomposition is:

1. R3 = FC, with FD F → C.
2. R4 = BEF, with no FDs.
3. R5 = AEG, with FD A → EG.
4. R6 = BCDH, with FDs D→ BCH and BH → CD.

Question 2:

Part a:

A+: A, B+: B, C+: C, D+: DABGFEC, E+: E, F+: F, G+: G

This means that D is a key, while the other attributes only provide trivial closures. Because D is a key, we can disregard any superset of D. However, given that D is not found on the RHS of any FD, it means that even if we use every attribute besides D, we still will not get a key.

ABCEFG+: ABCEFG

Therefore, D is the only key for relation R.

Part b:

To get a minimal basis, we must first get rid of any redundant FDs. The current set of FDs, S is as follows;

S = { DBE → FC, CD → AF, D → AB, D → G, BADE → C, ABD → E, D → F, EF → B }

To start, we will break up the FDs so that the RHS is composed of singletons.

S' = { DBE → F, DBE → C, CD → A, CD → F, D → A, D → B, D → G, BADE → C, ABD → E, D → F, EF → B }

Following this, we can see there are some cases of redundant FDs, where a LHS of a FD determines the same RHS as one of its subsets. In the case of D → F, both DBE → F and CD → F are redundant; D → A makes CD → A redundant; and DBE → C makes BADE → C redundant. Therefore, we can discard these FDs and produce a new set that we will call S1;

S1 = { DBE → C, D → A, D → B, D → G, ABD → E, D → F, EF → B }

Next, we can strengthen the FDs by reducing the number of attributes on the LHS. We do this by looking at the closures of each subset of the LHS and see if we can get the RHS with fewer attributes. In the case of both DBE → C and ABD → E, we already know from part a) that D is a key. Therefore, the FDs D → C and D → E are sufficient. In the case of EF → B, as we again saw in part a), the closures for E and F are trivial, and EF cannot be broken down into any further subsets, meaning the FD remains as is. This gives us our new set of FDs, S2;

S2 = { D → C, D → A, D → B, D → G, D → E, D → F, EF → B }

Finally, we must discard any other redundant FDs that have come about as a result of our simplification process. We do this by calculating the closures of the LHS' subsets for each FD and see if we can obtain the RHS even if we exclude that FD from our projection. If so, we discard it and exclude it when we calculate the rest of the closures.

#	FD	Exclude	Closure	Decision
1	D → C	1	No way to get C	Keep
2	D → A	2	No way to get A	Keep
3	D → B	3	D+: DHAGEFCB	Discard
4	D → G	3, 4	No way to get G	Keep
5	D → E	3, 5	No way to get E	Keep
6	D → F	3, 6	No way to get F	Keep
7	EF → B	3, 7	No way to get B	Keep

With this we have performed all of the simplifications we can, giving us the final set of FDs for our minimal basis;

M = { D → A, D → C, D → E, D → F, D → G, EF → B }

Part c:

To perform our 3NF decomposition, we must first merge the RHS of our FDs to reduce the number of relations we produce.

M' = { D → ACEFG, EF → B }

With this we get the following relations;

R1(A, C, D, E, F, G)   R2(B, E, F)

Because the two relations only have E and F in common, we can keep both of them. Furthermore, since D is a key of R, there is no need to introduce any further relations. This means R1 and R2 are our final set of relations.

Part d:

Since B is not an attribute of R1, the FD EF → B does not really apply, meaning it does not violate BCNF. On the other hand, D → ACEFG cannot be projected onto R2 since D is not an attribute of that relation, so BCNF is not violated here either. This means that this schema does not allow for any redundancy.