Task 1)

A)
Transitivity: If X → Y and Y → Z, then X → Z
Gives us in our case:
C → {A,D} and A → {C,B},

This gives us

{C} →{A},{A} →{B}

B)
Transitivity: If X → Y and Y → Z, then X → Z
Gives us:
{A} --> {C}, {C} --> {D}, {D,E} –-> {F}

{C} -> {B}

Task 2)

A)

Given that we know {A} we can use Armstrong’s rules to also find {B,C,D}

F+ = {A,B,C,D}

B) }

From Transitivity we see that {C,E} gives us all attributes in the relation. {C,E} can thereby count as a super key-couple and X --> R. F+ = {A,B,C,D,E,F}.

Task 3)
A) {A,B} is a candidate key. We cant find any smaller subset of attributes that is a super key. Transitivity gives us {D} --> {B}.

{A,D} is also a candidate key. The same relations hold after we use {D}->{B}, Transitivity.

B) If R is BCNF we should be able to reach all attributes within R from our super key without taking unnecessary long paths. In our case we can reach all attributes from the Candidate {A,B}. We can also reach all attributes from {A,D} but with a longer path. If we excluded FD2 and FD3 we would have a BCNF with {A,B} as super key.

C) {A,B} is key

- FD1 is bcnf.
- FD2 is not.
- FD3 is not.
  - Using FD2, we decompose R.
    - R1(<u>E, F</u>) with FD2.

                - {E} -> {F}
            - R2(A, B, C, D, E) with FD1, FD3
                - {A, B} -> {C}
                - {A, B} -> {D}
                - {A, B} -> {E}

- R1 is in BCNF but not R2 —-> Decompose R2.
- R2 is not in BCNF.
    - Using FD3, we decompose R2.
        - R2A (D, B) with FD3
            - {D} -> {B}
        - R2B (A, C, D, E) with FD1
            - {A} -> {C}
            - {A} -> {D}
            - {A} -> {E}
- R2A & R2B is in BCNF (FD1 is in BCNF)
- The result of decomposing R consists of R1, R2A & R2B.

Task 4)

A)

Relation schema R with a set F of functional dependencies is in BCNF if for every non-trivial FD X→Y in F+ we have that X is a superkey

BC is our candidate key. -> FD3 is not in BCNF

- FD1: {A,B,C} → {D,E}
  FD2: {B,C,D} → {A,E}
  FD3: {C} → {D}

B)

- FD1 is bcnf.
- FD2 is bcnf
- FD3 is not bcnf.
    - Using FD1, we decompose R.
        - R1(A, B, C, D, E) with FD1.
            - {A, B, C} -> {D}
            - {A, B, C} -> {E}
        - R2(B, C, A) with FD2, FD3
            - {B, C} -> {A}

The result of decomposing R consists of R1, R2