***Consider a relation R(A, B, C, D, E) with FD ={AB -> C, CD -> E, C -> A, C -> D, D -> B}. Determine all the candidate keys of relation R. Please include the details of how to find these keys. DO NOT list superkeys. (30pts)***

---

Candidate Keys:
FD ={AB -> C, CD -> E, C -> A, C -> D, D -> B}
$( C )^{+}$ = CADBE
$( AB )^{+}$ = ABCDE
$( AD )^{+}$ = ADBCE

C, AB, & AD. You can find Candidate Keys as the values that have no redundancy in their attributes.

---

***Given relation schema R(A, B, C, D, E, F) with associated functional dependencies FD={AB -> C, AD->B, C->B, F-> AD, F-> E}, use Armstrong's Axioms to prove that F is a superkey of R. (20pts)***

---

We use Armstrong's Axioms to infer the functional dependencies.

- F-> AD and AD-> B (given)
    - F -> B (Axiom of transitivity)
- F-> E (given)
- F->AD (given)
    - F->A and F->D (Decomposition)
- F->AB (given)
    - F->B and F->A (Union)
- AB -> C (given)
    - F->C (Axiom of transitivity)
- **Therefore: F is a superkey of R as {F} += {A,B,C,D,E,F}.**

---

*Consider a relation StudentInfo(StudNo, StudName, Major, Advisor, CourseNo, Ctitle, InstrucName, InstrucLocn, Grade). Assume it has the following functional dependencies:*
*FD1: StudNo -> StudName*
*FD2: CourseNo -> Ctitle,InstrucName*
*FD3: InstrucName -> InstrucLocn*
*FD4: StudNo,CourseNo,Major -> Grade*
*FD5: StudNo,Major -> Advisor*
*FD6: Advisor -> Major*

*To make the StudentInfo table to be BCNF, the StudentInfo table has been decomposed into the following relations:*

- *Student(StudNo, StudName)*
- *StudMajor(StudNo, Major, Advisor)*
- *StudCourse(StudNo, Major, CourseNo, Ctitle, Instrucname, InstrucLocn, Grade).*

***Does this decomposition satisfy BCNF? If not, apply further decomposition to make all relations satisfy BCNF (50pts).***

---

This decomposition does not satisfy BCNF. Here we breakdown and check the decomposition and go further if need be.

- *Student(StudNo, StudName)*
    - Rule 1: StudNo is a superkey
    - Rule 2- Rule 6: Not applicable dependencies
    - Table: **Student(StudNo, StudName)**

- *StudMajor(StudNo, Major, Advisor)*
    - Rule 1 - Rule 4: Not applicable dependencies
    - Rule 5: (StudNo, Major) is superkey
    - Rule 6: **Advisor not superkey, doesn't satisfy BCNF**
        - We split StudMajor(StudNo, Major, Advisor) into two tables:
        - Make tables: **StudMajor2(StudNo, Advisor)** and **Advisor(Advisor, Major)** these now satisfy the BCNF conditions.

- *StudCourse(StudNo, Major, CourseNo, Ctitle, Instrucname, InstrucLocn, Grade)*
    - Rule 1, Rule 5, & Rule 6: Not applicable
    - Rule 4: StudNo, CourseNo, Major is superkey
    - Rule 2 & 3: **doesn't satisfy BCNF (CourseNo and InstrucName aren't superkeys)**
        - First split based on Rule 2:
        - **StudCourse2(StudNo, Major, CourseNo, InstrucLocn, Grade)**
            - Key is (StudNO, CourseNo, Major). The Rule 3 is no longer applicable and StudCourse2 now satisfies BCNF.
        - **Course(CourseNo, Ctitle, InstrucName)**
            - Key is CourseNo. The only rule that holds is FD2. This course table now satisfies BCNF.

**The final decomposition tables are:**

Student(StudNo, StudName)
StudMajor2(StudNo, Advisor)
Advisor(Advisor, Major)
StudCourse2(StudNo, Major, CourseNo, InstrucLocn, Grade)
Course(CourseNo, Ctitle, InstrucName)