Consider the following (partial) schedule in a centralized database system. Assume the initial values of data items A, B, C, D, and E are 10, 20, 30, 40, and 50, respectively. Assume logging with immediate updates is used for recovery. Assume the system has crashed right after step 25.

1. \( R(A) \)
2. \( R(D) \)
3. \( R(B) \)
4. \( R(E) \)
5. \( B = B - A \)
6. \( W(B) \)
7. \( R(C) \)
8. commit
9. \( R(B) \)
10. \( R(A) \)
11. \( B = B + E \)
12. \( R(A) \)
13. \( W(B) \)
14. \( D = D + A \)
15. commit
16. \( W(D) \)
17. \( R(B) \)
18. \( R(E) \)
19. \( B = B + A \)
20. \( W(B) \)
21. \( C = C + 20 \)
22. \( E = E - 20 \)
23. \( W(C) \)
24. commit
25. \( W(E) \)

(a) Show the complete log just before the crash.

(b) Which portion of the log is in stable storage when the system crashes? Why?

(c) When the system is brought back up, what are the recovery actions it performs, and in which order are the actions performed? Be specific and give the details in the sequence they are performed (for example, you should say the value of \( X \) is restored to \( V \), where an undo is performed corresponding to an entry < \( T_i, X, V, U > \)).

(d) Answer parts (a) through (c) again assuming a checkpoint is taken immediately after step 13.