The following table (adapted from McConnell, 1998) shows that mistakes in software engineering can be related to the people, the process, the product and/or the technology.

|  |  |  |  |
| --- | --- | --- | --- |
| **Mistakes in Software Engineering** | | | |
| **People Related Mistakes** | **Process-Related Mistakes** | **Product-Related Mistakes** | **Technology-Related Mistakes** |
| 1. Undermined motivation  2. Weak/problem employees  3. Heroics  4. Adding people to a late project  5. Noisy, crowded offices  6. Friction between developers and customers  7. Unrealistic expectations  8. Lack of effective project sponsorship  9. Lack of stakeholder buy-in  10. Lack of user input  11. Politics placed over substance  12. Wishful thinking | 13. Overly optimistic schedules  14. Insufficient risk management  15. Insufficient planning  16. Abandonment of planning under pressure  17. Wasted time during the front end  18. Inadequate upstream activities  19. Inadequate design  20. Inadequate quality assurance  21. Insufficient management controls  22. Omitting necessary tasks from estimates  23. Planning to catch up later  24. “Code-like-hell” programming | 25. Requirements gold-plating  26. Feature creep  27. Developer gold-plating  28. Push-me, pull-me negotiation | 29. Silver-bullet syndrome  30. Overestimated savings from new tools or methods  31. Switching tools in the middle of a project  32. Lack of automated source-code control |

**QUESTION:**

**Listed above are the mistakes that cause of software problems. Which do you think is a primary cause? Why?**