1. Is there a fault in the program? If so, locate it and explain.
   Yes. The loop starts at index 1 instead of 0, so the first attendance value is never checked. This causes the absence counting to be incorrect as it will always skip over the first value in the list since the first index of a list always starts at 0.

2. Define a test case that does not execute the fault. If impossible, explain.
   It is impossible to define a valid test case that does not execute the fault. The loop always starts at index 1 for any 10-element attendance record, so the code is always executed, unless a vector of size 0 or 1 was used; however, all attendance records are supposed to be a vector of size 10, so this would not be possible.

3. Define a test case that executes the fault but does not result in an error state.
   Test case:
   [1, 1, 1, 1, 1, 1, 1, 1, 1, 1]
   Explanation:
   Since this student was present everyday, the program does not count any absences. However, the fault is still executed since it still can iterate (and does iterate) through indexes 1-9, and not 0-9.

4. Define a test case that results in an error state but not a failure.
   Test case:
   [0, 0, 1, 1, 1, 1, 1, 1, 1, 1]
   Explanation:
   Skipping index 0 leads to the total amount of absences being incorrect (the program counts 1 absence instead of 2), creating an error state. However, the expected output and the real output match, so no failure occurs.

5. Define a test case that results in a failure.
   Test case:
   [0, 0, 0, 1, 1, 1, 1, 1, 1, 1]
   Explanation:
   The correct behavior is to count all 3 absences and return true. However, since indexing starts at 1, it only counts 2 absences and returns false. This incorrect output results in a failure.