## Bước 1 – Viết code theo mô tả

Viết dấu nhắc yêu cầu AI tạo hàm Python có chức năng:  
- Nhận danh sách điểm của sinh viên (liệt kê các số thực từ 0–10).  
- Trả về điểm trung bình, điểm cao nhất, điểm thấp nhất.  
- Xử lý trường hợp danh sách trống.  
  
Ví dụ dấu nhắc:  
“Hãy viết hàm Python ‘analyze\_score(scores)’ trả về từ điển bao gồm ‘trung bình’, ‘tối đa’ và ‘tối thiểu’. Nếu danh sách trống thì trả về giá trị ‘None’ cho cả 3. Giải thích từng bước logic.”

## Bước 2 – Phân tích và sửa lỗi

Giả sử khi thử nghiệm, chương trình xuất hiện lỗi (ví dụ lỗi chia cho 0).  
Hãy viết dấu nhắc yêu cầu AI phân tích lỗi và sửa mã.  
  
Ví dụ dấu nhắc:  
“Đây là đoạn mã Python tôi viết, nhưng khi truyền ‘scores = []’ thì gặp lỗi ‘ZeroDivisionError’. Hãy chỉ ra nguyên nhân và sửa lại hàm sao cho xử lý an toàn hơn.”

## Bước 3 – Refactor và tối ưu

Khi chức năng hoạt động đúng, bạn muốn tối ưu và làm cho code sạch hơn, dễ mở rộng hơn.  
  
Viết dấu nhắc yêu cầu AI refactor lại đoạn code, ví dụ: viết bình luận, giảm số dòng, chuẩn hóa biến.  
  
Ví dụ dấu nhắc:  
“Hãy cấu trúc lại hàm ‘analyze\_scores’ để code rõ ràng, dễ đọc, có bình luận rõ ràng. Đảm bảo logic không thay đổi.”

## Bước 4 – Thêm test case và kiểm tra logic

Viết dấu nhắc yêu cầu AI tạo test case (dùng pytest hoặc unittest) để đảm bảo chức năng hoạt động đúng trong các trường hợp:  
- Danh sách bình thường  
- Danh sách rỗng  
- Danh sách chỉ có 1 phần tử  
- Danh sách có nhiều giá trị giống nhau  
  
Ví dụ dấu nhắc:  
“Hãy viết 5 test case cho hàm ‘analyze\_scores’ bằng pytest, bao gồm các vấn đề đặc biệt (rỗng, 1 phần tử, nhiều phần tử giống nhau). Giải thích hợp lý để kiểm tra logic.”